

SQL Anywhere サーバ  
文書バージョン: 17 - 2016-05-11

## SQL Anywhere .NET API リファレンス

# 目次

<b>1</b>	<b>SQL Anywhere .NET API リファレンス</b>	<b>12</b>
1.1	SABulkCopy クラス	14
	SABulkCopy コンストラクタ	17
	Close() メソッド	20
	Dispose() メソッド	20
	WriteToServer メソッド	21
	WriteToServerAsync メソッド	24
	BatchSize プロパティ	33
	BulkCopyTimeout プロパティ	33
	ColumnMappings プロパティ	34
	DestinationTableName プロパティ	34
	NotifyAfter プロパティ	35
	SARowsCopied イベント	36
1.2	SABulkCopyColumnMapping クラス	36
	SABulkCopyColumnMapping コンストラクタ	38
	DestinationColumn プロパティ	41
	DestinationOrdinal プロパティ	42
	SourceColumn プロパティ	43
	SourceOrdinal プロパティ	44
1.3	SABulkCopyColumnMappingCollection クラス	44
	Add メソッド	46
	Contains(SABulkCopyColumnMapping) メソッド	50
	CopyTo(SABulkCopyColumnMapping[], int) メソッド	51
	IndexOf(SABulkCopyColumnMapping) メソッド	51
	Remove(SABulkCopyColumnMapping) メソッド	52
	RemoveAt(int) メソッド	52
	this[int index] プロパティ	53
1.4	SACommand クラス	53
	SACommand コンストラクタ	57
	BeginExecuteNonQuery メソッド	60
	BeginExecuteReader メソッド	63
	Cancel() メソッド	68
	CreateDbParameter() メソッド	69
	CreateParameter() メソッド	69

	Dispose(bool) メソッド	70
	EndExecuteNonQuery(IAsyncResult) メソッド	70
	EndExecuteReader(IAsyncResult) メソッド	73
	ExecuteDbDataReader(CommandBehavior) メソッド	75
	ExecuteNonQuery() メソッド	75
	ExecuteReader メソッド	76
	ExecuteReaderAsync メソッド	79
	ExecuteScalar() メソッド	83
	Prepare() メソッド	84
	ResetCommandTimeout() メソッド	85
	CommandText プロパティ	85
	CommandTimeout プロパティ	86
	CommandType プロパティ	86
	Connection プロパティ	87
	DbConnection プロパティ	87
	DbParameterCollection プロパティ	88
	DbTransaction プロパティ	88
	DesignTimeVisible プロパティ	89
	Parameters プロパティ	89
	Transaction プロパティ	90
	UpdatedRowSource プロパティ	91
1.5	SACommandBuilder クラス	91
	SACommandBuilder コンストラクタ	94
	ApplyParameterInfo(DbParameter, DataRow, StatementType, bool) メソッド	95
	DeriveParameters(SACommand) メソッド	96
	GetDeleteCommand メソッド	97
	GetInsertCommand メソッド	100
	GetParameterName メソッド	102
	GetParameterPlaceholder(int) メソッド	104
	GetSchemaTable(DbCommand) メソッド	105
	GetUpdateCommand メソッド	106
	InitializeCommand(DbCommand) メソッド	108
	QuoteIdentifier(string) メソッド	109
	SetRowUpdatingHandler(DbDataAdapter) メソッド	109
	UnquoteIdentifier(string) メソッド	110
	DataAdapter プロパティ	111
1.6	SACommLinksOptionsBuilder クラス	111
	SACommLinksOptionsBuilder コンストラクタ	113
	GetUseLongNameAsKeyword() メソッド	115

	SetUseLongNameAsKeyword(bool) メソッド	115
	ToString() メソッド	116
	All プロパティ	117
	ConnectionString プロパティ	117
	SharedMemory プロパティ	117
	TcpOptionsBuilder プロパティ	118
	TcpOptionsString プロパティ	118
1.7	SACConnection クラス	118
	SACConnection コンストラクタ	122
	BeginDbTransaction(IsolationLevel) メソッド	125
	BeginTransaction メソッド	125
	ChangeDatabase(string) メソッド	129
	ChangePassword(string, string) メソッド	129
	ClearAllPools() メソッド	130
	ClearPool(SACConnection) メソッド	130
	Close() メソッド	131
	CreateCommand() メソッド	131
	CreateDbCommand() メソッド	132
	Dispose(bool) メソッド	133
	EnlistDistributedTransaction(System.EnterpriseServices.ITransaction) メソッド	133
	EnlistTransaction(System.Transactions.Transaction) メソッド	134
	GetSchema メソッド	134
	Open() メソッド	141
	ConnectionString プロパティ	141
	ConnectionTimeout プロパティ	143
	Credential プロパティ	143
	Database プロパティ	144
	DataSource プロパティ	144
	InitString プロパティ	145
	ServerVersion プロパティ	145
	State プロパティ	146
	InfoMessage イベント	146
	StateChange イベント	147
1.8	SACConnectionStringBuilder クラス	147
	SACConnectionStringBuilder コンストラクタ	154
	AppInfo プロパティ	155
	AutoStart プロパティ	156
	AutoStop プロパティ	156
	Charset プロパティ	156

CommBufferSize プロパティ	157
CommLinks プロパティ	157
Compress プロパティ	157
CompressionThreshold プロパティ	158
ConnectionLifetime プロパティ	158
ConnectionName プロパティ	158
ConnectionPool プロパティ	159
ConnectionReset プロパティ	159
ConnectionTimeout プロパティ	159
DatabaseFile プロパティ	160
DatabaseKey プロパティ	160
DatabaseName プロパティ	160
DatabaseSwitches プロパティ	161
DataSourceName プロパティ	161
DisableMultiRowFetch プロパティ	161
Elevate プロパティ	162
EncryptedPassword プロパティ	162
Encryption プロパティ	162
Enlist プロパティ	163
FileDataSourceName プロパティ	163
ForceStart プロパティ	163
Host プロパティ	164
IdleTimeout プロパティ	164
InitString プロパティ	164
Integrated プロパティ	165
Kerberos プロパティ	165
Language プロパティ	165
LazyClose プロパティ	166
LivenessTimeout プロパティ	166
LogFile プロパティ	166
MaxPoolSize プロパティ	167
MinPoolSize プロパティ	167
NewPassword プロパティ	167
NodeType プロパティ	168
Password プロパティ	168
PersistSecurityInfo プロパティ	168
Pooling プロパティ	169
PrefetchBuffer プロパティ	169
PrefetchRows プロパティ	169

	RetryConnectionTimeout プロパティ	170
	ServerName プロパティ	170
	StartLine プロパティ	170
	Unconditional プロパティ	171
	UserID プロパティ	171
1.9	SACConnectionStringBuilderBase クラス	171
	ContainsKey(string) メソッド	173
	GetKeyword(string) メソッド	174
	GetUseLongNameAsKeyword() メソッド	174
	Remove(string) メソッド	175
	SetUseLongNameAsKeyword(bool) メソッド	176
	ShouldSerialize(string) メソッド	176
	TryGetValue(string, out object) メソッド	177
	Keys プロパティ	178
	this[string keyword] プロパティ	178
1.10	SACredential クラス	179
	SACredential(string, SecureString) コンストラクタ	180
	Password プロパティ	180
	UserId プロパティ	181
1.11	SADDataAdapter クラス	181
	SADDataAdapter コンストラクタ	185
	ClearBatch() メソッド	189
	CreateRowUpdatedEvent(DataRow, IDbCommand, StatementType, DataTableMapping) メソッド	189
	CreateRowUpdatingEvent(DataRow, IDbCommand, StatementType, DataTableMapping) メソッド	190
	Dispose(bool) メソッド	191
	Fill メソッド	192
	FillSchema メソッド	195
	GetFillParameters() メソッド	198
	InitializeBatching() メソッド	198
	OnRowUpdated(RowUpdatedEventArgs) メソッド	199
	OnRowUpdating(RowUpdatingEventArgs) メソッド	199
	TerminateBatching() メソッド	200
	Update(DataRow[], DataTableMapping) メソッド	200
	DeleteCommand プロパティ	201
	InsertCommand プロパティ	202
	SelectCommand プロパティ	202
	TableMappings プロパティ	203

	UpdateBatchSize プロパティ	204
	UpdateCommand プロパティ	204
	RowUpdated イベント	205
	RowUpdating イベント	205
1.12	SADataReader クラス	206
	Close() メソッド	211
	GetBoolean(int) メソッド	212
	GetByte(int) メソッド	213
	GetBytes(int, long, byte[], int, int) メソッド	213
	GetChar(int) メソッド	214
	GetChars(int, long, char[], int, int) メソッド	215
	GetData(int) メソッド	217
	GetDataTypeName(int) メソッド	217
	GetDateTime(int) メソッド	218
	GetDateTimeOffset(int) メソッド	219
	GetDecimal(int) メソッド	220
	GetDouble(int) メソッド	221
	GetEnumerator() メソッド	222
	GetFieldType(int) メソッド	222
	GetFloat(int) メソッド	223
	GetGuid(int) メソッド	224
	GetInt16(int) メソッド	225
	GetInt32(int) メソッド	225
	GetInt64(int) メソッド	226
	GetName(int) メソッド	227
	GetOrdinal(string) メソッド	227
	GetSchemaTable() メソッド	228
	GetString(int) メソッド	230
	GetTimeSpan(int) メソッド	231
	GetUInt16(int) メソッド	232
	GetUInt32(int) メソッド	233
	GetUInt64(int) メソッド	233
	GetValue メソッド	234
	GetValues(object[]) メソッド	236
	IsDBNull(int) メソッド	237
	myDispose() メソッド	238
	NextResult() メソッド	238
	Read() メソッド	239
	Depth プロパティ	240

	FieldCount プロパティ	240
	HasRows プロパティ	241
	IsClosed プロパティ	241
	RecordsAffected プロパティ	242
	this プロパティ	242
1.13	SADataSourceEnumerator クラス	244
	GetDataSources() メソッド	245
	Instance プロパティ	245
1.14	SADefault クラス	246
1.15	SAError クラス	246
	ToString() メソッド	248
	Message プロパティ	248
	NativeError プロパティ	248
	Source プロパティ	249
	SqlState プロパティ	249
1.16	SAErrorCollection クラス	249
	CopyTo(Array, int) メソッド	251
	GetEnumerator() メソッド	251
	Count プロパティ	252
	this[int index] プロパティ	252
1.17	SAException クラス	253
	GetObjectData(SerializationInfo, StreamingContext) メソッド	254
	Errors プロパティ	255
	Message プロパティ	255
	NativeError プロパティ	256
	Source プロパティ	256
1.18	SAFactory クラス	257
	CreateCommand() メソッド	259
	CreateCommandBuilder() メソッド	260
	CreateConnection() メソッド	260
	CreateConnectionStringBuilder() メソッド	261
	CreateDataAdapter() メソッド	262
	CreateDataSourceEnumerator() メソッド	262
	CreateParameter() メソッド	263
	CreatePermission(PermissionState) メソッド	263
	CanCreateDataSourceEnumerator プロパティ	264
1.19	SAInfoMessageEventArgs クラス	265
	ToString() メソッド	266
	Errors プロパティ	266



	Message プロパティ	267
	MessageType プロパティ	267
	NativeError プロパティ	268
	Source プロパティ	268
1.20	SAMetaDataCollectionNames クラス	268
1.21	SAPParameter クラス	270
	SAPParameter コンストラクタ	273
	ResetDbType() メソッド	278
	ToString() メソッド	279
	DbType プロパティ	279
	Direction プロパティ	280
	Nullable プロパティ	280
	Offset プロパティ	281
	ParameterName プロパティ	281
	Precision プロパティ	282
	SADbType プロパティ	282
	Scale プロパティ	283
	Size プロパティ	283
	SourceColumn プロパティ	284
	SourceColumnNullMapping プロパティ	285
	SourceVersion プロパティ	285
	Value プロパティ	286
1.22	SAPParameterCollection クラス	286
	Add メソッド	289
	AddRange メソッド	296
	AddWithValue(string, object) メソッド	297
	Clear() メソッド	298
	Contains メソッド	298
	CopyTo(Array, int) メソッド	300
	GetEnumerator() メソッド	301
	GetParameter メソッド	302
	IndexOf メソッド	304
	Insert(int, object) メソッド	306
	Remove(object) メソッド	306
	RemoveAt メソッド	307
	SetParameter メソッド	308
	Count プロパティ	310
	IsFixedSize プロパティ	311
	IsReadOnly プロパティ	311

	IsSynchronized プロパティ	312
	SyncRoot プロパティ	312
	this プロパティ	313
1.23	SAPermission クラス	315
	SAPermission(PermissionState) コンストラクタ	316
	CreateInstance() メソッド	316
1.24	SAPermissionAttribute クラス	317
	SAPermissionAttribute(SecurityAction) コンストラクタ	318
	CreatePermission() メソッド	318
1.25	SARowsCopiedEventArgs クラス	319
	SARowsCopiedEventArgs(long) コンストラクタ	320
	Abort プロパティ	320
	RowsCopied プロパティ	320
1.26	SARowUpdatedEventArgs クラス	321
	SARowUpdatedEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) コンストラクタ	322
	Command プロパティ	322
	RecordsAffected プロパティ	323
1.27	SARowUpdatingEventArgs クラス	323
	SARowUpdatingEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) コンストラクタ	324
	Command プロパティ	325
1.28	SAServerSideConnection クラス	325
	Connection プロパティ	326
1.29	SATcpOptionsBuilder クラス	327
	SATcpOptionsBuilder コンストラクタ	330
	ToString() メソッド	332
	Broadcast プロパティ	332
	BroadcastListener プロパティ	332
	ClientPort プロパティ	333
	DoBroadcast プロパティ	333
	Host プロパティ	333
	IPv6 プロパティ	334
	LDAP プロパティ	334
	LocalOnly プロパティ	334
	MyIP プロパティ	335
	ReceiveBufferSize プロパティ	335
	SendBufferSize プロパティ	335
	ServerPort プロパティ	336

	TDS プロパティ	336
	Timeout プロパティ	336
	VerifyServerName プロパティ	337
1.30	SATransaction クラス	337
	Commit() メソッド	339
	Dispose(bool) メソッド	339
	Rollback メソッド	340
	Save(string) メソッド	341
	Connection プロパティ	342
	DbConnection プロパティ	342
	IsolationLevel プロパティ	343
	SAIsolationLevel プロパティ	343
1.31	SAInfoMessageEventHandler(object, SAInfoMessageEventArgs) デリゲート	344
1.32	SARowsCopiedEventHandler(object, SARowsCopiedEventArgs) デリゲート	345
1.33	SARowUpdatedEventHandler(object, SARowUpdatedEventArgs) デリゲート	345
1.34	SARowUpdatingEventHandler(object, SARowUpdatingEventArgs) デリゲート	346
1.35	SABulkCopyOptions 列挙	346
1.36	SADbType 列挙	348
1.37	SAIsolationLevel 列挙	352
1.38	SAMessageType 列挙	354
<b>2</b>	<b>このマニュアルの印刷、再生、および再配布</b>	<b>355</b>

# 1 SQL Anywhere .NET API リファレンス

SQL Anywhere .NET Data Provider の API を含め、SQL Anywhere を .NET で使用します。

## ネームスペース

- `Sap.Data.SQLAnywhere`
- `Sap.SQLAnywhere.Server`

### i 注記

主な **SQL Anywhere** マニュアルをお探しですか。マニュアルをローカルにインストールした場合は、Windows のスタートメニューを使用してアクセスするか (Microsoft Windows)、`C:\Program Files\SQL Anywhere 17\Documentation` にナビゲートします。

また、DocCommentXchange の Web で、主な SQL Anywhere API リファレンスマニュアルにアクセスすることもできます。<http://dcx.sap.com>

このセクションの内容:

#### [SABulkCopy クラス \[14 ページ\]](#)

別のソースのデータを使用して、データベーステーブルを効率的にバルクロードします。

#### [SABulkCopyColumnMapping クラス \[36 ページ\]](#)

SABulkCopy インスタンスのデータソース内のカラムと、インスタンスの送信先テーブル内のカラムの間のマッピングを定義します。

#### [SABulkCopyColumnMappingCollection クラス \[44 ページ\]](#)

System.Collections.CollectionBase から継承した SABulkCopyColumnMapping オブジェクトのコレクションです。

#### [SACommand クラス \[53 ページ\]](#)

データベースに対して実行される SQL 文またはストアードプロシージャです。

#### [SACommandBuilder クラス \[91 ページ\]](#)

DataSet の変更内容を関連するデータベース内のデータに一致させる単一テーブルの SQL 文を生成する方法です。

#### [SACommLinksOptionsBuilder クラス \[111 ページ\]](#)

SAConnection クラスが使用する接続文字列の CommLinks オプション部分を作成および管理する簡単な方法を提供します。

#### [SAConnection クラス \[118 ページ\]](#)

データベースへの接続を表します。

#### [SAConnectionStringBuilder クラス \[147 ページ\]](#)

SACConnection クラスが使用する接続文字列の内容を作成および管理する簡単な方法を提供します。

#### [SACConnectionStringBuilderBase クラス \[171 ページ\]](#)

SACConnectionStringBuilder クラスの基本クラスです。

#### [SACredential クラス \[179 ページ\]](#)

SACredential は、データベースサーバ認証を使用してログインのパスワードを指定する、よりセキュアな方法を提供します。

#### [SADDataAdapter クラス \[181 ページ\]](#)

System.Data.DataSet に入力したりデータベースを更新したりするために使用する一連のコマンドとデータベース接続を表します。

#### [SADDataReader クラス \[206 ページ\]](#)

クエリまたはストアドプロシージャからの読み込み専用、前方専用の結果セットです。

#### [SADDataSourceEnumerator クラス \[244 ページ\]](#)

ローカルネットワーク内で有効なデータベースサーバのすべてのインスタンスを列挙するメカニズムを提供します。

#### [SADefault クラス \[246 ページ\]](#)

デフォルト値が設定されたパラメータを表します。

#### [SAError クラス \[246 ページ\]](#)

データソースによって返された警告またはエラーに関する情報を収集します。

#### [SAErrorCollection クラス \[249 ページ\]](#)

.NET データプロバイダによって生成されたすべてのエラーを収集します。

#### [SAException クラス \[253 ページ\]](#)

データベースサーバが警告またはエラーを返したときにスローされる例外です。

#### [SAFactory クラス \[257 ページ\]](#)

データソースクラスの Sap.Data.SQLAnywhere プロバイダの実装のインスタンスを作成するメソッドのセットを表します。

#### [SAInfoMessageEventArgs クラス \[265 ページ\]](#)

InfoMessage イベントのデータを提供します。

#### [SAMetaDataCollectionNames クラス \[268 ページ\]](#)

メタデータコレクションを取得する SACConnection.GetSchema(string) メソッドで使用する定数のリストを提供します。

#### [SAPParameter クラス \[270 ページ\]](#)

SACCommand のパラメータと必要に応じて DataSet カラムへのマッピングを表します。

#### [SAPParameterCollection クラス \[286 ページ\]](#)

SACCommand オブジェクトのすべてのパラメータと必要に応じて DataSet カラムへのマッピングを表します。

#### [SAPPermission クラス \[315 ページ\]](#)

ユーザがデータソースへのアクセスに適したセキュリティレベルを持っていることを .NET データプロバイダが確認できるようにします。

#### [SAPPermissionAttribute クラス \[317 ページ\]](#)

カスタムセキュリティ属性にセキュリティアクションを関連付けます。

#### [SARowsCopiedEventArgs クラス \[319 ページ\]](#)

SARowsCopiedEventHandler に渡される引数のセットを表します。

[SARowUpdatedEventArgs クラス \[321 ページ\]](#)

RowUpdated イベントのデータを提供します。

[SARowUpdatingEventArgs クラス \[323 ページ\]](#)

RowUpdating イベントのデータを提供します。

[SAServerSideConnection クラス \[325 ページ\]](#)

データベースへのサーバ側の接続を表します。

[SATcpOptionsBuilder クラス \[327 ページ\]](#)

SAConnection オブジェクトが使用する接続文字列の TCP オプション部分を作成および管理する簡単な方法を提供します。

[SATransaction クラス \[337 ページ\]](#)

SQL トランザクションを表します。

[SAInfoMessageEventHandler\(object, SAInfoMessageEventArgs\) デリゲート \[344 ページ\]](#)

SAConnection オブジェクトの SAConnection.InfoMessage イベントを処理するメソッドを表します。

[SARowsCopiedEventHandler\(object, SARowsCopiedEventArgs\) デリゲート \[345 ページ\]](#)

SABulkCopy の SABulkCopy.SARowsCopied イベントを処理するメソッドを表します。

[SARowUpdatedEventHandler\(object, SARowUpdatedEventArgs\) デリゲート \[345 ページ\]](#)

SADataAdapter の RowUpdated イベントを処理するメソッドを表します。

[SARowUpdatingEventHandler\(object, SARowUpdatingEventArgs\) デリゲート \[346 ページ\]](#)

SADataAdapter の RowUpdating イベントを処理するメソッドを表します。

[SABulkCopyOptions 列挙 \[346 ページ\]](#)

SABulkCopy のインスタンスで使用する 1 つ以上のオプションを指定するビット単位フラグです。

[SADbType 列挙 \[348 ページ\]](#)

データベースサーバの .NET データベースデータ型を列挙します。

[SAIsolationLevel 列挙 \[352 ページ\]](#)

データベースサーバの独立性レベルを指定します。

[SAMessageType 列挙 \[354 ページ\]](#)

メッセージのタイプを示します。

## 1.1 SABulkCopy クラス

別のソースのデータを使用して、データベーステーブルを効率的にバルクロードします。

### 構文

#### Visual Basic

```
Public NotInheritable Class SABulkCopy Implements System.IDisposable
```

#### C#

```
public sealed class SABulkCopy : System.IDisposable
```

## メンバー

SABulkCopy のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変数とタイプ	コンストラクタ	説明
public	<a href="#">SABulkCopy [17 ページ]</a>	SABulkCopy オブジェクトを初期化します。

### メソッド

変数とタイプ	メソッド	説明
public void	<a href="#">Close() [20 ページ]</a>	SABulkCopy インスタンスを閉じます。
public void	<a href="#">Dispose() [20 ページ]</a>	SABulkCopy インスタンスを破棄します。
public void	<a href="#">WriteToServer [21 ページ]</a>	指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public Task	<a href="#">WriteToServerAsync [24 ページ]</a>	指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### プロパティ

変数とタイプ	プロパティ	説明
public int	<a href="#">BatchSize [33 ページ]</a>	各バッチの中のローの数を取得または設定します。
public int	<a href="#">BulkCopyTimeout [33 ページ]</a>	タイムアウトするまでに完了するオペレーションの秒数を取得または設定します。
public SABulkCopyColumnMappingCollection	<a href="#">ColumnMappings [34 ページ]</a>	SABulkCopyColumnMapping 項目のコレクションを返します。
public string	<a href="#">DestinationTableName [34 ページ]</a>	データベースサーバ上の送信先テーブルの名前を取得または設定します。
public int	<a href="#">NotifyAfter [35 ページ]</a>	通知イベントが生成されるまでに処理されるローの数を取得または設定します。

### イベント

変数とタイプ	イベント	説明
public SARowsCopiedEventHandler	<a href="#">SARowsCopied [36 ページ]</a>	NotifyAfter プロパティで指定される数のローが処理されるたびに、このイベントが発生します。

## 備考

実装: System.IDisposable

このセクションの内容:

### [SABulkCopy コンストラクタ \[17 ページ\]](#)

SABulkCopy オブジェクトを初期化します。

### [Close\(\) メソッド \[20 ページ\]](#)

SABulkCopy インスタンスを閉じます。

### [Dispose\(\) メソッド \[20 ページ\]](#)

SABulkCopy インスタンスを破棄します。

### [WriteToServer メソッド \[21 ページ\]](#)

指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### [WriteToServerAsync メソッド \[24 ページ\]](#)

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### [BatchSize プロパティ \[33 ページ\]](#)

各バッチの中のローの数を取得または設定します。

### [BulkCopyTimeout プロパティ \[33 ページ\]](#)

タイムアウトするまでに完了するオペレーションの秒数を取得または設定します。

### [ColumnMappings プロパティ \[34 ページ\]](#)

SABulkCopyColumnMapping 項目のコレクションを返します。

### [DestinationTableName プロパティ \[34 ページ\]](#)

データベースサーバ上の送信先テーブルの名前を取得または設定します。

### [NotifyAfter プロパティ \[35 ページ\]](#)

通知イベントが生成されるまでに処理されるローの数を取得または設定します。

### [SARowsCopied イベント \[36 ページ\]](#)

NotifyAfter プロパティで指定される数のローが処理されるたびに、このイベントが発生します。



## 1.1.1 SABulkCopy コンストラクタ

SABulkCopy オブジェクトを初期化します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SABulkCopy(SAConnection) [17 ページ]</a>	SABulkCopy オブジェクトを初期化します。
public	<a href="#">SABulkCopy(SAConnection, SABulkCopyOptions, SATransaction) [18 ページ]</a>	SABulkCopy オブジェクトを初期化します。
public	<a href="#">SABulkCopy(string) [19 ページ]</a>	SABulkCopy オブジェクトを初期化します。
public	<a href="#">SABulkCopy(string, SABulkCopyOptions) [19 ページ]</a>	SABulkCopy オブジェクトを初期化します。

このセクションの内容:

[SABulkCopy\(SAConnection\) コンストラクタ \[17 ページ\]](#)

SABulkCopy オブジェクトを初期化します。

[SABulkCopy\(SAConnection, SABulkCopyOptions, SATransaction\) コンストラクタ \[18 ページ\]](#)

SABulkCopy オブジェクトを初期化します。

[SABulkCopy\(string\) コンストラクタ \[19 ページ\]](#)

SABulkCopy オブジェクトを初期化します。

[SABulkCopy\(string, SABulkCopyOptions\) コンストラクタ \[19 ページ\]](#)

SABulkCopy オブジェクトを初期化します。

### 1.1.1.1 SABulkCopy(SAConnection) コンストラクタ

SABulkCopy オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopy (ByVal connection As SAConnection)
```

##### C#

```
public SABulkCopy (SAConnection connection)
```

## パラメータ

**connection** バルクコピーオペレーションの実行に使用する SAConnection オブジェクト。接続が開いていない場合は、WriteToServer に例外がスローされます。

### 1.1.1.2 SABulkCopy(SAConnection, SABulkCopyOptions, SATransaction) コンストラクタ

SABulkCopy オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopy (  
    ByVal connection As SAConnection,  
    ByVal copyOptions As SABulkCopyOptions,  
    ByVal externalTransaction As SATransaction  
)
```

##### C#

```
public SABulkCopy (  
    SAConnection connection,  
    SABulkCopyOptions copyOptions,  
    SATransaction externalTransaction  
)
```

## パラメータ

**connection** バルクコピーオペレーションの実行に使用する SAConnection オブジェクト。接続が開いていない場合は、WriteToServer に例外がスローされます。

**copyOptions** 目的のテーブルにコピーされるデータソースローを決定する、SABulkCopyOptions 列挙の値の組み合わせ。

**externalTransaction** バルクコピーが発生する、既存の SATransaction インスタンス。externalTransaction が NULL でない場合、バルクコピーオペレーションは其中で行われます。外部トランザクションと UseInternalTransaction オプションの両方を指定すると、エラーになります。

### 1.1.1.3 SABulkCopy(string) コンストラクタ

SABulkCopy オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopy (ByVal connectionString As String)
```

##### C#

```
public SABulkCopy (string connectionString)
```

#### パラメータ

**connectionString** SABulkCopy インスタンスによって使用されるために開かれる接続を定義する文字列。接続文字列は keyword=value のペアがセミコロンで区切られたリストです。

#### 備考

この構文は、connectionString を使用して WriteToServer の実行中に接続を開きます。WriteToServer が終了すると、接続が閉じます。

### 1.1.1.4 SABulkCopy(string, SABulkCopyOptions) コンストラクタ

SABulkCopy オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopy (  
    ByVal connectionString As String,  
    ByVal copyOptions As SABulkCopyOptions  
)
```

##### C#

```
public SABulkCopy (  
    string connectionString,  
    SABulkCopyOptions copyOptions  
)
```

## パラメータ

**connectionString** SABulkCopy インスタンスによって使用されるために開かれる接続を定義する文字列。接続文字列は keyword=value のペアがセミコロンで区切られたリストです。

**copyOptions** 目的のテーブルにコピーされるデータソースローを決定する、SABulkCopyOptions 列挙の値の組み合わせ。

## 備考

この構文は、connectionString を使用して WriteToServer の実行中に接続を開きます。WriteToServer が終了すると、接続が閉じます。copyOptions パラメータには、上記の効果がありません。

## 1.1.2 Close() メソッド

SABulkCopy インスタンスを閉じます。

### 構文

#### Visual Basic

```
Public Sub Close ()
```

#### C#

```
public void Close ()
```

## 1.1.3 Dispose() メソッド

SABulkCopy インスタンスを破棄します。

### 構文

#### Visual Basic

```
Public Sub Dispose ()
```

#### C#

```
public void Dispose ()
```

## 1.1.4 WriteToServer メソッド

指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public void	<a href="#">WriteToServer(DataRow[]) [22 ページ]</a>	指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public void	<a href="#">WriteToServer(DataTable) [22 ページ]</a>	指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public void	<a href="#">WriteToServer(DataTable, DataRowState) [23 ページ]</a>	指定されたローの状態を持つ指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public void	<a href="#">WriteToServer(IDataReader) [24 ページ]</a>	指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

このセクションの内容:

#### [WriteToServer\(DataRow\[\]\) メソッド \[22 ページ\]](#)

指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServer\(DataTable\) メソッド \[22 ページ\]](#)

指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServer\(DataTable, DataRowState\) メソッド \[23 ページ\]](#)

指定されたローの状態を持つ指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServer\(IDataReader\) メソッド \[24 ページ\]](#)

指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### 1.1.4.1 WriteToServer(DataRow[]) メソッド

指定された System.Data.DataRow オブジェクトの配列内のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Sub WriteToServer (ByVal rows As DataRow())
```

##### C#

```
public void WriteToServer (DataRow[] rows)
```

#### パラメータ

**rows** 送信先テーブルにコピーされる System.Data.DataRow オブジェクトの配列。

#### 関連情報

[DestinationTableName プロパティ \[34 ページ\]](#)

### 1.1.4.2 WriteToServer(DataTable) メソッド

指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Sub WriteToServer (ByVal table As DataTable)
```

##### C#

```
public void WriteToServer (DataTable table)
```

#### パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

## 関連情報

[DestinationTableName プロパティ \[34 ページ\]](#)

### 1.1.4.3 WriteToServer(DataTable, DataRowState) メソッド

指定されたローの状態を持つ指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Sub WriteToServer (  
    ByVal table As DataTable,  
    ByVal rowState As DataRowState  
)
```

##### C#

```
public void WriteToServer (  
    DataTable table,  
    DataRowState rowState  
)
```

## パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

**rowState** System.Data.DataRowState 列挙体の値。ローステータスに一致するローのみ、送信先にコピーされます。

## 備考

ローステータスに一致するローのみ、コピーされます。

## 関連情報

[DestinationTableName プロパティ \[34 ページ\]](#)

## 1.1.4.4 WriteToServer(IDataReader) メソッド

指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### 構文

#### Visual Basic

```
Public Sub WriteToServer (ByVal reader As IDataReader)
```

#### C#

```
public void WriteToServer (IDataReader reader)
```

### パラメータ

**reader** ローが送信先テーブルにコピーされる System.Data.IDataReader。

### 関連情報

[DestinationTableName プロパティ \[34 ページ\]](#)

## 1.1.5 WriteToServerAsync メソッド

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public Task	<a href="#">WriteToServerAsync(DataRow[]) [26 ページ]</a>	指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。



変更子とタイプ	オーバーロード名	説明
public Task	<a href="#">WriteToServerAsync(DataRow[], CancellationToken) [27 ページ]</a>	指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public Task	<a href="#">WriteToServerAsync(DataTable) [28 ページ]</a>	指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public Task	<a href="#">WriteToServerAsync(DataTable, CancellationToken) [28 ページ]</a>	指定された System.Data.DataTable のローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。
public Task	<a href="#">WriteToServerAsync(DataTable, DataRowState) [29 ページ]</a>	指定された System.Data.DataTable で指定ロー状態に一致するローのみを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public Task	<a href="#">WriteToServerAsync(DataTable, DataRowState, CancellationToken) [30 ページ]</a>	指定された System.Data.DataTable の指定ロー状態に一致するローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。
public Task	<a href="#">WriteToServerAsync(IDataReader) [31 ページ]</a>	指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。
public Task	<a href="#">WriteToServerAsync(IDataReader, CancellationToken) [32 ページ]</a>	指定された System.Data.IDataReader のすべてのローを SACLient.SqlBulkCopy オブジェクトの SACLient.SqlBulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

このセクションの内容:

[WriteToServerAsync\(DataRow\[\]\) メソッド \[26 ページ\]](#)

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServerAsync\(DataRow\[\], CancellationToken\) メソッド \[27 ページ\]](#)

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServerAsync\(DataTable\) メソッド \[28 ページ\]](#)

指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServerAsync\(DataTable, CancellationToken\) メソッド \[28 ページ\]](#)

指定された System.Data.DataTable のローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

#### [WriteToServerAsync\(DataTable, DataRowState\) メソッド \[29 ページ\]](#)

指定された System.Data.DataTable で指定ロー状態に一致するローのみを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServerAsync\(DataTable, DataRowState, CancellationToken\) メソッド \[30 ページ\]](#)

指定された System.Data.DataTable の指定ロー状態に一致するローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

#### [WriteToServerAsync\(IDataReader\) メソッド \[31 ページ\]](#)

指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### [WriteToServerAsync\(IDataReader, CancellationToken\) メソッド \[32 ページ\]](#)

指定された System.Data.IDataReader のすべてローを SAClient.SqlBulkCopy オブジェクトの SAClient.SqlBulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

## 1.1.5.1 WriteToServerAsync(DataRow[]) メソッド

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### 構文

#### Visual Basic

```
Public Function WriteToServerAsync (ByVal rows As DataRow()) As Task
```

#### C#

```
public Task WriteToServerAsync (DataRow[] rows)
```

## パラメータ

**rows** 送信先テーブルにコピーされる System.Data.DataRow オブジェクトの配列。

### 1.1.5.2 WriteToServerAsync(DataRow[], CancellationToken) メソッド

指定された System.Data.DataRow 配列のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Function WriteToServerAsync (  
    ByVal rows As DataRow(),  
    ByVal cancellationTok As CancellationToken  
) As Task
```

##### C#

```
public Task WriteToServerAsync (  
    DataRow[] rows,  
    CancellationToken cancellationToken  
)
```

## パラメータ

**rows** 送信先テーブルにコピーされる System.Data.DataRow オブジェクトの配列。

**cancellationTok** キャンセル手順。このパラメータの P:System.Threading.CancellationToken.None 値を使用すると、このメソッドは SABulkCopy.WriteToServerAsync(System.Data.DataTable) と同じになります。

## 戻り値

非同期操作を表すタスク。

## 例外

**SAException class** タスクオブジェクトで返されます。データのコピー中に発生したエラーが、データベースサーバによって返されます。

## 備考

キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。返された Task オブジェクトを介して例外がレポートされます。

### 1.1.5.3 WriteToServerAsync(DataTable) メソッド

指定された System.Data.DataTable のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Function WriteToServerAsync (ByVal table As DataTable) As Task
```

##### C#

```
public Task WriteToServerAsync (DataTable table)
```

## パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

### 1.1.5.4 WriteToServerAsync(DataTable, CancellationToken) メソッド

指定された System.Data.DataTable のローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

#### 構文

##### Visual Basic

```
Public Function WriteToServerAsync (  
    ByVal table As DataTable,  
    ByVal cancellationToken As CancellationToken  
) As Task
```

##### C#

```
public Task WriteToServerAsync (  
    DataTable table,  
    CancellationToken cancellationToken
```

)

## パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

**cancellationToken** キャンセル手順。このパラメータの P:System.Threading.CancellationToken.None 値を使用すると、このメソッドは SABulkCopy.WriteToServerAsync(System.Data.DataTable) と同じになります。

## 戻り値

非同期操作を表すタスク。

## 例外

**SAException** class タスクオブジェクトで返されます。データのコピー中に発生したエラーが、SQL サーバによって返されます。

## 備考

返された Task オブジェクトを介して例外がレポートされます。

## 1.1.5.5 WriteToServerAsync(DataTable, DataRowState) メソッド

指定された System.Data.DataTable で指定ロー状態に一致するローのみを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

### 構文

#### Visual Basic

```
Public Function WriteToServerAsync (  
    ByVal table As DataTable,  
    ByVal rowState As DataRowState  
) As Task
```

C#

```
public Task WriteToServerAsync (  
    DataTable table,  
    DataRowState rowState  
)
```

## パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

**rowState** System.Data.DataRowState 列挙体の値。ローステータスに一致するローのみ、送信先にコピーされます。

### 1.1.5.6 WriteToServerAsync(DataTable, DataRowState, CancellationToken) メソッド

指定された System.Data.DataTable の指定ロー状態に一致するローのみを SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

#### 構文

##### Visual Basic

```
Public Function WriteToServerAsync (  
    ByVal table As DataTable,  
    ByVal rowState As DataRowState,  
    ByVal cancellationTok As CancellationToken  
) As Task
```

##### C#

```
public Task WriteToServerAsync (  
    DataTable table,  
    DataRowState rowState,  
    CancellationToken cancellationToken  
)
```

## パラメータ

**table** ローが送信先テーブルにコピーされる System.Data.DataTable。

**rowState** System.Data.DataRowState 列挙体の値。ローステータスに一致するローのみ、送信先にコピーされます。

**cancellationTok** キャンセル手順。このパラメータの P:System.Threading.CancellationToken.None 値を使用すると、このメソッドは SABulkCopy.WriteToServerAsync(System.Data.DataTable, System.Data.DataRowState) と同じになります。

## 戻り値

非同期操作を表すタスク。

## 例外

**SAException class** タスクオブジェクトで返されます。データのコピー中に発生したエラーが、SQL サーバによって返されます。

## 備考

返された Task オブジェクトを介して例外がレポートされます。

### 1.1.5.7 WriteToServerAsync(IDataReader) メソッド

指定された System.Data.IDataReader のすべてのローを、SABulkCopy オブジェクトの SABulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。

#### 構文

##### Visual Basic

```
Public Function WriteToServerAsync (ByVal reader As IDataReader) As Task
```

##### C#

```
public Task WriteToServerAsync (IDataReader reader)
```

## パラメータ

**reader** ローが送信先テーブルにコピーされる System.Data.IDataReader。

## 1.1.5.8 WriteToServerAsync(IDataReader, CancellationTokentoken) メソッド

指定された System.Data.IDataReader のすべてのローを SACLient.SqlBulkCopy オブジェクトの SACLient.SqlBulkCopy.DestinationTableName プロパティで指定される送信先テーブルにコピーします。キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。

### 構文

#### Visual Basic

```
Public Function WriteToServerAsync (  
    ByVal reader As IDataReader,  
    ByVal cancellationTokentoken As CancellationTokentoken  
) As Task
```

#### C#

```
public Task WriteToServerAsync (  
    IDataReader reader,  
    CancellationTokentoken cancellationTokentoken  
)
```

### パラメータ

**reader** ローが送信先テーブルにコピーされる System.Data.IDataReader。

**cancellationTokentoken** キャンセル手順。このパラメータの P:System.Threading.CancellationTokentoken.None 値を使用すると、このメソッドは SACLient.SqlBulkCopy.WriteToServerAsync(System.Data.IDataReader) と同じになります。

### 戻り値

非同期操作を表すタスク。

### 例外

**SAException class** タスクオブジェクトで返されます。データのコピー中に発生したエラーが、SQL サーバによって返されます。

### 備考

返された Task オブジェクトを介して例外がレポートされます。



## 1.1.6 BatchSize プロパティ

各バッチの中のローの数を取得または設定します。

### 構文

#### Visual Basic

```
Public Property BatchSize As Integer
```

#### C#

```
public int BatchSize {get;set;}
```

### 備考

各バッチが終了すると、バッチ内のローがデータベースサーバに送信されます。

各バッチの中のロー数。デフォルトは 0 です。

このプロパティを 0 に設定すると、すべてのローが 1 つのバッチで送信されます。

このプロパティに 0 未満の値を設定すると、エラーになります。

バッチの進行中にこの値が変更された場合、現在のバッチはそのまま完了し、それ以降のバッチが新しい値を使用します。

## 1.1.7 BulkCopyTimeout プロパティ

タイムアウトするまでに完了するオペレーションの秒数を取得または設定します。

### 構文

#### Visual Basic

```
Public Property BulkCopyTimeout As Integer
```

#### C#

```
public int BulkCopyTimeout {get;set;}
```

### 備考

デフォルト値は 30 秒です。

値が 0 の場合、制限はありません。この場合、待機時間が無限になる可能性があるため、値は 0 にしないでください。

オペレーションがタイムアウトすると、現在のトランザクション内のすべてのローがロールバックされ、`SAException` が発生します。

このプロパティに 0 未満の値を設定すると、エラーになります。

## 1.1.8 ColumnMappings プロパティ

`SABulkCopyColumnMapping` 項目のコレクションを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property ColumnMappings As SABulkCopyColumnMappingCollection
```

#### C#

```
public SABulkCopyColumnMappingCollection ColumnMappings {get;}
```

### 備考

カラムマッピングは、データソース内のカラムと、送信先のカラムの間の関係を定義します。

デフォルトでは、空のコレクションです。

`WriteToServer` の実行中は、プロパティを変更できません。

`WriteToServer` の実行時に `ColumnMappings` が空の場合、ソース内の先頭のカラムが送信先の先頭のカラムにマッピングされ、2 番目は 2 番目にマッピングされます。以降についても同様です。この動作は、カラムの型が変換可能な場合、ソースカラム以上の送信先カラムがある場合、余分な送信先カラムが NULL 入力可のカラムである場合に行われます。

## 1.1.9 DestinationTableName プロパティ

データベースサーバ上の送信先テーブルの名前を取得または設定します。

### 構文

#### Visual Basic

```
Public Property DestinationTableName As String
```

#### C#

```
public string DestinationTableName {get;set;}
```

## 備考

デフォルト値は NULL 参照です。Visual Basic では、これは Nothing です。

WriteToServer の実行時に値が変更されても、変更は反映されません。

WriteToServer への呼び出しの前に値が設定されていない場合、InvalidOperationException が発生します。

値を NULL または空の文字列に設定すると、エラーになります。

## 1.1.10 NotifyAfter プロパティ

通知イベントが生成されるまでに処理されるローの数を取得または設定します。

### 構文

#### Visual Basic

```
Public Property NotifyAfter As Integer
```

#### C#

```
public int NotifyAfter {get;set;}
```

## 備考

プロパティが設定されていない場合は、0 が返されます。

WriteToServer の実行時に加えられる NotifyAfter への変更は、次の通知まで反映されません。

このプロパティに 0 未満の値を設定すると、エラーになります。

NotifyAfter と BulkCopyTimeout の値は相互に排他的であるため、データベースにローが送信されない場合や、コミットされない場合も、イベントは起動します。

## 関連情報

[BulkCopyTimeout プロパティ \[33 ページ\]](#)

## 1.1.1 SARowsCopied イベント

NotifyAfter プロパティで指定される数のローが処理されるたびに、このイベントが発生します。

### 構文

#### Visual Basic

```
Public Event SARowsCopied As SARowsCopiedEventHandler
```

#### C#

```
public SARowsCopiedEventHandler SARowsCopied;
```

### 備考

SARowsCopied イベントの受信は、ローがデータベースサーバに送信されたことやコミットされたことを意味するわけではありません。このイベントから Close メソッドを呼び出すことはできません。

### 関連情報

[NotifyAfter プロパティ \[35 ページ\]](#)

## 1.2 SABulkCopyColumnMapping クラス

SABulkCopy インスタンスのデータソース内のカラムと、インスタンスの送信先テーブル内のカラムの間のマッピングを定義します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SABulkCopyColumnMapping
```

#### C#

```
public sealed class SABulkCopyColumnMapping
```

## メンバー

SABulkCopyColumnMapping のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SABulkCopyColumnMapping [38 ページ]</a>	カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

### プロパティ

変更子とタイプ	プロパティ	説明
public string	<a href="#">DestinationColumn [41 ページ]</a>	マッピングされる送信先データベーステーブルのカラムの名前を取得または設定します。
public int	<a href="#">DestinationOrdinal [42 ページ]</a>	マッピングされる送信先テーブルにおけるカラムの順序を取得または設定します。
public string	<a href="#">SourceColumn [43 ページ]</a>	データソースにマッピングされるカラムの名前を取得または設定します。
public int	<a href="#">SourceOrdinal [44 ページ]</a>	データソース内のソースカラムの順序位置を取得または設定します。

このセクションの内容:

#### [SABulkCopyColumnMapping コンストラクタ \[38 ページ\]](#)

カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

#### [DestinationColumn プロパティ \[41 ページ\]](#)

マッピングされる送信先データベーステーブルのカラムの名前を取得または設定します。

#### [DestinationOrdinal プロパティ \[42 ページ\]](#)

マッピングされる送信先テーブルにおけるカラムの順序を取得または設定します。

#### [SourceColumn プロパティ \[43 ページ\]](#)

データソースにマッピングされるカラムの名前を取得または設定します。

#### [SourceOrdinal プロパティ \[44 ページ\]](#)

データソース内のソースカラムの順序位置を取得または設定します。

## 1.2.1 SABulkCopyColumnMapping コンストラクタ

カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SABulkCopyColumnMapping()</a> [39 ページ]	カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。
public	<a href="#">SABulkCopyColumnMapping(int, int)</a> [39 ページ]	カラムの順序を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。
public	<a href="#">SABulkCopyColumnMapping(int, string)</a> [40 ページ]	ソースカラムを参照するカラムの序数と、送信先カラムを参照するカラム名を使用して、新しいカラムマッピングを作成します。
public	<a href="#">SABulkCopyColumnMapping(string, int)</a> [40 ページ]	カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しいカラムマッピングを作成します。
public	<a href="#">SABulkCopyColumnMapping(string, string)</a> [41 ページ]	カラムの名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

このセクションの内容:

#### [SABulkCopyColumnMapping\(\) コンストラクタ \[39 ページ\]](#)

カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

#### [SABulkCopyColumnMapping\(int, int\) コンストラクタ \[39 ページ\]](#)

カラムの順序を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

#### [SABulkCopyColumnMapping\(int, string\) コンストラクタ \[40 ページ\]](#)

ソースカラムを参照するカラムの序数と、送信先カラムを参照するカラム名を使用して、新しいカラムマッピングを作成します。

#### [SABulkCopyColumnMapping\(string, int\) コンストラクタ \[40 ページ\]](#)

カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しいカラムマッピングを作成します。

#### [SABulkCopyColumnMapping\(string, string\) コンストラクタ \[41 ページ\]](#)

カラムの名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

## 1.2.1.1 SABulkCopyColumnMapping() コンストラクタ

カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

### 構文

#### Visual Basic

```
Public Sub SABulkCopyColumnMapping ()
```

#### C#

```
public SABulkCopyColumnMapping ()
```

## 1.2.1.2 SABulkCopyColumnMapping(int, int) コンストラクタ

カラムの順序を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

### 構文

#### Visual Basic

```
Public Sub SABulkCopyColumnMapping (  
    ByVal sourceColumnOrdinal As Integer,  
    ByVal destinationColumnOrdinal As Integer  
)
```

#### C#

```
public SABulkCopyColumnMapping (  
    int sourceColumnOrdinal,  
    int destinationColumnOrdinal  
)
```

## パラメータ

**sourceColumnOrdinal** データソース内のソースカラムの序数位置。データソースの最初のカラムの序数位置は 0 です。

**destinationColumnOrdinal** 送信先テーブル内の送信先カラムの序数位置。テーブルの最初のカラムの序数位置は 0 です。

### 1.2.1.3 SABulkCopyColumnMapping(int, string) コンストラクタ

ソースカラムを参照するカラムの序数と、送信先カラムを参照するカラム名を使用して、新しいカラムマッピングを作成します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopyColumnMapping (  
    ByVal sourceColumnOrdinal As Integer,  
    ByVal destinationColumn As String  
)
```

##### C#

```
public SABulkCopyColumnMapping (  
    int sourceColumnOrdinal,  
    string destinationColumn  
)
```

#### パラメータ

**sourceColumnOrdinal** データソース内のソースカラムの序数位置。データソースの最初のカラムの序数位置は 0 です。

**destinationColumn** 送信先テーブル内の送信先カラムの名前。

### 1.2.1.4 SABulkCopyColumnMapping(string, int) コンストラクタ

カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しいカラムマッピングを作成します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopyColumnMapping (  
    ByVal sourceColumn As String,  
    ByVal destinationColumnOrdinal As Integer  
)
```

##### C#

```
public SABulkCopyColumnMapping (  
    string sourceColumn,  
    int destinationColumnOrdinal  
)
```



## パラメータ

**sourceColumn** データソース内のソースカラムの名前。

**destinationColumnOrdinal** 送信先テーブル内の送信先カラムの序数位置。テーブルの最初のカラムの序数位置は 0 です。

### 1.2.1.5 SABulkCopyColumnMapping(string, string) コンストラクタ

カラムの名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

#### 構文

##### Visual Basic

```
Public Sub SABulkCopyColumnMapping (  
    ByVal sourceColumn As String,  
    ByVal destinationColumn As String  
)
```

##### C#

```
public SABulkCopyColumnMapping (  
    string sourceColumn,  
    string destinationColumn  
)
```

## パラメータ

**sourceColumn** データソース内のソースカラムの名前。

**destinationColumn** 送信先テーブル内の送信先カラムの名前。

### 1.2.2 DestinationColumn プロパティ

マッピングされる送信先データベーステーブルのカラムの名前を取得または設定します。

#### 構文

##### Visual Basic

```
Public Property DestinationColumn As String
```

##### C#

```
public string DestinationColumn {get;set;}
```

## 備考

送信先テーブルのカラムの名前を指定する文字列。DestinationOrdinal プロパティに優先度がない場合は NULL 参照 (Visual Basic の Nothing)。

DestinationColumn プロパティと DestinationOrdinal プロパティは、相互に排他的です。直前に設定された値が優先されます。

DestinationColumn プロパティを設定すると、DestinationOrdinal プロパティは -1 に設定されます。DestinationOrdinal プロパティを設定すると、DestinationColumn プロパティは NULL 参照 (Visual Basic の Nothing) に設定されます。

DestinationColumn を NULL または空の文字列に設定すると、エラーになります。

## 関連情報

[DestinationOrdinal プロパティ \[42 ページ\]](#)

### 1.2.3 DestinationOrdinal プロパティ

マッピングされる送信先テーブルにおけるカラムの順序を取得または設定します。

#### 構文

##### Visual Basic

```
Public Property DestinationOrdinal As Integer
```

##### C#

```
public int DestinationOrdinal {get;set;}
```

## 備考

マッピングされるカラムの送信先テーブルにおける順序を指定する整数。プロパティが設定されていない場合は -1。

DestinationColumn プロパティと DestinationOrdinal プロパティは、相互に排他的です。直前に設定された値が優先されます。

DestinationColumn プロパティを設定すると、DestinationOrdinal プロパティは -1 に設定されます。DestinationOrdinal プロパティを設定すると、DestinationColumn プロパティは null 参照 (Visual Basic の Nothing) に設定されます。

## 関連情報

[DestinationColumn プロパティ \[41 ページ\]](#)

### 1.2.4 SourceColumn プロパティ

データソースにマッピングされるカラムの名前を取得または設定します。

#### 構文

##### Visual Basic

```
Public Property SourceColumn As String
```

##### C#

```
public string SourceColumn {get;set;}
```

## 備考

データソースのカラムの名前を指定する文字列。SourceOrdinal プロパティに優先度がない場合は NULL 参照 (Visual Basic の Nothing)。

SourceColumn プロパティと SourceOrdinal プロパティは、相互に排他的です。直前に設定された値が優先されます。

SourceColumn プロパティを設定すると、SourceOrdinal プロパティは -1 に設定されます。SourceOrdinal プロパティを設定すると、SourceColumn プロパティは null 参照 (Visual Basic の Nothing) に設定されます。

SourceColumn を NULL または空の文字列に設定すると、エラーになります。

## 関連情報

[SourceOrdinal プロパティ \[44 ページ\]](#)

## 1.2.5 SourceOrdinal プロパティ

データソース内のソースカラムの順序位置を取得または設定します。

### 構文

#### Visual Basic

```
Public Property SourceOrdinal As Integer
```

#### C#

```
public int SourceOrdinal {get;set;}
```

### 備考

データソースのカラムの順序を指定する整数。プロパティが設定されていない場合は -1。

SourceColumn プロパティと SourceOrdinal プロパティは、相互に排他的です。直前に設定された値が優先されます。

SourceColumn プロパティを設定すると、SourceOrdinal プロパティは -1 に設定されます。SourceOrdinal プロパティを設定すると、SourceColumn プロパティは null 参照 (Visual Basic の Nothing) に設定されます。

### 関連情報

[SourceColumn プロパティ \[43 ページ\]](#)

## 1.3 SABulkCopyColumnMappingCollection クラス

System.Collections.CollectionBase から継承した SABulkCopyColumnMapping オブジェクトのコレクションです。

### 構文

#### Visual Basic

```
Public NotInheritable Class SABulkCopyColumnMappingCollection Inherits  
System.Collections.CollectionBase
```

#### C#

```
public sealed class SABulkCopyColumnMappingCollection :  
System.Collections.CollectionBase
```

## メンバー

SABulkCopyColumnMappingCollection のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変数とタイプ	メソッド	説明
public SABulkCopyColumnMapping	<a href="#">Add [46 ページ]</a>	指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。
public bool	<a href="#">Contains(SABulkCopyColumnMapping) [50 ページ]</a>	指定された SABulkCopyColumnMapping オブジェクトがコレクション内にあるかどうかを示す値を取得します。
public void	<a href="#">CopyTo(SABulkCopyColumnMapping[], int) [51 ページ]</a>	SABulkCopyColumnMappingCollection の要素を、特定のインデックスを先頭に、SABulkCopyColumnMapping 項目の配列にコピーします。
public int	<a href="#">IndexOf(SABulkCopyColumnMapping) [51 ページ]</a>	コレクション内の指定された SABulkCopyColumnMapping オブジェクトのインデックスを取得または設定します。
public void	<a href="#">Remove(SABulkCopyColumnMapping) [52 ページ]</a>	指定された SABulkCopyColumnMapping 要素を SABulkCopyColumnMappingCollection から削除します。
public new void	<a href="#">RemoveAt(int) [52 ページ]</a>	指定されたインデックス位置のマッピングをコレクションから削除します。

### プロパティ

変数とタイプ	プロパティ	説明
public SABulkCopyColumnMapping	<a href="#">this[int index] [53 ページ]</a>	指定されたインデックス位置の SABulkCopyColumnMapping オブジェクトを取得します。

## 備考

**実装:** ICollection、IEnumerable、IList

このセクションの内容:

#### [Add メソッド \[46 ページ\]](#)

指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。

#### [Contains\(SABulkCopyColumnMapping\) メソッド \[50 ページ\]](#)

指定された SABulkCopyColumnMapping オブジェクトがコレクション内にあるかどうかを示す値を取得します。

#### [CopyTo\(SABulkCopyColumnMapping\[\], int\) メソッド \[51 ページ\]](#)

SABulkCopyColumnMappingCollection の要素を、特定のインデックスを先頭に、SABulkCopyColumnMapping 項目の配列にコピーします。

### [IndexOf\(SABulkCopyColumnMapping\) メソッド \[51 ページ\]](#)

コレクション内の指定された SABulkCopyColumnMapping オブジェクトのインデックスを取得または設定します。

### [Remove\(SABulkCopyColumnMapping\) メソッド \[52 ページ\]](#)

指定された SABulkCopyColumnMapping 要素を SABulkCopyColumnMappingCollection から削除します。

### [RemoveAt\(int\) メソッド \[52 ページ\]](#)

指定されたインデックス位置のマッピングをコレクションから削除します。

### [this\[int index\] プロパティ \[53 ページ\]](#)

指定されたインデックス位置の SABulkCopyColumnMapping オブジェクトを取得します。

## 1.3.1 Add メソッド

指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。

### オーバードリスト

変数とタイプ	オーバーロード名	説明
public SABulkCopyColumnMapping	<a href="#">Add(SABulkCopyColumnMapping) [47 ページ]</a>	指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。
public SABulkCopyColumnMapping	<a href="#">Add(int, int) [48 ページ]</a>	ソースカラムと送信先カラムの両方を指定する順序を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。
public SABulkCopyColumnMapping	<a href="#">Add(int, string) [48 ページ]</a>	カラムの順序を使用してソースカラムを参照し、カラム名を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。
public SABulkCopyColumnMapping	<a href="#">Add(string, int) [49 ページ]</a>	カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。
public SABulkCopyColumnMapping	<a href="#">Add(string, string) [50 ページ]</a>	ソースカラムと送信先カラムの両方を指定するカラム名を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

このセクションの内容:

#### [Add\(SABulkCopyColumnMapping\) メソッド \[47 ページ\]](#)

指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。

#### [Add\(int, int\) メソッド \[48 ページ\]](#)

ソースカラムと送信先カラムの両方を指定する順序を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### [Add\(int, string\) メソッド \[48 ページ\]](#)

カラムの順序を使用してソースカラムを参照し、カラム名を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### [Add\(string, int\) メソッド \[49 ページ\]](#)

カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### [Add\(string, string\) メソッド \[50 ページ\]](#)

ソースカラムと送信先カラムの両方を指定するカラム名を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

### 1.3.1.1 Add(SABulkCopyColumnMapping) メソッド

指定された SABulkCopyColumnMapping オブジェクトをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (ByVal bulkCopyColumnMapping As SABulkCopyColumnMapping)
    As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping Add (SABulkCopyColumnMapping
    bulkCopyColumnMapping)
```

#### パラメータ

**bulkCopyColumnMapping** コレクションに追加される、マッピングを記述する SABulkCopyColumnMapping オブジェクト。

#### 関連情報

[SABulkCopyColumnMapping クラス \[36 ページ\]](#)

### 1.3.1.2 Add(int, int) メソッド

ソースカラムと送信先カラムの両方を指定する順序を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (
    ByVal sourceColumnOrdinal As Integer,
    ByVal destinationColumnOrdinal As Integer
) As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping Add (
    int sourceColumnOrdinal,
    int destinationColumnOrdinal
)
```

#### パラメータ

**sourceColumnOrdinal** データソース内のソースカラムの序数位置。

**destinationColumnOrdinal** 送信先テーブル内の送信先カラムの序数位置。

### 1.3.1.3 Add(int, string) メソッド

カラムの順序を使用してソースカラムを参照し、カラム名を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (
    ByVal sourceColumnOrdinal As Integer,
    ByVal destinationColumn As String
) As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping Add (
    int sourceColumnOrdinal,
    string destinationColumn
)
```



## パラメータ

**sourceColumnOrdinal** データソース内のソースカラムの序数位置。

**destinationColumn** 送信先テーブル内の送信先カラムの名前。

### 1.3.1.4 Add(string, int) メソッド

カラム名を使用してソースカラムを参照し、カラムの順序を使用して送信先カラムを参照する、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (  
    ByVal sourceColumn As String,  
    ByVal destinationColumnOrdinal As Integer  
) As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping Add (  
    string sourceColumn,  
    int destinationColumnOrdinal  
)
```

## パラメータ

**sourceColumn** データソース内のソースカラムの名前。

**destinationColumnOrdinal** 送信先テーブル内の送信先カラムの序数位置。

## 備考

カラムの順序または名前を使用してソースカラムと送信先カラムを参照する、新しいカラムマッピングを作成します。

### 1.3.1.5 Add(string, string) メソッド

ソースカラムと送信先カラムの両方を指定するカラム名を使用して、新しい SABulkCopyColumnMapping オブジェクトを作成し、このマッピングをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (
    ByVal sourceColumn As String,
    ByVal destinationColumn As String
) As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping Add (
    string sourceColumn,
    string destinationColumn
)
```

#### パラメータ

**sourceColumn** データソース内のソースカラムの名前。

**destinationColumn** 送信先テーブル内の送信先カラムの名前。

### 1.3.2 Contains(SABulkCopyColumnMapping) メソッド

指定された SABulkCopyColumnMapping オブジェクトがコレクション内にあるかどうかを示す値を取得します。

#### 構文

##### Visual Basic

```
Public Function Contains (ByVal value As SABulkCopyColumnMapping) As Boolean
```

##### C#

```
public bool Contains (SABulkCopyColumnMapping value)
```

#### パラメータ

**value** 有効な SABulkCopyColumnMapping オブジェクト。

戻り値

指定のマッピングがコレクション内にある場合は true、ない場合は false。

### 1.3.3 CopyTo(SABulkCopyColumnMapping[], int) メソッド

SABulkCopyColumnMappingCollection の要素を、特定のインデックスを先頭に、SABulkCopyColumnMapping 項目の配列にコピーします。

#### 構文

##### Visual Basic

```
Public Sub CopyTo (  
    ByVal array As SABulkCopyColumnMapping(),  
    ByVal index As Integer  
)
```

##### C#

```
public void CopyTo (  
    SABulkCopyColumnMapping[] array,  
    int index  
)
```

#### パラメータ

**array** SABulkCopyColumnMappingCollection の要素のコピー先である、1 次元の SABulkCopyColumnMapping 配列。配列のインデックスは 0 から始まります。

**index** コピーが開始される、配列内の 0 から始まるインデックス。

### 1.3.4 IndexOf(SABulkCopyColumnMapping) メソッド

コレクション内の指定された SABulkCopyColumnMapping オブジェクトのインデックスを取得または設定します。

#### 構文

##### Visual Basic

```
Public Function IndexOf (ByVal value As SABulkCopyColumnMapping) As Integer
```

##### C#

```
public int IndexOf (SABulkCopyColumnMapping value)
```

## パラメータ

**value** 検索対象の SABulkCopyColumnMapping オブジェクト。

## 戻り値

カラムマッピングの 0 から始まるインデックス、またはコレクション内にカラムマッピングが見つからない場合は -1 が返されます。

### 1.3.5 Remove(SABulkCopyColumnMapping) メソッド

指定された SABulkCopyColumnMapping 要素を SABulkCopyColumnMappingCollection から削除します。

#### 構文

##### Visual Basic

```
Public Sub Remove (ByVal value As SABulkCopyColumnMapping)
```

##### C#

```
public void Remove (SABulkCopyColumnMapping value)
```

## パラメータ

**value** コレクションから削除する SABulkCopyColumnMapping オブジェクト。

### 1.3.6 RemoveAt(int) メソッド

指定されたインデックス位置のマッピングをコレクションから削除します。

#### 構文

##### Visual Basic

```
Public Shadows Sub RemoveAt (ByVal index As Integer)
```

##### C#

```
public new void RemoveAt (int index)
```

## パラメータ

`index` コレクションから削除する `SABulkCopyColumnMapping` オブジェクトの、0 から始まるインデックス。

### 1.3.7 `this[int index]` プロパティ

指定されたインデックス位置の `SABulkCopyColumnMapping` オブジェクトを取得します。

#### 構文

##### Visual Basic

```
Public ReadOnly Property Item (ByVal index As Integer) As SABulkCopyColumnMapping
```

##### C#

```
public SABulkCopyColumnMapping this[int index] {get;}
```

## 戻り値

`SABulkCopyColumnMapping` オブジェクトが返されます。

## 1.4 SACommand クラス

データベースに対して実行される SQL 文またはストアプロシージャです。

#### 構文

##### Visual Basic

```
Public NotInheritable Class SACommand Inherits System.Data.Common.DbCommand  
Implements System.ICloneable
```

##### C#

```
public sealed class SACommand : System.Data.Common.DbCommand, System.ICloneable
```

## メンバー

`SACommand` のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

## コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SACommand [57 ページ]</a>	SACommand オブジェクトを初期化します。

## メソッド

変更子とタイプ	メソッド	説明
public IAsyncResult	<a href="#">BeginExecuteNonQuery [60 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。
public IAsyncResult	<a href="#">BeginExecuteReader [63 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから 1 つまたは複数の結果セットを取得します。
public override void	<a href="#">Cancel() [68 ページ]</a>	SACommand オブジェクトの実行をキャンセルします。
protected override DbParameter	<a href="#">CreateDbParameter() [69 ページ]</a>	System.Data.Common.DbParameter オブジェクトの新しいインスタンスを作成します。
public new SqlParameter	<a href="#">CreateParameter() [69 ページ]</a>	SACommand オブジェクトにパラメータを指定するために SqlParameter オブジェクトを提供します。
protected override void	<a href="#">Dispose(bool) [70 ページ]</a>	オブジェクトに関連付けられているリソースを解放します。
public unsafe int	<a href="#">EndExecuteNonQuery(IAsyncResult) [70 ページ]</a>	SQL 文またはストアプロシージャの非同期実行を終了します。
public unsafe SqlDataReader	<a href="#">EndExecuteReader(IAsyncResult) [73 ページ]</a>	SQL 文またはストアプロシージャの非同期実行を終了し、要求された SqlDataReader を返します。
protected override DbDataReader	<a href="#">ExecuteDbDataReader(CommandBehavior) [75 ページ]</a>	接続に対してコマンドテキストを実行します。
public override unsafe int	<a href="#">ExecuteNonQuery() [75 ページ]</a>	結果セットを返さない文 (INSERT、UPDATE、DELETE など) や、データ定義文を実行します。
public new SqlDataReader	<a href="#">ExecuteReader [76 ページ]</a>	結果セットを返す SQL 文を実行します。
public new Task< SqlDataReader >	<a href="#">ExecuteReaderAsync [79 ページ]</a>	SACommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。
public override object	<a href="#">ExecuteScalar() [83 ページ]</a>	単一の値を返す文を実行します。
public override void	<a href="#">Prepare() [84 ページ]</a>	データソース上で SACommand を準備またはコンパイルします。
public void	<a href="#">ResetCommandTimeout() [85 ページ]</a>	CommandTimeout プロパティをデフォルト値の 30 秒にリセットします。

## プロパティ

変数とタイプ	プロパティ	説明
public override string	<a href="#">CommandText [85 ページ]</a>	SQL 文またはストアドプロシージャのテキストを取得または設定します。
public override int	<a href="#">CommandTimeout [86 ページ]</a>	この機能は .NET データプロバイダでサポートされていません。
public override CommandType	<a href="#">CommandType [86 ページ]</a>	SACommand によって示されるコマンドのタイプを取得または設定します。
public new SAConnection	<a href="#">Connection [87 ページ]</a>	SACommand オブジェクトが適用される接続オブジェクトを取得または設定します。
protected override DbConnection	<a href="#">DbConnection [87 ページ]</a>	この SACommand オブジェクトによって使用される System.Data.Common.DbConnection を取得または設定します。
protected override DbParameterCollection	<a href="#">DbParameterCollection [88 ページ]</a>	System.Data.Common.DbParameter オブジェクトのコレクションを取得します。
protected override DbTransaction	<a href="#">DbTransaction [88 ページ]</a>	この SACommand オブジェクトが実行される System.Data.Common.DbTransaction を取得または設定します。
public override bool	<a href="#">DesignTimeVisible [89 ページ]</a>	Windows Form Designer 制御で SACommand を参照できるようにするかどうかを指定する値を取得または設定します。
public new SAParameterCollection	<a href="#">Parameters [89 ページ]</a>	現在の文のパラメータのコレクションを指定します。
public new SATransaction	<a href="#">Transaction [90 ページ]</a>	SACommand が実行される SATransaction オブジェクトを指定します。
public override UpdateRowSource	<a href="#">UpdatedRowSource [91 ページ]</a>	SADaAdapter の Update メソッドによって使用されるときにコマンドの結果が DataRow に適用される方法を取得または設定します。

## 備考

**実装:** IDbCommand、ICloneable

詳細については、「データのアクセスとデータの操作」を参照してください。

このセクションの内容:

[SACommand コンストラクタ \[57 ページ\]](#)

SACommand オブジェクトを初期化します。

[BeginExecuteNonQuery メソッド \[60 ページ\]](#)

この SACommand で記述される SQL 文またはストアドプロシージャの非同期実行を開始します。

[BeginExecuteReader メソッド \[63 ページ\]](#)

この SACommand で記述される SQL 文またはストアードプロシージャの非同期実行を開始し、データベースサーバから 1 つまたは複数の結果セットを取得します。

#### [Cancel\(\) メソッド \[68 ページ\]](#)

SACommand オブジェクトの実行をキャンセルします。

#### [CreateDbParameter\(\) メソッド \[69 ページ\]](#)

System.Data.Common.DbParameter オブジェクトの新しいインスタンスを作成します。

#### [CreateParameter\(\) メソッド \[69 ページ\]](#)

SACommand オブジェクトにパラメータを指定するために SAParameter オブジェクトを提供します。

#### [Dispose\(bool\) メソッド \[70 ページ\]](#)

オブジェクトに関連付けられているリソースを解放します。

#### [EndExecuteNonQuery\(IAsyncResult\) メソッド \[70 ページ\]](#)

SQL 文またはストアードプロシージャの非同期実行を終了します。

#### [EndExecuteReader\(IAsyncResult\) メソッド \[73 ページ\]](#)

SQL 文またはストアードプロシージャの非同期実行を終了し、要求された SADATAReader を返します。

#### [ExecuteDbDataReader\(CommandBehavior\) メソッド \[75 ページ\]](#)

接続に対してコマンドテキストを実行します。

#### [ExecuteNonQuery\(\) メソッド \[75 ページ\]](#)

結果セットを返さない文 (INSERT、UPDATE、DELETE など) や、データ定義文を実行します。

#### [ExecuteReader メソッド \[76 ページ\]](#)

結果セットを返す SQL 文を実行します。

#### [ExecuteReaderAsync メソッド \[79 ページ\]](#)

SACommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### [ExecuteScalar\(\) メソッド \[83 ページ\]](#)

単一の値を返す文を実行します。

#### [Prepare\(\) メソッド \[84 ページ\]](#)

データソース上で SACommand を準備またはコンパイルします。

#### [ResetCommandTimeout\(\) メソッド \[85 ページ\]](#)

CommandTimeout プロパティをデフォルト値の 30 秒にリセットします。

#### [CommandText プロパティ \[85 ページ\]](#)

SQL 文またはストアードプロシージャのテキストを取得または設定します。

#### [CommandTimeout プロパティ \[86 ページ\]](#)

この機能は .NET データプロバイダでサポートされていません。

#### [CommandType プロパティ \[86 ページ\]](#)

SACommand によって示されるコマンドのタイプを取得または設定します。

#### [Connection プロパティ \[87 ページ\]](#)

SACommand オブジェクトが適用される接続オブジェクトを取得または設定します。

#### [DbConnection プロパティ \[87 ページ\]](#)

この SACommand オブジェクトによって使用される System.Data.Common.DbConnection を取得または設定します。



#### [DbParameterCollection プロパティ \[88 ページ\]](#)

System.Data.Common.DbParameter オブジェクトのコレクションを取得します。

#### [DbTransaction プロパティ \[88 ページ\]](#)

この SACommand オブジェクトが実行される System.Data.Common.DbTransaction を取得または設定します。

#### [DesignTimeVisible プロパティ \[89 ページ\]](#)

Windows Form Designer 制御で SACommand を参照できるようにするかどうかを指定する値を取得または設定します。

#### [Parameters プロパティ \[89 ページ\]](#)

現在の文のパラメータのコレクションを指定します。

#### [Transaction プロパティ \[90 ページ\]](#)

SACommand が実行される SATransaction オブジェクトを指定します。

#### [UpdatedRowSource プロパティ \[91 ページ\]](#)

SDataAdapter の Update メソッドによって使用されるときにコマンドの結果が DataRow に適用される方法を取得または設定します。

## 1.4.1 SACommand コンストラクタ

SACommand オブジェクトを初期化します。

### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public	<a href="#">SACommand() [58 ページ]</a>	SACommand オブジェクトを初期化します。
public	<a href="#">SACommand(string) [58 ページ]</a>	SACommand オブジェクトを初期化します。
public	<a href="#">SACommand(string, SAConnection) [59 ページ]</a>	データベースに対して実行される SQL 文またはストアプロシージャです。
public	<a href="#">SACommand(string, SAConnection, SATransaction) [59 ページ]</a>	データベースに対して実行される SQL 文またはストアプロシージャです。

このセクションの内容:

#### [SACommand\(\) コンストラクタ \[58 ページ\]](#)

SACommand オブジェクトを初期化します。

#### [SACommand\(string\) コンストラクタ \[58 ページ\]](#)

SACommand オブジェクトを初期化します。

#### [SACommand\(string, SAConnection\) コンストラクタ \[59 ページ\]](#)

データベースに対して実行される SQL 文またはストアプロシージャです。

#### [SACommand\(string, SAConnection, SATransaction\) コンストラクタ \[59 ページ\]](#)

データベースに対して実行される SQL 文またはストアドプロシージャです。

### 1.4.1.1 SACommand() コンストラクタ

SACommand オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SACommand ()
```

##### C#

```
public SACommand ()
```

### 1.4.1.2 SACommand(string) コンストラクタ

SACommand オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SACommand (ByVal cmdText As String)
```

##### C#

```
public SACommand (string cmdText)
```

## パラメータ

**cmdText** SQL 文またはストアドプロシージャのテキスト。パラメータ化された文の場合、疑問符 (?) プレースホルダを使用してパラメータを渡します。

### 1.4.1.3 SACommand(string, SAConnection) コンストラクタ

データベースに対して実行される SQL 文またはストアプロシージャです。

#### 構文

##### Visual Basic

```
Public Sub SACommand (  
    ByVal cmdText As String,  
    ByVal connection As SAConnection  
)
```

##### C#

```
public SACommand (  
    string cmdText,  
    SAConnection connection  
)
```

#### パラメータ

**cmdText** SQL 文またはストアプロシージャのテキスト。パラメータ化された文の場合、疑問符 (?) プレースホルダを使用してパラメータを渡します。

**connection** 現在の接続。

### 1.4.1.4 SACommand(string, SAConnection, SATransaction) コンストラクタ

データベースに対して実行される SQL 文またはストアプロシージャです。

#### 構文

##### Visual Basic

```
Public Sub SACommand (  
    ByVal cmdText As String,  
    ByVal connection As SAConnection,  
    ByVal transaction As SATransaction  
)
```

##### C#

```
public SACommand (  
    string cmdText,  
    SAConnection connection,  
    SATransaction transaction  
)
```

## パラメータ

**cmdText** SQL 文またはストアプロシージャのテキスト。パラメータ化された文の場合、疑問符 (?) プレースホルダを使用してパラメータを渡します。

**connection** 現在の接続。

**transaction** SACConnection が実行される SATransaction オブジェクト。

## 関連情報

[SATransaction クラス \[337 ページ\]](#)

## 1.4.2 BeginExecuteNonQuery メソッド

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public IAsyncResult	<a href="#">BeginExecuteNonQuery() [61 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。
public IAsyncResult	<a href="#">BeginExecuteNonQuery(AsyncCallback, object) [62 ページ]</a>	コールバックプロシージャとステータス情報を指定し、この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。

このセクションの内容:

#### [BeginExecuteNonQuery\(\) メソッド \[61 ページ\]](#)

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。

#### [BeginExecuteNonQuery\(AsyncCallback, object\) メソッド \[62 ページ\]](#)

コールバックプロシージャとステータス情報を指定し、この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。

## 1.4.2.1 BeginExecuteNonQuery() メソッド

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始します。

### 構文

#### Visual Basic

```
Public Function BeginExecuteNonQuery () As IAsyncResult
```

#### C#

```
public IAsyncResult BeginExecuteNonQuery ()
```

### 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、影響を受けたローの数を返す EndExecuteNonQuery(IAsyncResult) を起動する場合にも必要です。

### 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

### 備考

非同期コマンドでは、パラメータの順序は CommandText と一致している必要があります。

### 関連情報

[EndExecuteNonQuery\(IAsyncResult\) メソッド \[70 ページ\]](#)

## 1.4.2.2 BeginExecuteNonQuery(AsyncCallback, object) メソッド

コールバックプロシージャとステータス情報を指定し、この SACommand で記述される SQL 文またはストアードプロシージャの非同期実行を開始します。

### 構文

#### Visual Basic

```
Public Function BeginExecuteNonQuery (  
    ByVal callback As AsyncCallback,  
    ByVal stateObject As Object  
) As IAsyncResult
```

#### C#

```
public IAsyncResult BeginExecuteNonQuery (  
    AsyncCallback callback,  
    object stateObject  
)
```

### パラメータ

**callback** コマンドの実行が終了すると起動される System.AsyncCallback デリゲート。コールバックが必要ないことを示すには、NULL (Microsoft Visual Basic の場合は Nothing) を渡します。

**stateObject** コールバックプロシージャに渡される、ユーザ定義のステータスオブジェクト。コールバックプロシージャからこのオブジェクトを取得するには、System.IAsyncResult.AsyncState プロパティを使用します。

### 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、影響を受けたローの数を返す EndExecuteNonQuery(IAsyncResult) を起動する場合にも必要です。

### 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

### 備考

非同期コマンドでは、パラメータの順序は CommandText と一致している必要があります。

## 関連情報

[EndExecuteNonQuery\(IAsyncResult\) メソッド \[70 ページ\]](#)

### 1.4.3 BeginExecuteReader メソッド

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

#### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public IAsyncResult	<a href="#">BeginExecuteReader() [64 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。
public IAsyncResult	<a href="#">BeginExecuteReader(AsyncCallback, object) [65 ページ]</a>	コールバックプロシージャとステータス情報を指定し、この SACommand オブジェクトで記述される SQL 文の非同期実行を開始し、結果セットを取得します。
public IAsyncResult	<a href="#">BeginExecuteReader(AsyncCallback, object, CommandBehavior) [66 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。
public IAsyncResult	<a href="#">BeginExecuteReader(CommandBehavior) [67 ページ]</a>	この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

このセクションの内容:

#### [BeginExecuteReader\(\) メソッド \[64 ページ\]](#)

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

#### [BeginExecuteReader\(AsyncCallback, object\) メソッド \[65 ページ\]](#)

コールバックプロシージャとステータス情報を指定し、この SACommand オブジェクトで記述される SQL 文の非同期実行を開始し、結果セットを取得します。

#### [BeginExecuteReader\(AsyncCallback, object, CommandBehavior\) メソッド \[66 ページ\]](#)

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

#### [BeginExecuteReader\(CommandBehavior\) メソッド \[67 ページ\]](#)

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

### 1.4.3.1 BeginExecuteReader() メソッド

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

#### 構文

##### Visual Basic

```
Public Function BeginExecuteReader () As IAsyncResult
```

##### C#

```
public IAsyncResult BeginExecuteReader ()
```

#### 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、返されたローを取得するために使用する SADATAReader オブジェクトを返す、EndExecuteReader(IAsyncResult) を起動する場合にも必要です。

#### 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

#### 備考

非同期コマンドでは、パラメータの順序は CommandText と一致している必要があります。

#### 関連情報

[EndExecuteReader\(IAsyncResult\) メソッド \[73 ページ\]](#)

[SADATAReader クラス \[206 ページ\]](#)



## 1.4.3.2 BeginExecuteReader(AsyncCallback, object) メソッド

コールバックプロシージャとステータス情報を指定し、この SACommand オブジェクトで記述される SQL 文の非同期実行を開始し、結果セットを取得します。

### 構文

#### Visual Basic

```
Public Function BeginExecuteReader (  
    ByVal callback As AsyncCallback,  
    ByVal stateObject As Object  
) As IAsyncResult
```

#### C#

```
public IAsyncResult BeginExecuteReader (  
    AsyncCallback callback,  
    object stateObject  
)
```

### パラメータ

**callback** コマンドの実行が終了すると起動される System.AsyncCallback デリゲート。コールバックが必要ないことを示すには、NULL (Microsoft Visual Basic の場合は Nothing) を渡します。

**stateObject** コールバックプロシージャに渡される、ユーザ定義のステータスオブジェクト。コールバックプロシージャからこのオブジェクトを取得するには、System.IAsyncResult.AsyncState プロパティを使用します。

### 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、返されたローを取得するために使用する SADATAReader オブジェクトを返す、EndExecuteReader(IAsyncResult) を起動する場合にも必要です。

### 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

### 備考

非同期コマンドでは、パラメータの順序は CommandText と一致している必要があります。

## 関連情報

[EndExecuteReader\(IAsyncResult\) メソッド \[73 ページ\]](#)

[SADaReader クラス \[206 ページ\]](#)

### 1.4.3.3 BeginExecuteReader(AsyncCallback, object, CommandBehavior) メソッド

この SACommand で記述される SQL 文またはストアプロシージャの非同期実行を開始し、データベースサーバから 1 つまたは複数の結果セットを取得します。

#### 構文

##### Visual Basic

```
Public Function BeginExecuteReader (
    ByVal callback As AsyncCallback,
    ByVal stateObject As Object,
    ByVal behavior As CommandBehavior
) As IAsyncResult
```

##### C#

```
public IAsyncResult BeginExecuteReader (
    AsyncCallback callback,
    object stateObject,
    CommandBehavior behavior
)
```

## パラメータ

**callback** コマンドの実行が終了すると起動される System.AsyncCallback デリゲート。コールバックが必要ないことを示すには、NULL (Microsoft Visual Basic の場合は Nothing) を渡します。

**stateObject** コールバックプロシージャに渡される、ユーザ定義のステータスオブジェクト。コールバックプロシージャからこのオブジェクトを取得するには、System.IAsyncResult.AsyncState プロパティを使用します。

**behavior** クエリの結果の記述と、接続への影響の記述の System.Data.CommandBehavior フラグのビット単位の組み合わせ。

## 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、返されたローを取得するために使用する SADaReader オブジェクトを返す、EndExecuteReader(IAsyncResult) を起動する場合にも必要です。

## 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

## 備考

非同期コマンドでは、パラメータの順序は `CommandText` と一致している必要があります。

## 関連情報

[EndExecuteReader\(IAsyncResult\) メソッド \[73 ページ\]](#)

[SADataReader クラス \[206 ページ\]](#)

### 1.4.3.4 BeginExecuteReader(CommandBehavior) メソッド

この `SACommand` で記述される SQL 文またはストアドプロシージャの非同期実行を開始し、データベースサーバから1つまたは複数の結果セットを取得します。

#### 構文

##### Visual Basic

```
Public Function BeginExecuteReader (ByVal behavior As CommandBehavior) As IAsyncResult
```

##### C#

```
public IAsyncResult BeginExecuteReader (CommandBehavior behavior)
```

## パラメータ

**behavior** クエリの結果の記述と、接続への影響の記述の `System.Data.CommandBehavior` フラグのビット単位の組み合わせ。

## 戻り値

ポーリング、結果の待機、または両方に使用できる System.IAsyncResult。この値は、返されたローを取得するために使用する SADATAReader オブジェクトを返す、EndExecuteReader(IAsyncResult) を起動する場合にも必要です。

## 例外

**SAException class** コマンドテキストの実行中に発生したエラー。

## 備考

非同期コマンドでは、パラメータの順序は CommandText と一致している必要があります。

## 関連情報

[EndExecuteReader\(IAsyncResult\) メソッド \[73 ページ\]](#)

[SADATAReader クラス \[206 ページ\]](#)

## 1.4.4 Cancel() メソッド

SACOMMAND オブジェクトの実行をキャンセルします。

### 構文

#### Visual Basic

```
Public Overrides Sub Cancel ()
```

#### C#

```
public override void Cancel ()
```

## 備考

キャンセル対象がない場合は、何も行われません。実行中のコマンドがある場合は、「文の実行がユーザによって中断させられました。」という例外がスローされます。

## 1.4.5 CreateDbParameter() メソッド

System.Data.Common.DbParameter オブジェクトの新しいインスタンスを作成します。

### 構文

#### Visual Basic

```
Protected Overrides Function CreateDbParameter () As DbParameter
```

#### C#

```
protected override DbParameter CreateDbParameter ()
```

### 戻り値

System.Data.Common.DbParameter オブジェクト。

## 1.4.6 CreateParameter() メソッド

SACommand オブジェクトにパラメータを指定するために SAPParameter オブジェクトを提供します。

### 構文

#### Visual Basic

```
Public Shadows Function CreateParameter () As SAPParameter
```

#### C#

```
public new SAPParameter CreateParameter ()
```

### 戻り値

SAPParameter オブジェクトとして返される新しいパラメータ。

### 備考

ストアードプロシージャとその他一部の SQL 文は、疑問符 (?) によって文のテキストに示されているパラメータを使用できます。

CreateParameter メソッドは、SAPParameter オブジェクトを提供します。SAPParameter オブジェクトにプロパティを設定することで、パラメータの値やデータ型などを指定します。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

### 1.4.7 Dispose(bool) メソッド

オブジェクトに関連付けられているリソースを解放します。

#### 構文

##### Visual Basic

```
Protected Overrides Sub Dispose (ByVal disposing As Boolean)
```

##### C#

```
protected override void Dispose (bool disposing)
```

### 1.4.8 EndExecuteNonQuery(IAsyncResult) メソッド

SQL 文またはストアプロシージャの非同期実行を終了します。

#### 構文

##### Visual Basic

```
Public Function EndExecuteNonQuery (ByVal asyncResult As IAsyncResult) As Integer
```

##### C#

```
public unsafe int EndExecuteNonQuery (IAsyncResult asyncResult)
```

## パラメータ

**asyncResult** SACommand.BeginExecuteNonQuery への呼び出しによって返される IAsyncResult。

## 戻り値

影響されるローの数 (SACommand.ExecuteNonQuery と同じ動作)。

## 例外

**ArgumentException** asyncResult パラメータが null (Microsoft Visual Basic の Nothing) です。  
**InvalidOperationException** 1 回のコマンド実行に対して SACommand.EndExecuteNonQuery(IAsyncResult) が複数回呼び出されたか、メソッドがその実行メソッドと一致しませんでした。

## 備考

BeginExecuteNonQuery を呼び出すごとに、EndExecuteNonQuery を呼び出します。呼び出しは、BeginExecuteNonQuery が返されてから行います。ADO.NET はスレッドに対応していないため、各自で BeginExecuteNonQuery が返されたことを確認する必要があります。EndExecuteNonQuery に渡される IAsyncResult は、完了する BeginExecuteNonQuery 呼び出しから返される IAsyncResult と同じです。EndExecuteNonQuery を呼び出して、BeginExecuteReader への呼び出しを終了すると、エラーになります。逆についても同様です。

コマンドの実行中にエラーが発生すると、EndExecuteNonQuery が呼び出されるときに例外がスローされます。

実行の完了を待機するには、4 とおりの方法があります。

(1) EndExecuteNonQuery を呼び出します。

EndExecuteNonQuery を呼び出すと、コマンドが完了するまでブロックします。次に例を示します。

```
SAConnection conn = new SAConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACommand cmd = new SACommand(
    "UPDATE Departments"
    + " SET DepartmentName = 'Engineering'"
    + " WHERE DepartmentID=100",
    conn );
IAsyncResult res = cmd.BeginExecuteNonQuery();
// perform other work
// this will block until the command completes
int rowCount = cmd.EndExecuteNonQuery( res );
```

(2) IAsyncResult の IsCompleted プロパティをポーリングします。

次に例を示します。

```
SAConnection conn = new SAConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACommand cmd = new SACommand(
    "UPDATE Departments"
    + " SET DepartmentName = 'Engineering'"
    + " WHERE DepartmentID=100",
    conn );
IAsyncResult res = cmd.BeginExecuteNonQuery();
while( !res.IsCompleted ) {
    // do other work
```

```
}
// this will not block because the command is finished
int rowCount = cmd.EndExecuteNonQuery( res );
```

(3) `IAsyncResult.AsyncWaitHandle` プロパティを使用して同期オブジェクトを取得し、その状態で待機します。

次に例を示します。

```
SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACCommand cmd = new SACCommand(
    "UPDATE Departments"
    + " SET DepartmentName = 'Engineering'"
    + " WHERE DepartmentID=100",
    conn );
IAsyncResult res = cmd.BeginExecuteNonQuery();
// perform other work
WaitHandle wh = res.AsyncWaitHandle;
wh.WaitOne();
// this will not block because the command is finished
int rowCount = cmd.EndExecuteNonQuery( res );
```

(4) `BeginExecuteNonQuery` の呼び出し時にコールバック関数を指定します。

次に例を示します。

```
private void callbackFunction( IAsyncResult ar ) {
    SACCommand cmd = (SACCommand) ar.AsyncState;
    // this won't block since the command has completed
    int rowCount = cmd.EndExecuteNonQuery( ar );
}
// elsewhere in the code
private void DoStuff() {
    SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
    conn.Open();
    SACCommand cmd = new SACCommand(
        "UPDATE Departments"
        + " SET DepartmentName = 'Engineering'"
        + " WHERE DepartmentID=100",
        conn );
    IAsyncResult res = cmd.BeginExecuteNonQuery( callbackFunction, cmd );
    // perform other work. The callback function is
    // called when the command completes
}
```

コールバック関数は別のスレッドで実行するため、スレッド化されたプログラム内でのユーザインタフェースの更新に関する通常の注意が適用されます。

## 関連情報

[BeginExecuteNonQuery\(\) メソッド \[61 ページ\]](#)



## 1.4.9 EndExecuteReader(IAsyncResult) メソッド

SQL 文またはストアプロシージャの非同期実行を終了し、要求された `SADaReader` を返します。

### 構文

#### Visual Basic

```
Public Function EndExecuteReader (ByVal asyncResult As IAsyncResult) As SADaReader
```

#### C#

```
public unsafe SADaReader EndExecuteReader (IAsyncResult asyncResult)
```

### パラメータ

**asyncResult** `SACommand.BeginExecuteReader` への呼び出しによって返される `IAsyncResult`。

### 戻り値

要求されたローの取り出しに使用する `SADaReader` オブジェクト (`SACommand.ExecuteReader` と同じ動作)。

### 例外

**ArgumentException** `asyncResult` パラメータが `null` (Microsoft Visual Basic の `Nothing`) です。

**InvalidOperationException** 1 回のコマンド実行に対して `SACommand.EndExecuteReader(IAsyncResult)` が複数回呼び出されたか、メソッドがその実行メソッドと一致しませんでした。

### 備考

`BeginExecuteReader` を呼び出すごとに、`EndExecuteReader` を呼び出します。呼び出しは、`BeginExecuteReader` が返されてから行います。ADO.NET はスレッドに対応していないため、各自で `BeginExecuteReader` が返されたことを確認する必要があります。`EndExecuteReader` に渡される `IAsyncResult` は、完了する `BeginExecuteReader` 呼び出しから返される `IAsyncResult` と同じです。`EndExecuteReader` を呼び出して、`BeginExecuteNonQuery` への呼び出しを終了すると、エラーになります。逆についても同様です。

コマンドの実行中にエラーが発生すると、`EndExecuteReader` が呼び出されるときに例外がスローされます。

実行の完了を待機するには、4 とおりの方法があります。

(1) EndExecuteReader を呼び出す。

EndExecuteReader を呼び出すと、コマンドが完了するまでブロックします。次に例を示します。

```
SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACCommand cmd = new SACCommand( "SELECT * FROM Departments", conn );
IAsyncResult res = cmd.BeginExecuteReader();
// perform other work
// this blocks until the command completes
SADaReader reader = cmd.EndExecuteReader( res );
```

(2) IAsyncResult の IsCompleted プロパティをポーリングします。

次に例を示します。

```
SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACCommand cmd = new SACCommand( "SELECT * FROM Departments", conn );
IAsyncResult res = cmd.BeginExecuteReader();
while( !res.IsCompleted ) {
    // do other work
}
// this does not block because the command is finished
SADaReader reader = cmd.EndExecuteReader( res );
```

(3) IAsyncResult.AsyncWaitHandle プロパティを使用して同期オブジェクトを取得し、その状態で待機します。

次に例を示します。

```
SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
conn.Open();
SACCommand cmd = new SACCommand( "SELECT * FROM Departments", conn );
IAsyncResult res = cmd.BeginExecuteReader();
// perform other work
WaitHandle wh = res.AsyncWaitHandle;
wh.WaitOne();
// this does not block because the command is finished
SADaReader reader = cmd.EndExecuteReader( res );
```

(4) BeginExecuteReader の呼び出し時にコールバック関数を指定します。

次に例を示します。

```
private void callbackFunction( IAsyncResult ar ) {
    SACCommand cmd = (SACCommand) ar.AsyncState;
    // this does not block since the command has completed
    SADaReader reader = cmd.EndExecuteReader();
}
// elsewhere in the code
private void DoStuff() {
    SACConnection conn = new SACConnection("DSN=SQL Anywhere 17 Demo");
    conn.Open();
    SACCommand cmd = new SACCommand( "SELECT * FROM Departments", conn );
    IAsyncResult res = cmd.BeginExecuteReader( callbackFunction, cmd );
    // perform other work. The callback function is
    // called when the command completes
}
```

コールバック関数は別のスレッドで実行するため、スレッド化されたプログラム内でのユーザインタフェースの更新に関する通常の注意が適用されます。

## 関連情報

[BeginExecuteReader\(\) メソッド \[64 ページ\]](#)

[SADataReader クラス \[206 ページ\]](#)

### 1.4.10 ExecuteDbDataReader(CommandBehavior) メソッド

接続に対してコマンドテキストを実行します。

#### 構文

##### Visual Basic

```
Protected Overrides Function ExecuteDbDataReader (ByVal behavior As CommandBehavior) As DbDataReader
```

##### C#

```
protected override DbDataReader ExecuteDbDataReader (CommandBehavior behavior)
```

## パラメータ

**behavior** System.Data.CommandBehavior のインスタンス。

## 戻り値

System.Data.Common.DbDataReader。

### 1.4.11 ExecuteNonQuery() メソッド

結果セットを返さない文 (INSERT、UPDATE、DELETE など) や、データ定義文を実行します。

#### 構文

##### Visual Basic

```
Public Overrides Function ExecuteNonQuery () As Integer
```

##### C#

```
public override unsafe int ExecuteNonQuery ()
```

## 戻り値

影響を受けたローの数。

## 備考

DataSet を使用せず、ExecuteNonQuery を使用してデータベースのデータを変更します。これを行うには、UPDATE、INSERT、または DELETE 文を実行します。

ExecuteNonQuery はローを返しません。パラメータにマッピングされる出力パラメータまたは戻り値にはデータが入力されます。

UPDATE、INSERT、DELETE 文の場合、戻り値はコマンドの影響を受けるローの数です。その他すべての文のタイプとロールバックの場合、戻り値は -1 です。

## 関連情報

[ExecuteReader\(\) メソッド \[77 ページ\]](#)

## 1.4.12 ExecuteReader メソッド

結果セットを返す SQL 文を実行します。

## オーバードリスト

変更子とタイプ	オーバード名	説明
public new SADATAReader	<a href="#">ExecuteReader() [77 ページ]</a>	結果セットを返す SQL 文を実行します。
public new SADATAReader	<a href="#">ExecuteReader(CommandBehavior) [78 ページ]</a>	結果セットを返す SQL 文を実行します。

このセクションの内容:

[ExecuteReader\(\) メソッド \[77 ページ\]](#)

結果セットを返す SQL 文を実行します。

[ExecuteReader\(CommandBehavior\) メソッド \[78 ページ\]](#)

結果セットを返す SQL 文を実行します。

## 1.4.12.1 ExecuteReader() メソッド

結果セットを返す SQL 文を実行します。

### 構文

#### Visual Basic

```
Public Shadows Function ExecuteReader () As SqlDataReader
```

#### C#

```
public new SqlDataReader ExecuteReader ()
```

### 戻り値

SqlDataReader オブジェクトとして返される結果セット。

### 備考

この文は、必要に応じて CommandText と Parameters を持つ現在の SqlCommand オブジェクトです。SqlDataReader オブジェクトは、読み込み専用、前方専用の結果セットです。修正可能な結果セットの場合、SqlDataAdapter を使用します。

### 関連情報

[ExecuteNonQuery\(\) メソッド \[75 ページ\]](#)

[SqlDataReader クラス \[206 ページ\]](#)

[SqlDataAdapter クラス \[181 ページ\]](#)

[CommandText プロパティ \[85 ページ\]](#)

[Parameters プロパティ \[89 ページ\]](#)

## 1.4.12.2 ExecuteReader(CommandBehavior) メソッド

結果セットを返す SQL 文を実行します。

### 構文

#### Visual Basic

```
Public Shadows Function ExecuteReader (ByVal behavior As CommandBehavior) As  
SADaReader
```

#### C#

```
public new SADaReader ExecuteReader (CommandBehavior behavior)
```

### パラメータ

**behavior** CloseConnection、Default、KeyInfo、SchemaOnly、SequentialAccess、SingleResult、SingleRow のいずれか 1 つ。このパラメータの詳細については、.NET Framework のマニュアルの CommandBehavior 列挙を参照してください。

### 戻り値

SADaReader オブジェクトとして返される結果セット。

### 備考

この文は、必要に応じて CommandText と Parameters を持つ現在の SACommand オブジェクトです。SADaReader オブジェクトは、読み込み専用、前方専用の結果セットです。修正可能な結果セットの場合、SADaAdapter を使用します。

### 関連情報

[ExecuteNonQuery\(\) メソッド \[75 ページ\]](#)

[SADaReader クラス \[206 ページ\]](#)

[SADaAdapter クラス \[181 ページ\]](#)

[CommandText プロパティ \[85 ページ\]](#)

[Parameters プロパティ \[89 ページ\]](#)

## 1.4.13 ExecuteReaderAsync メソッド

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public new Task< SDataReader >	<a href="#">ExecuteReaderAsync() [79 ページ]</a>	SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。
public new Task< SDataReader >	<a href="#">ExecuteReaderAsync(CancellationTok en) [80 ページ]</a>	SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。
public new Task< SDataReader >	<a href="#">ExecuteReaderAsync(CommandBehavi or) [81 ページ]</a>	SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。
public new Task< SDataReader >	<a href="#">ExecuteReaderAsync(CommandBehavi or, CancellationToken) [82 ページ]</a>	SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

このセクションの内容:

#### [ExecuteReaderAsync\(\) メソッド \[79 ページ\]](#)

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### [ExecuteReaderAsync\(CancellationTok en\) メソッド \[80 ページ\]](#)

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### [ExecuteReaderAsync\(CommandBehavi or\) メソッド \[81 ページ\]](#)

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### [ExecuteReaderAsync\(CommandBehavi or, CancellationToken\) メソッド \[82 ページ\]](#)

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

## 1.4.13.1 ExecuteReaderAsync() メソッド

SACCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

### 構文

#### Visual Basic

```
Public Shadows Function ExecuteReaderAsync () As Task< SDataReader >
```

## C#

```
public new Task< SDataReader > ExecuteReaderAsync ()
```

## 戻り値

非同期操作を表すタスク。

## 例外

**SAException class** データベースサーバはコマンドテキストの実行中にエラーを返しました。

## 備考

返された Task オブジェクトを介して例外がレポートされます。

## 1.4.13.2 ExecuteReaderAsync(CancellationTokens) メソッド

SqlCommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

### 構文

#### Visual Basic

```
Public Shadows Function ExecuteReaderAsync (ByVal cancellationTokens As CancellationTokens) As Task< SDataReader >
```

#### C#

```
public new Task< SDataReader > ExecuteReaderAsync (CancellationTokens cancellationTokens)
```

## パラメータ

**cancellationTokens** キャンセル手順。



## 戻り値

非同期操作を表すタスク。

## 例外

**SAException class** データベースサーバはコマンドテキストの実行中にエラーを返しました。

## 備考

キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。返された Task オブジェクトを介して例外がレポートされます。

### 1.4.13.3 ExecuteReaderAsync(CommandBehavior) メソッド

SACommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### 構文

##### Visual Basic

```
Public Shadows Function ExecuteReaderAsync (ByVal behavior As CommandBehavior)  
As Task< SDataReader >
```

##### C#

```
public new Task< SDataReader > ExecuteReaderAsync (CommandBehavior behavior)
```

## パラメータ

**behavior** 文の実行およびデータの取得のオプション。

## 戻り値

非同期操作を表すタスク。

## 例外

**SQLException class** データベースサーバはコマンドテキストの実行中にエラーを返しました。

## 備考

返された Task オブジェクトを介して例外がレポートされます。

### 1.4.13.4 ExecuteReaderAsync(CommandBehavior, CancellationToken) メソッド

SACommand.ExecuteReader の非同期バージョン。結果セットを返す SQL 文を実行します。

#### 構文

##### Visual Basic

```
Public Shadows Function ExecuteReaderAsync (  
    ByVal behavior As CommandBehavior,  
    ByVal cancellationToken As CancellationTok  
) As Task< SDataReader >
```

##### C#

```
public new Task< SDataReader > ExecuteReaderAsync (  
    CommandBehavior behavior,  
    CancellationTok  
)
```

## パラメータ

**behavior** 文の実行およびデータの取得のオプション。

**cancellationToken** キャンセル手順。

## 戻り値

非同期操作を表すタスク。

## 例外

**SAException class** データベースサーバはコマンドテキストの実行中にエラーを返しました。

## 備考

キャンセルトークンを使用すると、コマンドタイムアウトになる前に操作の取り消しを要求できます。返された Task オブジェクトを介して例外がレポートされます。

## 1.4.14 ExecuteScalar() メソッド

単一の値を返す文を実行します。

### 構文

#### Visual Basic

```
Public Overrides Function ExecuteScalar () As Object
```

#### C#

```
public override object ExecuteScalar ()
```

## 戻り値

結果セットの最初のローの最初のカラム。結果セットが空の場合は NULL 参照。

## 備考

複数のローとカラムを返すクエリでこのメソッドが呼び出されると、最初のローの最初のカラムのみが返されます。

## 1.4.15 Prepare() メソッド

データソース上で SACommand を準備またはコンパイルします。

### 構文

#### Visual Basic

```
Public Overrides Sub Prepare ()
```

#### C#

```
public override void Prepare ()
```

### 備考

Prepare を呼び出してから ExecuteNonQuery、ExecuteReader、ExecuteScalar のいずれかのメソッドを呼び出すと、Size プロパティによって指定されている値よりも大きいパラメータ値は、元々指定されているパラメータのサイズに自動的にトランケートされ、トランケーションエラーは返されません。

トランケーションは次のデータ型に対してのみ実行されます。

- CHAR
- VARCHAR
- LONG VARCHAR
- TEXT
- NCHAR
- NVARCHAR
- LONG NVARCHAR
- NTEXT
- BINARY
- LONG BINARY
- VARBINARY
- IMAGE

Size プロパティの指定がなく、デフォルト値が使用されている場合、データはトランケートされません。

### 関連情報

[ExecuteNonQuery\(\) メソッド \[75 ページ\]](#)

[ExecuteReader\(\) メソッド \[77 ページ\]](#)

[ExecuteScalar\(\) メソッド \[83 ページ\]](#)

## 1.4.16 ResetCommandTimeout() メソッド

CommandTimeout プロパティをデフォルト値の 30 秒にリセットします。

### 構文

#### Visual Basic

```
Public Sub ResetCommandTimeout ()
```

#### C#

```
public void ResetCommandTimeout ()
```

## 1.4.17 CommandText プロパティ

SQL 文またはストアプロシージャのテキストを取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property CommandText As String
```

#### C#

```
public override string CommandText {get;set;}
```

## 備考

実行する SQL 文またはストアプロシージャの名前。デフォルトは、空の文字列です。

## 関連情報

[SACommand\(\) コンストラクタ \[58 ページ\]](#)

## 1.4.18 CommandTimeout プロパティ

この機能は .NET データプロバイダでサポートされていません。

### 構文

#### Visual Basic

```
Public Overrides Property CommandTimeout As Integer
```

#### C#

```
public override int CommandTimeout {get;set;}
```

### 備考

```
To set a request timeout, use the following example. cmd.CommandText = "SET OPTION  
request_timeout = 30";  
cmd.ExecuteNonQuery();
```

## 1.4.19 CommandType プロパティ

SACommand によって示されるコマンドのタイプを取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property CommandType As CommandType
```

#### C#

```
public override CommandType CommandType {get;set;}
```

### 備考

System.Data.CommandType 値の 1 つ。デフォルトは System.Data.CommandType.Text です。

サポートされているコマンドタイプは、次のとおりです。

- System.Data.CommandType.StoredProcedure。この CommandType を指定する場合、コマンドテキストはストアードプロシージャの名前にし、引数はすべて SAParameter オブジェクトとして指定してください。
- System.Data.CommandType.Text。これはデフォルト値です。

CommandType プロパティを StoredProcedure に設定する場合、CommandText プロパティをストアードプロシージャの名前に設定してください。Execute メソッドの 1 つを呼び出すと、このストアードプロシージャが実行されます。

疑問符 (?) プレースホルダを使用してパラメータを渡します。次に例を示します。

```
SELECT * FROM Customers WHERE ID = ?
```

SAPParameter オブジェクトが SAPParameterCollection に追加される順序は、パラメータの疑問符の位置にそのまま対応している必要があります。

## 1.4.20 Connection プロパティ

SACommand オブジェクトが適用される接続オブジェクトを取得または設定します。

### 構文

#### Visual Basic

```
Public Shadows Property Connection As SAConnection
```

#### C#

```
public new SAConnection Connection {get;set;}
```

### 備考

デフォルト値は NULL 参照です。Visual Basic では、これは Nothing です。

## 1.4.21 DbConnection プロパティ

この SACommand オブジェクトによって使用される System.Data.Common.DbConnection を取得または設定します。

### 構文

#### Visual Basic

```
Protected Overrides Property DbConnection As DbConnection
```

#### C#

```
protected override DbConnection DbConnection {get;set;}
```

戻り値

データソースへの接続。

関連情報

[SACommand クラス \[53 ページ\]](#)

## 1.4.22 DbParameterCollection プロパティ

System.Data.Common.DbParameter オブジェクトのコレクションを取得します。

≡ 構文

**Visual Basic**

```
Protected ReadOnly Overrides Property DbParameterCollection As  
DbParameterCollection
```

**C#**

```
protected override DbParameterCollection DbParameterCollection {get;}
```

戻り値

SQL 文またはストアドプロシージャのパラメータ。

## 1.4.23 DbTransaction プロパティ

この SACommand オブジェクトが実行される System.Data.Common.DbTransaction を取得または設定します。

≡ 構文

**Visual Basic**

```
Protected Overrides Property DbTransaction As DbTransaction
```

**C#**

```
protected override DbTransaction DbTransaction {get;set;}
```



## 戻り値

.NET Framework データプロバイダの Command オブジェクトが実行されるトランザクション。デフォルト値は NULL 参照 (Visual Basic の Nothing) です。

## 1.4.24 DesignTimeVisible プロパティ

Windows Form Designer 制御で SACCommand を参照できるようにするかどうかを指定する値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property DesignTimeVisible As Boolean
```

#### C#

```
public override bool DesignTimeVisible {get;set;}
```

## 備考

デフォルトは true です。

SACCommand インスタンスを参照できるようにする場合は true、このインスタンスを参照できないようにする場合は false です。デフォルトは false です。

## 1.4.25 Parameters プロパティ

現在の文のパラメータのコレクションを指定します。

### 構文

#### Visual Basic

```
Public ReadOnly Shadows Property Parameters As SAPParameterCollection
```

#### C#

```
public new SAPParameterCollection Parameters {get;}
```

## 備考

CommandText で疑問符を使用してパラメータを示します。

SQL 文またはストアプロシージャのパラメータ。デフォルト値は空のコレクションです。

CommandType を Text に設定する場合、疑問符プレースホルダを使用してパラメータを渡します。次に例を示します。

```
SELECT * FROM Customers WHERE ID = ?
```

SAParameter オブジェクトが SAParameterCollection に追加される順序は、コマンドテキストのパラメータの疑問符の位置にそのまま対応している必要があります。

コレクション内のパラメータが、実行されるクエリの要件と一致しない場合、エラーが発生したり例外がスローされることがあります。

## 関連情報

[SAParameterCollection クラス \[286 ページ\]](#)

## 1.4.26 Transaction プロパティ

SACommand が実行される SATransaction オブジェクトを指定します。

### 構文

#### Visual Basic

```
Public Shadows Property Transaction As SATransaction
```

#### C#

```
public new SATransaction Transaction {get;set;}
```

## 備考

デフォルト値は NULL 参照です。Visual Basic では、これは Nothing です。

Transaction プロパティがすでに特定の値に設定されていて、コマンドが実行されている場合、このプロパティは設定できません。SACommand オブジェクトと同じ SAConnection オブジェクトに接続されていない SATransaction オブジェクトに対して Transaction プロパティを設定した場合、次に文を実行しようとする例外がスローされます。

詳細については、「トランザクション処理」を参照してください。

## 関連情報

[SATransaction クラス \[337 ページ\]](#)

### 1.4.27 UpdatedRowSource プロパティ

SDataAdapter の Update メソッドによって使用されるときにコマンドの結果が DataRow に適用される方法を取得または設定します。

#### 構文

##### Visual Basic

```
Public Overrides Property UpdatedRowSource As UpdateRowSource
```

##### C#

```
public override UpdateRowSource UpdatedRowSource {get;set;}
```

## 備考

UpdatedRowSource 値の 1 つ。デフォルト値は UpdateRowSource.OutputParameters です。コマンドが自動的に生成される場合、このプロパティは UpdateRowSource.None です。

結果セットと出力パラメータの両方を返す UpdatedRowSource.Both は、サポートされていません。

## 1.5 SACommandBuilder クラス

DataSet の変更内容を関連するデータベース内のデータに一致させる単一テーブルの SQL 文を生成する方法です。

#### 構文

##### Visual Basic

```
Public NotInheritable Class SACommandBuilder Inherits  
System.Data.Common.DbCommandBuilder
```

##### C#

```
public sealed class SACommandBuilder : System.Data.Common.DbCommandBuilder
```

## メンバー

SACommandBuilder のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変数とタイプ	コンストラクタ	説明
public	<a href="#">SACommandBuilder [94 ページ]</a>	SACommandBuilder オブジェクトを初期化します。

### メソッド

変数とタイプ	メソッド	説明
protected override void	<a href="#">ApplyParameterInfo(DbParameter, DataRow, StatementType, bool) [95 ページ]</a>	System.Data.Common.DbCommandBuilder のプロバイダ実装が追加のパラメータプロパティを処理することを許可します。
public static void	<a href="#">DeriveParameters(SACommand) [96 ページ]</a>	指定された SACommand オブジェクトの Parameters コレクションに入力します。
public new SACommand	<a href="#">GetDeleteCommand [97 ページ]</a>	SADDataAdapter.Update が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された SACommand オブジェクトを返します。
public new SACommand	<a href="#">GetInsertCommand [100 ページ]</a>	Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。
protected override string	<a href="#">GetParameterName [102 ページ]</a>	指定されたパラメータの名前を @p# のフォーマットで返します。
protected override string	<a href="#">GetParameterPlaceholder(int) [104 ページ]</a>	関連する SQL 文のパラメータのプレースホルダを返します。
protected override DataTable	<a href="#">GetSchemaTable(DbCommand) [105 ページ]</a>	SACommandBuilder オブジェクトのスキーマテーブルを返します。
public new SACommand	<a href="#">GetUpdateCommand [106 ページ]</a>	Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。
protected override DbCommand	<a href="#">InitializeCommand(DbCommand) [108 ページ]</a>	System.Data.Common.DbCommand の System.Data.Common.DbCommand.CommandTimeout、System.Data.Common.DbCommand.Transaction、System.Data.Common.DbCommand.CommandType、および System.Data.Common.DbCommand.UpdatedRowSource プロパティをリセットします。

変更子とタイプ	メソッド	説明
public override string	<a href="#">QuotIdentifier(string) [109 ページ]</a>	識別子に埋め込まれた引用符が適切にエスケープされた、引用符なし識別子の正しい引用付きの形式を返します。
protected override void	<a href="#">SetRowUpdatingHandler(DbDataAdapter) [109 ページ]</a>	SDataAdapter オブジェクトの SDataAdapter.RowUpdating イベントを処理する SACCommandBuilder オブジェクトを登録します。
public override string	<a href="#">UnquotIdentifier(string) [110 ページ]</a>	識別子に埋め込まれた引用符が適切にアンエスケープされた、引用符付き識別子の正しい引用符なしの形式を返します。

## プロパティ

変更子とタイプ	プロパティ	説明
public new SDataAdapter	<a href="#">DataAdapter [111 ページ]</a>	文を生成する SDataAdapter を指定します。

## このセクションの内容:

### [SACCommandBuilder コンストラクタ \[94 ページ\]](#)

SACCommandBuilder オブジェクトを初期化します。

### [ApplyParameterInfo\(DbParameter, DataRow, StatementType, bool\) メソッド \[95 ページ\]](#)

System.Data.Common.DbCommandBuilder のプロバイダ実装が追加のパラメータプロパティを処理することを許可します。

### [DeriveParameters\(SACCommand\) メソッド \[96 ページ\]](#)

指定された SACCommand オブジェクトの Parameters コレクションに入力します。

### [GetDeleteCommand メソッド \[97 ページ\]](#)

SDataAdapter.Update が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された SACCommand オブジェクトを返します。

### [GetInsertCommand メソッド \[100 ページ\]](#)

Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACCommand オブジェクトを返します。

### [GetParameterName メソッド \[102 ページ\]](#)

指定されたパラメータの名前を @p# のフォーマットで返します。

### [GetParameterPlaceholder\(int\) メソッド \[104 ページ\]](#)

関連する SQL 文のパラメータのプレースホルダを返します。

### [GetSchemaTable\(DbCommand\) メソッド \[105 ページ\]](#)

SACCommandBuilder オブジェクトのスキーマテーブルを返します。

### [GetUpdateCommand メソッド \[106 ページ\]](#)

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACCommand オブジェクトを返します。

### [InitializeCommand\(DbCommand\) メソッド \[108 ページ\]](#)

System.Data.Common.DbCommand の System.Data.Common.DbCommand.CommandTimeout、System.Data.Common.DbCommand.Transaction、System.Data.Common.DbCommand.CommandType、および System.Data.Common.DbCommand.UpdatedRowSource プロパティをリセットします。

#### [QuoteIdentifier\(string\) メソッド \[109 ページ\]](#)

識別子に埋め込まれた引用符が適切にエスケープされた、引用符なし識別子の正しい引用付きの形式を返します。

#### [SetRowUpdatingHandler\(DbDataAdapter\) メソッド \[109 ページ\]](#)

SDataAdapter オブジェクトの SDataAdapter.RowUpdating イベントを処理する SACommandBuilder オブジェクトを登録します。

#### [UnquoteIdentifier\(string\) メソッド \[110 ページ\]](#)

識別子に埋め込まれた引用符が適切にアンエスケープされた、引用符付き識別子の正しい引用符なしの形式を返します。

#### [DataAdapter プロパティ \[111 ページ\]](#)

文を生成する SDataAdapter を指定します。

## 1.5.1 SACommandBuilder コンストラクタ

SCommandBuilder オブジェクトを初期化します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SCommandBuilder() [95 ページ]</a>	SCommandBuilder オブジェクトを初期化します。
public	<a href="#">SCommandBuilder(SDataAdapter) [95 ページ]</a>	SCommandBuilder オブジェクトを初期化します。

このセクションの内容:

#### [SCommandBuilder\(\) コンストラクタ \[95 ページ\]](#)

SCommandBuilder オブジェクトを初期化します。

#### [SCommandBuilder\(SDataAdapter\) コンストラクタ \[95 ページ\]](#)

SCommandBuilder オブジェクトを初期化します。

### 1.5.1.1 SACommandBuilder() コンストラクタ

SACommandBuilder オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SACommandBuilder ()
```

##### C#

```
public SACommandBuilder ()
```

### 1.5.1.2 SACommandBuilder(SDataAdapter) コンストラクタ

SACommandBuilder オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SACommandBuilder (ByVal adapter As SDataAdapter)
```

##### C#

```
public SACommandBuilder (SDataAdapter adapter)
```

#### パラメータ

**adapter** 調整文を生成する SDataAdapter オブジェクト。

### 1.5.2 ApplyParameterInfo(DbParameter, DataRow, StatementType, bool) メソッド

System.Data.Common.DbCommandBuilder のプロバイダ実装が追加のパラメータプロパティを処理することを許可します。

#### 構文

##### Visual Basic

```
Protected Overrides Sub ApplyParameterInfo (
```

```
ByVal parameter As DbParameter,  
ByVal row As DataRow,  
ByVal statementType As StatementType,  
ByVal whereClause As Boolean  
)
```

#### C#

```
protected override void ApplyParameterInfo (  
    DbParameter parameter,  
    DataRow row,  
    StatementType statementType,  
    bool whereClause  
)
```

## パラメータ

**parameter** 追加的な修正が適用される System.Data.Common.DbParameter。

**row** SqlDataReader.GetSchemaTable によって提供されるスキーマテーブルからの System.Data.DataRow。

**statementType** INSERT、UPDATE、または DELETE といった、生成されるコマンドのタイプ。

**whereClause** パラメータが UPDATE または DELETE WHERE 文の一部である場合は true、INSERT または UPDATE 値の一部である場合は false。

## 関連情報

[GetSchemaTable\(\) メソッド \[228 ページ\]](#)

## 1.5.3 DeriveParameters(SACommand) メソッド

指定された SACommand オブジェクトの Parameters コレクションに入力します。

### 構文

#### Visual Basic

```
Public Shared Sub DeriveParameters (ByVal command As SACommand)
```

#### C#

```
public static void DeriveParameters (SACommand command)
```



## パラメータ

`command` パラメータを抽出する `SACCommand` オブジェクト。

## 備考

このメソッドは、`SACCommand` に指定されたストアプロシージャに対して使用されます。

`DeriveParameters` は、`SACCommand` の既存のパラメータ情報を上書きします。

`DeriveParameters` には、データベースサーバへの追加呼び出しが必要です。パラメータ情報が事前に分かっている場合、情報を明示的に設定して `Parameters` コレクションに入力する方が効率的です。

## 1.5.4 GetDeleteCommand メソッド

`SDataAdapter.Update` が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された `SACCommand` オブジェクトを返します。

### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public new SACCommand	<a href="#">GetDeleteCommand() [98 ページ]</a>	<code>SDataAdapter.Update</code> が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された <code>SACCommand</code> オブジェクトを返します。
public new SACCommand	<a href="#">GetDeleteCommand(bool) [98 ページ]</a>	<code>SDataAdapter.Update</code> が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された <code>SACCommand</code> オブジェクトを返します。

このセクションの内容:

#### [GetDeleteCommand\(\) メソッド \[98 ページ\]](#)

`SDataAdapter.Update` が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された `SACCommand` オブジェクトを返します。

#### [GetDeleteCommand\(bool\) メソッド \[98 ページ\]](#)

`SDataAdapter.Update` が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された `SACCommand` オブジェクトを返します。

## 1.5.4.1 GetDeleteCommand() メソッド

SDataAdapter.Update が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された SCommand オブジェクトを返します。

### 構文

#### Visual Basic

```
Public Shadows Function GetDeleteCommand () As SCommand
```

#### C#

```
public new SCommand GetDeleteCommand ()
```

### 戻り値

削除の実行に必要な、自動的に生成された SCommand オブジェクト。

### 備考

GetDeleteCommand メソッドは、実行対象の SCommand オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、GetDeleteCommand を修正されたコマンドの基礎としても使用できます。たとえば、GetDeleteCommand を呼び出して CommandTimeout 値を修正してから、SDataAdapter で値を明示的に設定できます。

アプリケーションが Update または GetDeleteCommand を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、RefreshSchema を明示的に呼び出す必要があります。この処理を行わないと、GetDeleteCommand は古い文の情報を使用し続けます。

## 1.5.4.2 GetDeleteCommand(bool) メソッド

SDataAdapter.Update が呼び出されたときにデータベース上で DELETE オペレーションを実行する、生成された SCommand オブジェクトを返します。

### 構文

#### Visual Basic

```
Public Shadows Function GetDeleteCommand (ByVal useColumnsForParameterNames As Boolean) As SCommand
```

## C#

```
public new SACommand GetDeleteCommand (bool useColumnsForParameterNames)
```

## パラメータ

**useColumnsForParameterNames** true の場合、可能であれば、カラム名に一致するパラメータ名を生成します。  
false の場合、@p1、@p2などを生成します。

## 戻り値

削除の実行に必要な、自動的に生成された SACommand オブジェクト。

## 備考

GetDeleteCommand メソッドは、実行対象の SACommand オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、GetDeleteCommand を修正されたコマンドの基礎としても使用できます。たとえば、GetDeleteCommand を呼び出して CommandTimeout 値を修正してから、SADDataAdapter で値を明示的に設定できます。

アプリケーションが Update または GetDeleteCommand を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、RefreshSchema を明示的に呼び出す必要があります。この処理を行わないと、GetDeleteCommand は古い文の情報を使用し続けます。

## 1.5.5 GetInsertCommand メソッド

Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public new SACommand	<a href="#">GetInsertCommand() [100 ページ]</a>	Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。
public new SACommand	<a href="#">GetInsertCommand(bool) [101 ページ]</a>	Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。

このセクションの内容:

#### [GetInsertCommand\(\) メソッド \[100 ページ\]](#)

Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。

#### [GetInsertCommand\(bool\) メソッド \[101 ページ\]](#)

Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。

### 1.5.5.1 GetInsertCommand() メソッド

Update が呼び出されたときにデータベース上で INSERT オペレーションを実行する、生成された SACommand オブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Shadows Function GetInsertCommand () As SACommand
```

##### C#

```
public new SACommand GetInsertCommand ()
```

## 戻り値

挿入の実行に必要な、自動的に生成された `SACCommand` オブジェクト。

## 備考

`GetInsertCommand` メソッドは、実行対象の `SACCommand` オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、`GetInsertCommand` を修正されたコマンドの基礎としても使用できます。たとえば、`GetInsertCommand` を呼び出して `CommandTimeout` 値を修正してから、`SADDataAdapter` で値を明示的に設定できます。

アプリケーションが `Update` または `GetInsertCommand` を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、`RefreshSchema` を明示的に呼び出す必要があります。この処理を行わないと、`GetInsertCommand` は、正しくない可能性がある古い文の情報を使用し続けます。

## 関連情報

[GetDeleteCommand\(\) メソッド \[98 ページ\]](#)

### 1.5.5.2 GetInsertCommand(bool) メソッド

`Update` が呼び出されたときにデータベース上で `INSERT` オペレーションを実行する、生成された `SACCommand` オブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Shadows Function GetInsertCommand (ByVal useColumnsForParameterNames As Boolean) As SACCommand
```

##### C#

```
public new SACCommand GetInsertCommand (bool useColumnsForParameterNames)
```

## パラメータ

`useColumnsForParameterNames true` の場合、可能であれば、カラム名に一致するパラメータ名を生成します。`false` の場合、`@p1`、`@p2`などを生成します。

## 戻り値

挿入の実行に必要な、自動的に生成された SACommand オブジェクト。

## 備考

GetInsertCommand メソッドは、実行対象の SACommand オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、GetInsertCommand を修正されたコマンドの基礎としても使用できます。たとえば、GetInsertCommand を呼び出して CommandTimeout 値を修正してから、SADDataAdapter で値を明示的に設定できます。

アプリケーションが Update または GetInsertCommand を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、RefreshSchema を明示的に呼び出す必要があります。この処理を行わないと、GetInsertCommand は、正しくない可能性がある古い文の情報を使用し続けます。

## 関連情報

[GetDeleteCommand\(\) メソッド \[98 ページ\]](#)

## 1.5.6 GetParameterName メソッド

指定されたパラメータの名前を @p# のフォーマットで返します。

## オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
protected override string	<a href="#">GetParameterName(int) [103 ページ]</a>	指定されたパラメータの名前を @p# のフォーマットで返します。
protected override string	<a href="#">GetParameterName(string) [103 ページ]</a>	指定された部分的パラメータ名に基づいて、完全なパラメータ名を返します。

このセクションの内容:

[GetParameterName\(int\) メソッド \[103 ページ\]](#)

指定されたパラメータの名前を @p# のフォーマットで返します。

[GetParameterName\(string\) メソッド \[103 ページ\]](#)

指定された部分的パラメータ名に基づいて、完全なパラメータ名を返します。

## 1.5.6.1 GetParameterName(int) メソッド

指定されたパラメータの名前を @p# のフォーマットで返します。

### 構文

#### Visual Basic

```
Protected Overrides Function GetParameterName (ByVal index As Integer) As String
```

#### C#

```
protected override string GetParameterName (int index)
```

### パラメータ

**index** パラメータの名前の一部として含まれる数。

### 戻り値

パラメータ名の一部として指定された数を追加されたパラメータ名。

### 備考

カスタムコマンドビルダを構築するときに使用します。

## 1.5.6.2 GetParameterName(string) メソッド

指定された部分的パラメータ名に基づいて、完全なパラメータ名を返します。

### 構文

#### Visual Basic

```
Protected Overrides Function GetParameterName (ByVal parameterName As String) As String
```

#### C#

```
protected override string GetParameterName (string parameterName)
```

## パラメータ

**parameterName** パラメータの名前の一部。

## 戻り値

要求されたパラメータ名の一部に対応する完全パラメータ名。

## 1.5.7 GetParameterPlaceholder(int) メソッド

関連する SQL 文のパラメータのプレースホルダを返します。

### 構文

#### Visual Basic

```
Protected Overrides Function GetParameterPlaceholder (ByVal index As Integer)  
As String
```

#### C#

```
protected override string GetParameterPlaceholder (int index)
```

## パラメータ

**index** パラメータの名前的一部分として含まれる数。

## 戻り値

指定された数が追加されたパラメータ名。



## 1.5.8 GetSchemaTable(DbCommand) メソッド

SACommandBuilder オブジェクトのスキーマテーブルを返します。

### 構文

#### Visual Basic

```
Protected Overrides Function GetSchemaTable (ByVal sourceCommand As DbCommand)  
As DataTable
```

#### C#

```
protected override DataTable GetSchemaTable (DbCommand sourceCommand)
```

### パラメータ

**sourceCommand** 対応するスキーマテーブルを取得する System.Data.Common.DbCommand。

### 戻り値

特定の System.Data.Common.DbCommand のスキーマを表す System.Data.DataTable。

### 関連情報

[SACommandBuilder クラス \[91 ページ\]](#)

## 1.5.9 GetUpdateCommand メソッド

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public new SACommand	<a href="#">GetUpdateCommand() [106 ページ]</a>	Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。
public new SACommand	<a href="#">GetUpdateCommand(bool) [107 ページ]</a>	Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

このセクションの内容:

#### [GetUpdateCommand\(\) メソッド \[106 ページ\]](#)

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

#### [GetUpdateCommand\(bool\) メソッド \[107 ページ\]](#)

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

### 1.5.9.1 GetUpdateCommand() メソッド

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Shadows Function GetUpdateCommand () As SACommand
```

##### C#

```
public new SACommand GetUpdateCommand ()
```

## 戻り値

更新の実行に必要な、自動的に生成された SACommand オブジェクト。

## 備考

GetUpdateCommand メソッドは、実行対象の SACommand オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、GetUpdateCommand を修正されたコマンドの基礎としても使用できます。たとえば、GetUpdateCommand を呼び出して CommandTimeout 値を修正してから、SDataAdapter で値を明示的に設定できます。

アプリケーションが Update または GetUpdateCommand を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、RefreshSchema を明示的に呼び出す必要があります。この処理を行わないと、GetUpdateCommand は、正しくない可能性がある古い文の情報を使用し続けます。

### 1.5.9.2 GetUpdateCommand(bool) メソッド

Update が呼び出されたときにデータベース上で UPDATE オペレーションを実行する、生成された SACommand オブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Shadows Function GetUpdateCommand (ByVal useColumnsForParameterNames As Boolean) As SACommand
```

##### C#

```
public new SACommand GetUpdateCommand (bool useColumnsForParameterNames)
```

## パラメータ

**useColumnsForParameterNames** true の場合、可能であれば、カラム名に一致するパラメータ名を生成します。  
false の場合、@p1、@p2などを生成します。

## 戻り値

更新の実行に必要な、自動的に生成された SACommand オブジェクト。

## 備考

GetUpdateCommand メソッドは、実行対象の SACommand オブジェクトを返すため、情報やトラブルシューティング用として役に立ちます。

または、GetUpdateCommand を修正されたコマンドの基礎としても使用できます。たとえば、GetUpdateCommand を呼び出して CommandTimeout 値を修正してから、SDataAdapter で値を明示的に設定できます。

アプリケーションが Update または GetUpdateCommand を呼び出すと、SQL 文が最初に生成されます。SQL 文が最初に生成された後で、アプリケーションが文を変更する場合、RefreshSchema を明示的に呼び出す必要があります。この処理を行わないと、GetUpdateCommand は、正しくない可能性がある古い文の情報を使用し続けます。

## 1.5.10 InitializeCommand(DbCommand) メソッド

System.Data.Common.DbCommand の System.Data.Common.DbCommand.CommandTimeout、System.Data.Common.DbCommand.Transaction、System.Data.Common.DbCommand.CommandType、および System.Data.Common.DbCommand.UpdatedRowSource プロパティをリセットします。

### 構文

#### Visual Basic

```
Protected Overrides Function InitializeCommand (ByVal command As DbCommand) As DbCommand
```

#### C#

```
protected override DbCommand InitializeCommand (DbCommand command)
```

## パラメータ

**command** 対応する挿入、更新、または削除コマンドを生成するコマンドビルダによって使用される System.Data.Common.DbCommand。

## 戻り値

各挿入、更新、または削除オペレーションで使用される System.Data.Common.DbCommand インスタンス。NULL 値を渡すことで、InitializeCommand メソッドは、SACommandBuilder オブジェクトに関連付けられている SELECT 文に基づいて System.Data.Common.DbCommand オブジェクトを作成できます。

## 関連情報

[SACommandBuilder クラス \[91 ページ\]](#)

### 1.5.11 QuoteIdentifier(string) メソッド

識別子に埋め込まれた引用符が適切にエスケープされた、引用符なし識別子の正しい引用付きの形式を返します。

#### 構文

##### Visual Basic

```
Public Overrides Function QuoteIdentifier (ByVal unquotedIdentifier As String)  
As String
```

##### C#

```
public override string QuoteIdentifier (string unquotedIdentifier)
```

## パラメータ

**unquotedIdentifier** 引用符を付ける必要のある引用符なしの識別子を表す文字列。

## 戻り値

埋め込まれた引用符が適切にエスケープされた、引用符なしの識別子の引用付きの形式を表す文字列を返します。

### 1.5.12 SetRowUpdatingHandler(DbDataAdapter) メソッド

SADaDataAdapter オブジェクトの SADaDataAdapter.RowUpdating イベントを処理する SACommandBuilder オブジェクトを登録します。

#### 構文

##### Visual Basic

```
Protected Overrides Sub SetRowUpdatingHandler (ByVal adapter As DbDataAdapter)
```

##### C#

```
protected override void SetRowUpdatingHandler (DbDataAdapter adapter)
```

## パラメータ

**adapter** 更新に使用される SDataAdapter オブジェクト。

## 関連情報

[SACommandBuilder クラス \[91 ページ\]](#)

[SDataAdapter クラス \[181 ページ\]](#)

[RowUpdating イベント \[205 ページ\]](#)

## 1.5.13 UnquoteIdentifier(string) メソッド

識別子に埋め込まれた引用符が適切にアンエスケープされた、引用符付き識別子の正しい引用符なしの形式を返します。

### 構文

#### Visual Basic

```
Public Overrides Function UnquoteIdentifier (ByVal quotedIdentifier As String)  
As String
```

#### C#

```
public override string UnquoteIdentifier (string quotedIdentifier)
```

## パラメータ

**quotedIdentifier** 埋め込まれた引用符が削除される、引用符付きの識別子を表す文字列。

## 戻り値

埋め込まれた引用符が適切にアンエスケープされた、引用符付き識別子の引用符なしの形式を表す文字列を返します。

## 1.5.14 DataAdapter プロパティ

文を生成する SDataAdapter を指定します。

### 構文

#### Visual Basic

```
Public Shadows Property DataAdapter As SDataAdapter
```

#### C#

```
public new SDataAdapter DataAdapter {get;set;}
```

### 備考

SDataAdapter オブジェクト。

SACCommandBuilder の新しいインスタンスを作成すると、この SDataAdapter に関連付けられている既存の SACCommandBuilder が解放されます。

## 1.6 SACCommLinksOptionsBuilder クラス

SACConnection クラスが使用する接続文字列の CommLinks オプション部分を作成および管理する簡単な方法を提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SACCommLinksOptionsBuilder
```

#### C#

```
public sealed class SACCommLinksOptionsBuilder
```

### メンバー

SACCommLinksOptionsBuilder のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SACommLinksOptionsBuilder [113 ページ]</a>	SACommLinksOptionsBuilder オブジェクトを初期化します。

## メソッド

変更子とタイプ	メソッド	説明
public bool	<a href="#">GetUseLongNameAsKeyword() [115 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。
public void	<a href="#">SetUseLongNameAsKeyword(bool) [115 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。
public override string	<a href="#">ToString() [116 ページ]</a>	SACommLinksOptionsBuilder オブジェクトを文字列表現に変換します。

## プロパティ

変更子とタイプ	プロパティ	説明
public bool	<a href="#">All [117 ページ]</a>	ALL CommLinks オプションを取得または設定します。
public string	<a href="#">ConnectionString [117 ページ]</a>	構築される接続文字列を取得または設定します。
public bool	<a href="#">SharedMemory [117 ページ]</a>	SharedMemory プロトコルを取得または設定します。
public SATcpOptionsBuilder	<a href="#">TcpOptionsBuilder [118 ページ]</a>	TCP オプション文字列の作成に使用する TcpOptionsBuilder オブジェクトを取得または設定します。
public string	<a href="#">TcpOptionsString [118 ページ]</a>	TCP オプションの文字列を取得または設定します。

## 備考

接続パラメータのリストについては、「[接続パラメータ](#)」を参照してください。

このセクションの内容:

[SACommLinksOptionsBuilder コンストラクタ \[113 ページ\]](#)

SACommLinksOptionsBuilder オブジェクトを初期化します。

[GetUseLongNameAsKeyword\(\) メソッド \[115 ページ\]](#)

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。

[SetUseLongNameAsKeyword\(bool\) メソッド \[115 ページ\]](#)

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。

[ToString\(\) メソッド \[116 ページ\]](#)



SACommLinksOptionsBuilder オブジェクトを文字列表現に変換します。

#### [All プロパティ \[117 ページ\]](#)

ALL CommLinks オプションを取得または設定します。

#### [ConnectionString プロパティ \[117 ページ\]](#)

構築される接続文字列を取得または設定します。

#### [SharedMemory プロパティ \[117 ページ\]](#)

SharedMemory プロトコルを取得または設定します。

#### [TcpOptionsBuilder プロパティ \[118 ページ\]](#)

TCP オプション文字列の作成に使用する TcpOptionsBuilder オブジェクトを取得または設定します。

#### [TcpOptionsString プロパティ \[118 ページ\]](#)

TCP オプションの文字列を取得または設定します。

## 1.6.1 SACommLinksOptionsBuilder コンストラクタ

SACommLinksOptionsBuilder オブジェクトを初期化します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SACommLinksOptionsBuilder()</a> [114 ページ]	SACommLinksOptionsBuilder オブジェクトを初期化します。
public	<a href="#">SACommLinksOptionsBuilder(string)</a> [114 ページ]	SACommLinksOptionsBuilder オブジェクトを初期化します。

このセクションの内容:

#### [SACommLinksOptionsBuilder\(\) コンストラクタ \[114 ページ\]](#)

SACommLinksOptionsBuilder オブジェクトを初期化します。

#### [SACommLinksOptionsBuilder\(string\) コンストラクタ \[114 ページ\]](#)

SACommLinksOptionsBuilder オブジェクトを初期化します。

## 1.6.1.1 SACommLinksOptionsBuilder() コンストラクタ

SACommLinksOptionsBuilder オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SACommLinksOptionsBuilder ()
```

#### C#

```
public SACommLinksOptionsBuilder ()
```

### 例

次の文は、SACommLinksOptionsBuilder オブジェクトを初期化します。

```
SACommLinksOptionsBuilder commLinks = new SACommLinksOptionsBuilder ();
```

## 1.6.1.2 SACommLinksOptionsBuilder(string) コンストラクタ

SACommLinksOptionsBuilder オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SACommLinksOptionsBuilder (ByVal options As String)
```

#### C#

```
public SACommLinksOptionsBuilder (string options)
```

### パラメータ

**options** CommLinks 接続パラメータ文字列。接続パラメータのリストについては、「接続パラメータ」を参照してください。

### 例

次の文は、SACommLinksOptionsBuilder オブジェクトを初期化します。

```
SACommLinksOptionsBuilder commLinks =  
new SACommLinksOptionsBuilder ("TCPIP (DoBroadcast=ALL;Timeout=20)");
```

## 1.6.2 GetUseLongNameAsKeyword() メソッド

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。

### 構文

#### Visual Basic

```
Public Function GetUseLongNameAsKeyword () As Boolean
```

#### C#

```
public bool GetUseLongNameAsKeyword ()
```

### 戻り値

長い接続パラメータ名を使用して接続文字列を作成する場合は true、それ以外の場合は false。

### 備考

接続パラメータの名前には、長い名前と短い名前の 2 種類があります。たとえば、接続文字列に ODBC データソースの名前を指定する場合は、DataSourceName と DSN のいずれかを使用します。デフォルトでは、長い接続パラメータ名を使用して接続文字列を作成します。

### 関連情報

[SetUseLongNameAsKeyword\(bool\) メソッド \[115 ページ\]](#)

## 1.6.3 SetUseLongNameAsKeyword(bool) メソッド

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。

### 構文

#### Visual Basic

```
Public Sub SetUseLongNameAsKeyword (ByVal useLongNameAsKeyword As Boolean)
```

#### C#

```
public void SetUseLongNameAsKeyword (bool useLongNameAsKeyword)
```

## パラメータ

**useLongNameAsKeyword** 長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値。

## 備考

デフォルトでは、長い接続パラメータ名が使用されます。

## 関連情報

[GetUseLongNameAsKeyword\(\) メソッド \[115 ページ\]](#)

## 1.6.4 ToString() メソッド

SACommLinksOptionsBuilder オブジェクトを文字列表現に変換します。

### 構文

#### Visual Basic

```
Public Overrides Function ToString () As String
```

#### C#

```
public override string ToString ()
```

## 戻り値

構築されるオプション文字列。

## 1.6.5 All プロパティ

ALL CommLinks オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property All As Boolean
```

#### C#

```
public bool All {get;set;}
```

### 備考

最初に共有メモリプロトコルを使用して接続を試行し、次に使用可能なすべての通信プロトコルを使用します。使用する通信プロトコルが不明の場合は、この設定を使用してください。

## 1.6.6ConnectionString プロパティ

構築される接続文字列を取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionString As String
```

#### C#

```
public string ConnectionString {get;set;}
```

## 1.6.7 SharedMemory プロパティ

SharedMemory プロトコルを取得または設定します。

### 構文

#### Visual Basic

```
Public Property SharedMemory As Boolean
```

C#

```
public bool SharedMemory {get;set;}
```

## 1.6.8 TcpOptionsBuilder プロパティ

TCP オプション文字列の作成に使用する TcpOptionsBuilder オブジェクトを取得または設定します。

📄 構文

Visual Basic

```
Public Property TcpOptionsBuilder As SATcpOptionsBuilder
```

C#

```
public SATcpOptionsBuilder TcpOptionsBuilder {get;set;}
```

## 1.6.9 TcpOptionsString プロパティ

TCP オプションの文字列を取得または設定します。

📄 構文

Visual Basic

```
Public Property TcpOptionsString As String
```

C#

```
public string TcpOptionsString {get;set;}
```

## 1.7 SAConnection クラス

データベースへの接続を表します。

📄 構文

Visual Basic

```
Public NotInheritable Class SAConnection Inherits  
System.Data.Common.DbConnection
```

## C#

```
public sealed class SACConnection : System.Data.Common.DbConnection
```

## メンバー

SACConnection のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SACConnection [122 ページ]</a>	SACConnection オブジェクトを初期化します。

### メソッド

変更子とタイプ	メソッド	説明
protected override DbTransaction	<a href="#">BeginDbTransaction(IsolationLevel) [125 ページ]</a>	データベーストランザクションを開始します。
public new SATransaction	<a href="#">BeginTransaction [125 ページ]</a>	トランザクションオブジェクトを返します。
public override void	<a href="#">ChangeDatabase(string) [129 ページ]</a>	開いている SACConnection の現在のデータベースを変更します。
public static void	<a href="#">ChangePassword(string, string) [129 ページ]</a>	接続文字列に示されたユーザのパスワードを、指定された新しいパスワードに変更します。
public static void	<a href="#">ClearAllPools() [130 ページ]</a>	すべての接続プールを空にします。
public static void	<a href="#">ClearPool(SACConnection) [130 ページ]</a>	指定の接続に関連付けられている接続プールを空にします。
public override void	<a href="#">Close() [131 ページ]</a>	データベース接続を閉じます。
public new SACCommand	<a href="#">CreateCommand() [131 ページ]</a>	SACCommand オブジェクトを初期化します。
protected override DbCommand	<a href="#">CreateDbCommand() [132 ページ]</a>	現在の接続に関連付けられている System.Data.Common.DbCommand を作成して返します。
protected override void	<a href="#">Dispose(bool) [133 ページ]</a>	オブジェクトに関連付けられているリソースを解放します。
public void	<a href="#">EnlistDistributedTransaction(System.EnterpriseServices.ITransaction) [133 ページ]</a>	指定されたトランザクションに分散トランザクションとしてエンリストします。
public override void	<a href="#">EnlistTransaction(System.Transactions.Transaction) [134 ページ]</a>	指定されたトランザクションに分散トランザクションとしてエンリストします。
public override DataTable	<a href="#">GetSchema [134 ページ]</a>	サポートされているスキーマコレクションのリストを返します。

変更子とタイプ	メソッド	説明
public override unsafe void	<a href="#">Open() [141 ページ]</a>	SACConnection.ConnectionString で指定されたプロパティ設定を使用してデータベース接続を開きます。

## プロパティ

変更子とタイプ	プロパティ	説明
public override string	<a href="#">ConnectionString [141 ページ]</a>	データベース接続文字列を指定します。
public override int	<a href="#">ConnectionTimeout [143 ページ]</a>	接続試行がエラーでタイムアウトするまでの秒数を取得します。
public SACredential	<a href="#">Credential [143 ページ]</a>	この接続の SACredential オブジェクトを取得または接続します。
public override string	<a href="#">Database [144 ページ]</a>	現在のデータベースの名前を取得します。
public override string	<a href="#">DataSource [144 ページ]</a>	データベースサーバの名前を取得します。
public string	<a href="#">InitString [145 ページ]</a>	接続の確立直後に実行するコマンドです。
public override string	<a href="#">ServerVersion [145 ページ]</a>	クライアントが接続するデータベースサーバのインスタンスのバージョンが含まれている文字列を取得します。
public override ConnectionState	<a href="#">State [146 ページ]</a>	SACConnection オブジェクトのステータスを示します。

## イベント

変更子とタイプ	イベント	説明
public SAInfoMessageEventHandler	<a href="#">InfoMessage [146 ページ]</a>	データベースサーバが警告または情報メッセージを返すときに発生します。
public override StateChangeEventHandler	<a href="#">StateChange [147 ページ]</a>	SACConnection オブジェクトのステータスが変ると発生します。

## 備考

接続パラメータのリストについては、「.NET 接続パラメータ」と「接続パラメータ」を参照してください。

このセクションの内容:

[SACConnection コンストラクタ \[122 ページ\]](#)

SACConnection オブジェクトを初期化します。

[BeginDbTransaction\(IsolationLevel\) メソッド \[125 ページ\]](#)

データベーストランザクションを開始します。

[BeginTransaction メソッド \[125 ページ\]](#)

トランザクションオブジェクトを返します。

[ChangeDatabase\(string\) メソッド \[129 ページ\]](#)



開いている SAConnection の現在のデータベースを変更します。

[ChangePassword\(string, string\) メソッド \[129 ページ\]](#)

接続文字列に示されたユーザのパスワードを、指定された新しいパスワードに変更します。

[ClearAllPools\(\) メソッド \[130 ページ\]](#)

すべての接続プールを空にします。

[ClearPool\(SAConnection\) メソッド \[130 ページ\]](#)

指定の接続に関連付けられている接続プールを空にします。

[Close\(\) メソッド \[131 ページ\]](#)

データベース接続をクローズします。

[CreateCommand\(\) メソッド \[131 ページ\]](#)

SACommand オブジェクトを初期化します。

[CreateDbCommand\(\) メソッド \[132 ページ\]](#)

現在の接続に関連付けられている System.Data.Common.DbCommand を作成して返します。

[Dispose\(bool\) メソッド \[133 ページ\]](#)

オブジェクトに関連付けられているリソースを解放します。

[EnlistDistributedTransaction\(System.EnterpriseServices.ITransaction\) メソッド \[133 ページ\]](#)

指定されたトランザクションに分散トランザクションとしてエンリストします。

[EnlistTransaction\(System.Transactions.Transaction\) メソッド \[134 ページ\]](#)

指定されたトランザクションに分散トランザクションとしてエンリストします。

[GetSchema メソッド \[134 ページ\]](#)

サポートされているスキーマコレクションのリストを返します。

[Open\(\) メソッド \[141 ページ\]](#)

SAConnection.ConnectionString で指定されたプロパティ設定を使用してデータベース接続を開きます。

[ConnectionString プロパティ \[141 ページ\]](#)

データベース接続文字列を指定します。

[ConnectionTimeout プロパティ \[143 ページ\]](#)

接続試行がエラーでタイムアウトするまでの秒数を取得します。

[Credential プロパティ \[143 ページ\]](#)

この接続の SACredential オブジェクトを取得または接続します。

[Database プロパティ \[144 ページ\]](#)

現在のデータベースの名前を取得します。

[DataSource プロパティ \[144 ページ\]](#)

データベースサーバの名前を取得します。

[InitString プロパティ \[145 ページ\]](#)

接続の確立直後に実行するコマンドです。

[ServerVersion プロパティ \[145 ページ\]](#)

クライアントが接続するデータベースサーバのインスタンスのバージョンが含まれている文字列を取得します。

[State プロパティ \[146 ページ\]](#)

SAConnection オブジェクトのステータスを示します。

[InfoMessage イベント \[146 ページ\]](#)

データベースサーバが警告または情報メッセージを返すときに発生します。

[StateChange イベント \[147 ページ\]](#)

SACConnection オブジェクトのステータスが変わると発生します。

## 1.7.1 SACConnection コンストラクタ

SACConnection オブジェクトを初期化します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SACConnection() [122 ページ]</a>	SACConnection オブジェクトを初期化します。
public	<a href="#">SACConnection(string) [123 ページ]</a>	SACConnection オブジェクトを初期化します。
public	<a href="#">SACConnection(string, SACredential) [124 ページ]</a>	接続文字列を指定した SACConnection クラスの新しいインスタンス、およびユーザ ID とパスワードを含んでいる SACredential オブジェクトを初期化します。

このセクションの内容:

[SACConnection\(\) コンストラクタ \[122 ページ\]](#)

SACConnection オブジェクトを初期化します。

[SACConnection\(string\) コンストラクタ \[123 ページ\]](#)

SACConnection オブジェクトを初期化します。

[SACConnection\(string, SACredential\) コンストラクタ \[124 ページ\]](#)

接続文字列を指定した SACConnection クラスの新しいインスタンス、およびユーザ ID とパスワードを含んでいる SACredential オブジェクトを初期化します。

### 1.7.1.1 SACConnection() コンストラクタ

SACConnection オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SACConnection ()
```

## C#

```
public SAConnection ()
```

## 備考

接続を開いてから、データベース操作を実行してください。

## 1.7.1.2 SAConnection(string) コンストラクタ

SAConnection オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SAConnection (ByVal connectionString As String)
```

#### C#

```
public SAConnection (string connectionString)
```

## パラメータ

**connectionString** 接続文字列です。接続文字列は keyword=value のペアがセミコロンで区切られたリストです。接続パラメータのリストについては、「[接続パラメータ](#)」を参照してください。

## 備考

接続を開いてから、データベース操作を実行してください。

### 例

次の文は、hr という名前のデータベースサーバ上で動作している policies という名前のデータベースへの接続について SAConnection オブジェクトを初期化します。接続では、ユーザ ID admin とパスワード money を使用します。

```
SAConnection conn = new SAConnection(  
    "UID=admin;PWD=money;SERVER=hr;DBN=policies" );  
conn.Open();
```

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

### 1.7.1.3 SAConnection(string, SACredential) コンストラクタ

接続文字列を指定した SAConnection クラスの新しいインスタンス、およびユーザ ID とパスワードを含んでいる SACredential オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SAConnection (  
    ByVal connectionString As String,  
    ByVal credential As SACredential  
)
```

##### C#

```
public SAConnection (  
    string connectionString,  
    SACredential credential  
)
```

## パラメータ

**connectionString** 接続文字列です。接続文字列は keyword=value のペアがセミコロンで区切られたリストです。接続パラメータのリストについては、「[接続パラメータ](#)」を参照してください。

**credential** SACredential オブジェクト。認証情報が NULL の場合、SAConnection(String, SACredential) は機能的に SAConnection(String) と同じです。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

## 1.7.2 BeginDbTransaction(IsolationLevel) メソッド

データベーストランザクションを開始します。

### 構文

#### Visual Basic

```
Protected Overrides Function BeginDbTransaction (ByVal isolationLevel As IsolationLevel) As DbTransaction
```

#### C#

```
protected override DbTransaction BeginDbTransaction (IsolationLevel isolationLevel)
```

### パラメータ

**isolationLevel** トランザクションの独立性レベルを指定します。

### 戻り値

新しいトランザクションを表すオブジェクト。

## 1.7.3 BeginTransaction メソッド

トランザクションオブジェクトを返します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public new SATransaction	<a href="#">BeginTransaction()</a> [126 ページ]	トランザクションオブジェクトを返します。
public new SATransaction	<a href="#">BeginTransaction(IsolationLevel)</a> [127 ページ]	トランザクションオブジェクトを返します。
public SATransaction	<a href="#">BeginTransaction(SAIsolationLevel)</a> [128 ページ]	トランザクションオブジェクトを返します。

このセクションの内容:

[BeginTransaction\(\) メソッド \[126 ページ\]](#)

トランザクションオブジェクトを返します。

[BeginTransaction\(IsolationLevel\) メソッド \[127 ページ\]](#)

トランザクションオブジェクトを返します。

[BeginTransaction\(SAIsolationLevel\) メソッド \[128 ページ\]](#)

トランザクションオブジェクトを返します。

### 1.7.3.1 BeginTransaction() メソッド

トランザクションオブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Shadows Function BeginTransaction () As SATransaction
```

##### C#

```
public new SATransaction BeginTransaction ()
```

#### 戻り値

新しいトランザクションを表す SATransaction オブジェクト。

#### 備考

トランザクションオブジェクトに関連付けられているコマンドは、単一のトランザクションとして実行されます。トランザクションは、Commit メソッドまたは Rollback メソッドへの呼び出しで終了します。

コマンドをトランザクションオブジェクトに関連付けるには、SACommand.Transaction プロパティを使用します。

#### 関連情報

[SATransaction クラス \[337 ページ\]](#)

[Transaction プロパティ \[90 ページ\]](#)

## 1.7.3.2 BeginTransaction(IsolationLevel) メソッド

トランザクションオブジェクトを返します。

### 構文

#### Visual Basic

```
Public Shadows Function BeginTransaction (ByVal isolationLevel As IsolationLevel) As SACTransaction
```

#### C#

```
public new SACTransaction BeginTransaction (IsolationLevel isolationLevel)
```

### パラメータ

**isolationLevel** SAIsolationLevel 列挙のメンバー。デフォルト値は ReadCommitted です。

### 戻り値

新しいトランザクションを表す SACTransaction オブジェクト。

### 備考

トランザクションオブジェクトに関連付けられているコマンドは、単一のトランザクションとして実行されます。トランザクションは、Commit メソッドまたは Rollback メソッドへの呼び出しで終了します。

コマンドをトランザクションオブジェクトに関連付けるには、SACCommand.Transaction プロパティを使用します。

### 例

```
SACTransaction tx =  
    conn.BeginTransaction( SAIsolationLevel.ReadUncommitted );
```

### 関連情報

[SACTransaction クラス \[337 ページ\]](#)

[Transaction プロパティ \[90 ページ\]](#)

### 1.7.3.3 BeginTransaction(SAIsolationLevel) メソッド

トランザクションオブジェクトを返します。

#### 構文

##### Visual Basic

```
Public Function BeginTransaction (ByVal isolationLevel As SAIsolationLevel) As  
    STransaction
```

##### C#

```
public STransaction BeginTransaction (SAIsolationLevel isolationLevel)
```

#### パラメータ

**isolationLevel** SAIsolationLevel 列挙のメンバー。デフォルト値は ReadCommitted です。

#### 戻り値

新しいトランザクションを表す STransaction オブジェクト。

#### 備考

トランザクションオブジェクトに関連付けられているコマンドは、単一のトランザクションとして実行されます。トランザクションは、Commit メソッドまたは Rollback メソッドへの呼び出しで終了します。

コマンドをトランザクションオブジェクトに関連付けるには、SACommand.Transaction プロパティを使用します。

詳細については、「トランザクション処理」を参照してください。

詳細については、「典型的な矛盾のケース」を参照してください。

#### 関連情報

[STransaction クラス \[337 ページ\]](#)



[Transaction プロパティ \[90 ページ\]](#)  
[SAIsolationLevel 列挙 \[352 ページ\]](#)  
[Commit\(\) メソッド \[339 ページ\]](#)  
[Rollback\(\) メソッド \[340 ページ\]](#)  
[Rollback\(string\) メソッド \[341 ページ\]](#)

## 1.7.4 ChangeDatabase(string) メソッド

開いている SAConnection の現在のデータベースを変更します。

### 構文

#### Visual Basic

```
Public Overrides Sub ChangeDatabase (ByVal database As String)
```

#### C#

```
public override void ChangeDatabase (string database)
```

### パラメータ

**database** 現在のデータベースの代わりに使用するデータベースの名前。

## 1.7.5 ChangePassword(string, string) メソッド

接続文字列に示されたユーザのパスワードを、指定された新しいパスワードに変更します。

### 構文

#### Visual Basic

```
Public Shared Sub ChangePassword (  
    ByVal connectionString As String,  
    ByVal newPassword As String  
)
```

#### C#

```
public static void ChangePassword (  
    string connectionString,  
    string newPassword  
)
```

## パラメータ

**connectionString** 目的のデータベースサーバに接続できるだけの情報が含まれる接続文字列。接続文字列には、ユーザ ID と現在のパスワードが含まれる可能性があります。

**newPassword** 設定する新しいパスワード。このパスワードは、最小文字数や特殊文字の要件など、データベースサーバ上で設定されているパスワードセキュリティポリシーに準拠する必要があります。

## 例外

**ArgumentNullException** connectionString または newPassword parameter のいずれかが null です。

**ArgumentException** 接続文字列に、統合化セキュリティを使用するオプションが含まれています。

## 1.7.6 ClearAllPools() メソッド

すべての接続プールを空にします。

### 構文

#### Visual Basic

```
Public Shared Sub ClearAllPools ()
```

#### C#

```
public static void ClearAllPools ()
```

## 1.7.7 ClearPool(SAConnection) メソッド

指定の接続に関連付けられている接続プールを空にします。

### 構文

#### Visual Basic

```
Public Shared Sub ClearPool (ByVal connection As SAConnection)
```

#### C#

```
public static void ClearPool (SAConnection connection)
```

## パラメータ

**connection** プールからクリアする SAConnection オブジェクト。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

## 1.7.8 Close() メソッド

データベース接続をクローズします。

### 構文

#### Visual Basic

```
Public Overrides Sub Close ()
```

#### C#

```
public override void Close ()
```

## 備考

Close メソッドは、保留中のトランザクションをロールバックします。次に、接続プールへの接続を解放します。また、接続プーリングが無効の場合は、接続を閉じます。StateChange イベントの処理中に Close が呼び出されても、追加の StateChange イベントは実行されません。アプリケーションは、Close を複数回呼び出すことができます。

## 1.7.9 CreateCommand() メソッド

SACommand オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Shadows Function CreateCommand () As SACommand
```

#### C#

```
public new SACommand CreateCommand ()
```

## 戻り値

SACommand オブジェクト。

## 備考

コマンドオブジェクトは SACConnection オブジェクトに関連付けられます。

## 関連情報

[SACommand クラス \[53 ページ\]](#)

[SACConnection クラス \[118 ページ\]](#)

## 1.7.10 CreateDbCommand() メソッド

現在の接続に関連付けられている System.Data.Common.DbCommand を作成して返します。

### 構文

#### Visual Basic

```
Protected Overrides Function CreateDbCommand () As DbCommand
```

#### C#

```
protected override DbCommand CreateDbCommand ()
```

## 戻り値

System.Data.Common.DbCommand オブジェクト。

## 1.7.11 Dispose(bool) メソッド

オブジェクトに関連付けられているリソースを解放します。

### 構文

#### Visual Basic

```
Protected Overrides Sub Dispose (ByVal disposing As Boolean)
```

#### C#

```
protected override void Dispose (bool disposing)
```

## 1.7.12 EnlistDistributedTransaction(System.EnterpriseServices.ITransaction) メソッド

指定されたトランザクションに分散トランザクションとしてエンリストします。

### 構文

#### Visual Basic

```
Public Sub EnlistDistributedTransaction (ByVal transaction As System.EnterpriseServices.ITransaction)
```

#### C#

```
public void EnlistDistributedTransaction  
(System.EnterpriseServices.ITransaction transaction)
```

## パラメータ

**transaction** エンリストする既存の System.EnterpriseServices.ITransaction への参照。

## 1.7.13 EnlistTransaction(System.Transactions.Transaction) メソッド

指定されたトランザクションに分散トランザクションとしてエンリストします。

### 構文

#### Visual Basic

```
Public Overrides Sub EnlistTransaction (ByVal transaction As System.Transactions.Transaction)
```

#### C#

```
public override void EnlistTransaction (System.Transactions.Transaction transaction)
```

### パラメータ

**transaction** エンリストする既存の System.Transactions.Transaction への参照。

## 1.7.14 GetSchema メソッド

サポートされているスキーマコレクションのリストを返します。

### オーバロードリスト

変更子とタイプ	オーバロード名	説明
public override DataTable	<a href="#">GetSchema()</a> [135 ページ]	サポートされているスキーマコレクションのリストを返します。
public override DataTable	<a href="#">GetSchema(string)</a> [135 ページ]	この SAConnection オブジェクトについて指定されたメタデータコレクションに関する情報を返します。
public override DataTable	<a href="#">GetSchema(string, string[])</a> [136 ページ]	この SAConnection オブジェクトのデータソースのスキーマ情報を返します。このとき、文字列が指定されている場合はスキーマ名として使用し、文字列配列が指定されている場合は制限値として使用します。

このセクションの内容:

[GetSchema\(\) メソッド \[135 ページ\]](#)

サポートされているスキーマコレクションのリストを返します。

#### [GetSchema\(string\) メソッド \[135 ページ\]](#)

この SAConnection オブジェクトについて指定されたメタデータコレクションに関する情報を返します。

#### [GetSchema\(string, string\[\]\) メソッド \[136 ページ\]](#)

この SAConnection オブジェクトのデータソースのスキーマ情報を返します。このとき、文字列が指定されている場合はスキーマ名として使用し、文字列配列が指定されている場合は制限値として使用します。

## 1.7.14.1 GetSchema() メソッド

サポートされているスキーマコレクションのリストを返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetSchema () As DataTable
```

#### C#

```
public override DataTable GetSchema ()
```

### 備考

使用可能なメタデータの詳細については、[GetSchema\(string,string\[\]\)](#) を参照してください。

## 1.7.14.2 GetSchema(string) メソッド

この SAConnection オブジェクトについて指定されたメタデータコレクションに関する情報を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetSchema (ByVal collection As String) As DataTable
```

#### C#

```
public override DataTable GetSchema (string collection)
```

## パラメータ

**collection** メタデータコレクションの名前。名前が提供されない場合は、MetaDataCollections が使用されます。

## 備考

使用可能なメタデータの詳細については、GetSchema(string,string[]) を参照してください。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

### 1.7.14.3 GetSchema(string, string[]) メソッド

この SAConnection オブジェクトのデータソースのスキーマ情報を返します。このとき、文字列が指定されている場合はスキーマ名として使用し、文字列配列が指定されている場合は制限値として使用します。

#### 構文

##### Visual Basic

```
Public Overrides Function GetSchema (  
    ByVal collection As String,  
    ByVal restrictions As String()  
) As DataTable
```

##### C#

```
public override DataTable GetSchema (  
    string collection,  
    string[] restrictions  
)
```

## 戻り値

スキーマ情報が格納されている DataTable。



## 備考

これらのメソッドを使用すると、データベースサーバに各種のメタデータを問い合わせることができます。メタデータの各型にはコレクション名が指定されており、そのデータを受け取るにはコレクション名を渡す必要があります。デフォルトのコレクション名は `MetaDataCollections` です。

引数を指定せずに、または `MetaDataCollections` というスキーマコレクション名を指定して `GetSchema` メソッドを呼び出すことによって、.NET データプロバイダに問い合わせをして、サポートされているスキーマコレクションのリストを判断できます。これによって、サポートされているスキーマコレクション (CollectionName)、それぞれがサポートする制限の数 (NumberOfRestrictions)、使用する識別子部分の数のリストから成る `DataTable` が返されます。

コレクション	メタデータ
Columns	データベース内のすべてのカラムに関する情報を返します。
DataSourceInformation	データベースサーバに関する情報を返します。
DataTypes	サポートされているデータ型のリストを返します。
ForeignKeys	データベース内のすべての外部キーに関する情報を返します。
IndexColumns	データベース内のすべてのインデックスカラムに関する情報を返します。
Indexes	データベース内のすべてのインデックスに関する情報を返します。
MetaDataCollections	すべてのコレクション名のリストを返します。
ProcedureParameters	データベース内のすべてのプロシージャパラメータに関する情報を返します。
Procedures	データベース内のすべてのプロシージャに関する情報を返します。
ReservedWords	データベースサーバが使用する予約語のリストを返します。
Restrictions	<code>GetSchema</code> で使用される制限に関する情報を返します。
Tables	データベース内のすべてのテーブルに関する情報を返します。
UserDefinedTypes	データベース内のすべてのユーザ定義データ型に関する情報を返します。
Users	データベース内のすべてのユーザに関する情報を返します。
ViewColumns	データベース内のすべてのビュー内のカラムに関する情報を返します。
Views	データベース内のすべてのビューに関する情報を返します。

これらのコレクション名は、`SAMetaDataCollectionNames` クラスの読み込み専用プロパティとしても使用できます。

返される結果は、`GetSchema` への呼び出しの中で制限の配列を指定することによってフィルタできます。

各コレクションで有効な制限は、次の文を呼び出すことで問い合わせることができます。

```
GetSchema( "Restrictions" )
```

コレクションが 4 つの制限を必要とする場合、制限パラメータは、NULL、または 4 つの値から成る文字列です。

特定の制限をフィルタするには、フィルタする文字列を配列内の所定の位置に指定し、使用しない位置には NULL を指定します。たとえば、`Tables` コレクションには 3 つの制限があります。Owner、Table、TableType です。

Table コレクションを table\_name でフィルタするには、次の手順に従います。

```
GetSchema( "Tables", new string[ ] { NULL, "my_table", NULL } )
```

この例は、my\_table という名前のすべてのテーブルに関する情報を返します。

```
GetSchema( "Tables", new string[ ] { "DBA", "my_table", NULL } )
```

この例は、DBA ユーザが所有する my\_table という名前のすべてのテーブルに関する情報を返します。

次に、各コレクションが返すカラムの概要リストを示します。カラムに制限を指定することで、コレクションが返すローの数を減らすことができる場合は、そのカラムの制限名をカッコ内に表示します。制限は、以下のリストで表示される順序で指定されます。

#### *Columns* コレクション

- TABLE\_SCHEMA (Owner)
- TABLE\_NAME (Table)
- COLUMN\_NAME (Column)
- ORDINAL\_POSITION
- DATA\_TYPE
- COLUMN\_DEFAULT
- IS\_NULLABLE
- NUMERIC\_PRECISION
- NUMERIC\_SCALE
- CHARACTER\_MAXIMUM\_LENGTH
- DATETIME\_PRECISION

#### *DataSourceInformation* コレクション

- CompositeIdentifierSeparatorPattern
- DataSourceProductName
- DataSourceProductVersion
- DataSourceProductVersionNormalized
- GroupByBehavior
- IdentifierPattern
- IdentifierCase
- OrderByColumnsInSelect
- ParameterMarkerFormat
- ParameterMarkerPattern
- ParameterNameMaxLength
- ParameterNamePattern
- QuotedIdentifierPattern
- QuotedIdentifierCase
- StatementSeparatorPattern
- StringLiteralPattern
- SupportedJoinOperators

#### *DataTypes* コレクション

- TypeName

- 
- ProviderDbType
  - ColumnSize
  - CreateFormat
  - CreateParameters
  - DataType
  - IsAutoIncrementable
  - IsBestMatch
  - IsCaseSensitive
  - IsFixedLength
  - IsFixedPrecisionScale
  - IsLong
  - IsNullable
  - IsSearchable
  - IsSearchableWithLike
  - IsUnsigned
  - MaximumScale
  - MinimumScale
  - IsConcurrencyType
  - IsLiteralSupported
  - LiteralPrefix
  - LiteralSuffix

#### *ForeignKeys* コレクション

- TABLE\_SCHEMA (Owner)
- TABLE\_NAME (Table)
- COLUMN\_NAME (Column)

#### *IndexColumns* コレクション

- TABLE\_SCHEMA (Owner)
- TABLE\_NAME (Table)
- INDEX\_NAME (Name)
- COLUMN\_NAME (Column)
- ORDER

#### *Indexes* コレクション

- TABLE\_SCHEMA (Owner)
- TABLE\_NAME (Table)
- INDEX\_NAME (Name)
- PRIMARY\_KEY
- IS\_UNIQUE

#### *MetaDataCollections* コレクション

- CollectionName
- NumberOfRestrictions
- NumberOfIdentifierParts

#### *ProcedureParameters* コレクション

- PROCEDURE\_SCHEMA (Owner)
- PROCEDURE\_NAME (Name)
- PARAMETER\_NAME (Parameter)
- DATA\_TYPE
- PARAMETER\_TYPE
- IS\_INPUT
- IS\_OUTPUT

#### *Procedures* コレクション

- PROCEDURE\_SCHEMA (Owner)
- PROCEDURE\_NAME (Name)

#### *ReservedWords* コレクション

- reserved\_word

#### *Restrictions* コレクション

- CollectionName
- RestrictionName
- RestrictionDefault
- RestrictionNumber

#### *Tables* コレクション

- TABLE\_SCHEMA (Owner)
- TABLE\_NAME (Table)
- TABLE\_TYPE (TableType) は、"BASE"、"VIEW"、"MAT VIEW"、"LCL TEMP"、"GBL TEMP"、"TEXT"、または "TEXT GBL TEMP"のいずれか。

#### *UserDefinedTypes* コレクション

- DATA\_TYPE
- DEFAULT
- PRECISION
- SCALE

#### *Users* コレクション

- USER\_NAME (UserName)
- RESOURCE\_AUTH
- DATABASE\_AUTH
- SCHEDULE\_AUTH
- USER\_GROUP

#### *ViewColumns* コレクション

- VIEW\_SCHEMA (Owner)
- VIEW\_NAME (Name)
- COLUMN\_NAME (Column)

#### *Views* コレクション

- VIEW\_SCHEMA (Owner)

- VIEW\_NAME (Name)

## 関連情報

[SACConnection クラス \[118 ページ\]](#)

### 1.7.15 Open() メソッド

SACConnection.ConnectionString で指定されたプロパティ設定を使用してデータベース接続を開きます。

#### 構文

##### Visual Basic

```
Public Overrides Sub Open ()
```

##### C#

```
public override unsafe void Open ()
```

## 関連情報

[ConnectionString プロパティ \[141 ページ\]](#)

### 1.7.16 ConnectionString プロパティ

データベース接続文字列を指定します。

#### 構文

##### Visual Basic

```
Public Overrides Property ConnectionString As String
```

##### C#

```
public override string ConnectionString {get;set;}
```

## 備考

ConnectionString は、データベースサーバ接続文字列のフォーマットとできるかぎり同じになるよう定義されています。ただし、Persist Security Info 値が false (デフォルト) に設定されている場合、返される接続文字列は、ユーザ定義の ConnectionString からセキュリティ情報を除いたものと同じになります。Persist Security Info 値を true に設定しないかぎり、NET データプロバイダは返される接続文字列にパスワードを保持しません。

ConnectionString プロパティを使用して、さまざまなデータソースに接続できます。

ConnectionString プロパティを設定できるのは、接続が閉じられている場合のみです。接続文字列値の多くには、対応する読み込み専用プロパティがあります。接続文字列が設定されると、エラーが検出されないかぎり、これらのプロパティのすべてが更新されます。エラーが検出されると、いずれのプロパティも更新されません。SAConnection プロパティは、ConnectionString に含まれているこれらの設定のみを返します。

閉じられた接続上で ConnectionString をリセットすると、パスワードを含むすべての接続文字列値と関連プロパティがリセットされます。

プロパティが設定されると、接続文字列の事前検証が実行されます。アプリケーションが Open メソッドを呼び出すと、接続文字列が完全に検証されます。接続文字列に無効なプロパティやサポートされていないプロパティが含まれる場合、ランタイム例外が生成されます。

値は、一重または二重引用符で区切ることができます。また、接続文字列内では一重または二重引用符を交互に使用できます。たとえば、name="value's" や name='value's' は使用できますが、name='value's' や name=""value"" は使用できません。値または引用符内に配置されていない空白文字は無視されます。keyword=value ペアはセミコロンで区切ってください。セミコロンが値の一部である場合、セミコロンも引用符で区切ってください。エスケープシーケンスはサポートされておらず、値タイプは関係ありません。名前の大文字と小文字は区別されません。接続文字列内でプロパティ名が複数回使用されている場合、最後のプロパティ名に関連付けられている値が使用されます。

ウィンドウからユーザ ID やパスワードを取得し、これらを接続文字列に付加する場合のように、ユーザ入力に基づいて接続文字列を構成するときは注意してください。アプリケーションでは、ユーザがこれらの値に余分な接続文字列パラメータを埋め込めないようにする必要があります。

接続プーリングのデフォルト値は、true (pooling=true) です。

### 例

次の文は、ODBC データソースに接続文字列を設定し、接続を開きます。

```
SAConnection conn = new SAConnection();
conn.ConnectionString = "DSN=SQL Anywhere 17 Demo";
conn.Open();
```

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

[Open\(\) メソッド \[141 ページ\]](#)

## 1.7.17 ConnectionTimeout プロパティ

接続試行がエラーでタイムアウトするまでの秒数を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property ConnectionTimeout As Integer
```

#### C#

```
public override int ConnectionTimeout {get;}
```

### 備考

ConnectionTimeout のデフォルト値は 15 秒です。

### 例

次の文は、ConnectionTimeout の値を表示します。

```
MessageBox.Show( conn.ConnectionTimeout.ToString( ) );
```

## 1.7.18 Credential プロパティ

この接続の SACredential オブジェクトを取得または接続します。

### 構文

#### Visual Basic

```
Public Property Credential As SACredential
```

#### C#

```
public SACredential Credential {get;set;}
```

### 戻り値

この接続の SACredential オブジェクト。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

[Open\(\) メソッド \[141 ページ\]](#)

## 1.7.19 Database プロパティ

現在のデータベースの名前を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Database As String
```

#### C#

```
public override string Database {get;}
```

## 備考

接続が開かれると、SAConnection オブジェクトは現在のデータベースの名前を返します。それ以外の場合、SAConnection は DatabaseName、DBN、DataSourceName、Data Source、DSN、DatabaseFile、DBF、FileDataSourceName、FileDSN の順に接続文字列を参照します。

## 1.7.20 DataSource プロパティ

データベースサーバの名前を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property DataSource As String
```

#### C#

```
public override string DataSource {get;}
```



## 備考

接続が開かれると、SAConnection オブジェクトが ServerName サーバプロパティを返します。それ以外の場合、SAConnection オブジェクトは ServerName 接続パラメータの接続文字列を参照します。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

### 1.7.21 InitString プロパティ

接続の確立直後に実行するコマンドです。

#### 構文

##### Visual Basic

```
Public Property InitString As String
```

##### C#

```
public string InitString {get;set;}
```

## 備考

InitString は、接続が開かれた直後に実行されます。

### 1.7.22 ServerVersion プロパティ

クライアントが接続するデータベースサーバのインスタンスのバージョンが含まれている文字列を取得します。

#### 構文

##### Visual Basic

```
Public ReadOnly Overrides Property ServerVersion As String
```

##### C#

```
public override string ServerVersion {get;}
```

## 戻り値

データベースサーバのインスタンスのバージョン。

## 備考

バージョンのフォームは ##.##.#### です。最初の 2 桁はメジャーバージョン、次の 2 桁はマイナーバージョン、最後の 4 桁はリリースバージョンを示します。付加されている文字列のフォームは major.minor.build です。major と minor は 2 桁、build は 4 桁です。

## 1.7.23 State プロパティ

SAConnection オブジェクトのステータスを示します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property State As ConnectionState
```

#### C#

```
public override ConnectionState State {get;}
```

## 戻り値

System.Data.ConnectionState 列挙。

## 1.7.24 InfoMessage イベント

データベースサーバが警告または情報メッセージを返すときに発生します。

### 構文

#### Visual Basic

```
Public Event InfoMessage As SAInfoMessageEventHandler
```

#### C#

```
public SAInfoMessageEventHandler InfoMessage;
```

## 備考

イベントハンドラは、このイベントに関するデータが含まれるタイプ `SALnfoMessageEventArgs` の引数を受け取ります。次の `SALnfoMessageEventArgs` プロパティは、このイベントに固有の情報 (`NativeError`、`Errors`、`Message`、`MessageType`、`Source`) を提供します。

詳細については、.NET Framework のマニュアルの `OleDbConnection.InfoMessage` イベントを参照してください。

## 1.7.25 StateChange イベント

`SACConnection` オブジェクトのステータスが変わると発生します。

### 構文

#### Visual Basic

```
Public Event StateChange As StateChangeEventHandler
```

#### C#

```
public override StateChangeEventHandler StateChange;
```

## 備考

イベントハンドラは、このイベントに関するデータが含まれるタイプ `StateChangeEventArgs` の引数を受け取ります。次の `StateChangeEventArgs` プロパティは、このイベントに固有の情報 (`CurrentState` および `OriginalState`) を提供します。

詳細については、.NET Framework のマニュアルの `OleDbConnection.StateChange` イベントを参照してください。

## 1.8 SACConnectionStringBuilder クラス

`SACConnection` クラスが使用する接続文字列の内容を作成および管理する簡単な方法を提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SACConnectionStringBuilder Inherits  
SACConnectionStringBuilderBase
```

#### C#

```
public sealed class SACConnectionStringBuilder : SACConnectionStringBuilderBase
```

## メンバー

SACConnectionStringBuilder のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SACConnectionStringBuilder [154 ページ]</a>	SACConnectionStringBuilder クラスの新しいインスタンスを初期化します。

### プロパティ

変更子とタイプ	プロパティ	説明
public string	<a href="#">AppInfo [155 ページ]</a>	AppInfo 接続プロパティを取得または設定します。
public string	<a href="#">AutoStart [156 ページ]</a>	AutoStart 接続プロパティを取得または設定します。
public string	<a href="#">AutoStop [156 ページ]</a>	AutoStop 接続プロパティを取得または設定します。
public string	<a href="#">Charset [156 ページ]</a>	Charset 接続プロパティを取得または設定します。
public int	<a href="#">CommBufferSize [157 ページ]</a>	CommBufferSize 接続プロパティを取得または設定します。
public string	<a href="#">CommLinks [157 ページ]</a>	CommLinks 接続プロパティを取得または設定します。
public string	<a href="#">Compress [157 ページ]</a>	Compress 接続プロパティを取得または設定します。
public int	<a href="#">CompressionThreshold [158 ページ]</a>	CompressionThreshold 接続プロパティを取得または設定します。
public int	<a href="#">ConnectionLifetime [158 ページ]</a>	ConnectionLifetime 接続プロパティを取得または設定します。
public string	<a href="#">ConnectionName [158 ページ]</a>	ConnectionName 接続プロパティを取得または設定します。
public string	<a href="#">ConnectionPool [159 ページ]</a>	ConnectionPool 接続プロパティを取得または設定します。
public bool	<a href="#">ConnectionReset [159 ページ]</a>	ConnectionReset 接続プロパティを取得または設定します。
public int	<a href="#">ConnectionTimeout [159 ページ]</a>	ConnectionTimeout 接続プロパティを取得または設定します。
public string	<a href="#">DatabaseFile [160 ページ]</a>	DatabaseFile 接続プロパティを取得または設定します。
public string	<a href="#">DatabaseKey [160 ページ]</a>	DatabaseKey 接続プロパティを取得または設定します。
public string	<a href="#">DatabaseName [160 ページ]</a>	DatabaseName 接続プロパティを取得または設定します。

変更子とタイプ	プロパティ	説明
public string	<a href="#">DatabaseSwitches [161 ページ]</a>	DatabaseSwitches 接続プロパティを取得または設定します。
public string	<a href="#">DataSourceName [161 ページ]</a>	DataSourceName 接続プロパティを取得または設定します。
public string	<a href="#">DisableMultiRowFetch [161 ページ]</a>	DisableMultiRowFetch 接続プロパティを取得または設定します。
public string	<a href="#">Elevate [162 ページ]</a>	Elevate 接続プロパティを取得または設定します。
public string	<a href="#">EncryptedPassword [162 ページ]</a>	EncryptedPassword 接続プロパティを取得または設定します。
public string	<a href="#">Encryption [162 ページ]</a>	Encryption 接続プロパティを取得または設定します。
public bool	<a href="#">Enlist [163 ページ]</a>	Enlist 接続プロパティを取得または設定します。
public string	<a href="#">FileDataSourceName [163 ページ]</a>	FileDataSourceName 接続プロパティを取得または設定します。
public string	<a href="#">ForceStart [163 ページ]</a>	ForceStart 接続プロパティを取得または設定します。
public string	<a href="#">Host [164 ページ]</a>	Host 接続プロパティを取得または設定します。
public int	<a href="#">IdleTimeout [164 ページ]</a>	IdleTimeout 接続プロパティを取得または設定します。
public string	<a href="#">InitString [164 ページ]</a>	InitString 接続プロパティを取得または設定します。
public string	<a href="#">Integrated [165 ページ]</a>	Integrated 接続プロパティを取得または設定します。
public string	<a href="#">Kerberos [165 ページ]</a>	Kerberos 接続プロパティを取得または設定します。
public string	<a href="#">Language [165 ページ]</a>	Language 接続プロパティを取得または設定します。
public string	<a href="#">LazyClose [166 ページ]</a>	LazyClose 接続プロパティを取得または設定します。
public int	<a href="#">LivenessTimeout [166 ページ]</a>	LivenessTimeout 接続プロパティを取得または設定します。
public string	<a href="#">LogFile [166 ページ]</a>	LogFile 接続プロパティを取得または設定します。
public int	<a href="#">MaxPoolSize [167 ページ]</a>	MaxPoolSize 接続プロパティを取得または設定します。
public int	<a href="#">MinPoolSize [167 ページ]</a>	MinPoolSize 接続プロパティを取得または設定します。

変更子とタイプ	プロパティ	説明
public string	<a href="#">NewPassword [167 ページ]</a>	NewPassword 接続プロパティを取得または設定します。
public string	<a href="#">NodeType [168 ページ]</a>	NodeType 接続プロパティを取得または設定します。
public string	<a href="#">Password [168 ページ]</a>	Password 接続プロパティを取得または設定します。
public bool	<a href="#">PersistSecurityInfo [168 ページ]</a>	PersistSecurityInfo 接続プロパティを取得または設定します。
public bool	<a href="#">Pooling [169 ページ]</a>	Pooling 接続プロパティを取得または設定します。
public int	<a href="#">PrefetchBuffer [169 ページ]</a>	PrefetchBuffer 接続プロパティを取得または設定します。
public int	<a href="#">PrefetchRows [169 ページ]</a>	PrefetchRows 接続プロパティを取得または設定します。
public int	<a href="#">RetryConnectionTimeout [170 ページ]</a>	RetryConnectionTimeout 接続プロパティを取得または設定します。
public string	<a href="#">ServerName [170 ページ]</a>	ServerName 接続プロパティを取得または設定します。
public string	<a href="#">StartLine [170 ページ]</a>	StartLine 接続プロパティを取得または設定します。
public string	<a href="#">Unconditional [171 ページ]</a>	Unconditional 接続プロパティを取得または設定します。
public string	<a href="#">UserID [171 ページ]</a>	UserID 接続プロパティを取得または設定します。

#### SAConnectionStringBuilderBase から継承されたメンバー

変更子とタイプ	メンバー	説明
public override bool	<a href="#">ContainsKey(string) [173 ページ]</a>	SAConnectionStringBuilder オブジェクトに特定のキーワードが含まれているかどうかを判断します。
public string	<a href="#">GetKeyword(string) [174 ページ]</a>	指定された SAConnectionStringBuilder プロパティのキーワードを取得します。
public bool	<a href="#">GetUseLongNameAsKeyword() [174 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。
public override ICollection	<a href="#">Keys [178 ページ]</a>	SAConnectionStringBuilder のキーが含まれた System.Collections.ICollection を取得します。
public override bool	<a href="#">Remove(string) [175 ページ]</a>	指定されたキーが設定されたエントリを SAConnectionStringBuilder インスタンスから削除します。

変数とタイプ	メンバー	説明
public void	<a href="#">SetUseLongNameAsKeyword(bool) [176 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。
public override bool	<a href="#">ShouldSerialize(string) [176 ページ]</a>	指定されたキーが、この SAConnectionStringBuilder インスタンスに存在するかどうかを示します。
public override object	<a href="#">this[string keyword] [178 ページ]</a>	接続キーワードの値を取得または設定します。
public override bool	<a href="#">TryGetValue(string, out object) [177 ページ]</a>	入力されたキーに対応する値を、この SAConnectionStringBuilder から取り出します。

## 備考

SAConnectionStringBuilder クラスは SAConnectionStringBuilderBase を継承し、SAConnectionStringBuilderBase は DbConnectionStringBuilder を継承しています。

接続パラメータのリストについては、「.NET 接続パラメータ」と「接続パラメータ」を参照してください。

このセクションの内容:

### [SAConnectionStringBuilder コンストラクタ \[154 ページ\]](#)

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

### [AppInfo プロパティ \[155 ページ\]](#)

AppInfo 接続プロパティを取得または設定します。

### [AutoStart プロパティ \[156 ページ\]](#)

AutoStart 接続プロパティを取得または設定します。

### [AutoStop プロパティ \[156 ページ\]](#)

AutoStop 接続プロパティを取得または設定します。

### [Charset プロパティ \[156 ページ\]](#)

Charset 接続プロパティを取得または設定します。

### [CommBufferSize プロパティ \[157 ページ\]](#)

CommBufferSize 接続プロパティを取得または設定します。

### [CommLinks プロパティ \[157 ページ\]](#)

CommLinks 接続プロパティを取得または設定します。

### [Compress プロパティ \[157 ページ\]](#)

Compress 接続プロパティを取得または設定します。

### [CompressionThreshold プロパティ \[158 ページ\]](#)

CompressionThreshold 接続プロパティを取得または設定します。

### [ConnectionLifetime プロパティ \[158 ページ\]](#)

ConnectionLifetime 接続プロパティを取得または設定します。

#### [ConnectionString プロパティ \[158 ページ\]](#)

ConnectionString 接続プロパティを取得または設定します。

#### [ConnectionPool プロパティ \[159 ページ\]](#)

ConnectionPool 接続プロパティを取得または設定します。

#### [ConnectionReset プロパティ \[159 ページ\]](#)

ConnectionReset 接続プロパティを取得または設定します。

#### [ConnectionTimeout プロパティ \[159 ページ\]](#)

ConnectionTimeout 接続プロパティを取得または設定します。

#### [DatabaseFile プロパティ \[160 ページ\]](#)

DatabaseFile 接続プロパティを取得または設定します。

#### [DatabaseKey プロパティ \[160 ページ\]](#)

DatabaseKey 接続プロパティを取得または設定します。

#### [DatabaseName プロパティ \[160 ページ\]](#)

DatabaseName 接続プロパティを取得または設定します。

#### [DatabaseSwitches プロパティ \[161 ページ\]](#)

DatabaseSwitches 接続プロパティを取得または設定します。

#### [DataSourceName プロパティ \[161 ページ\]](#)

DataSourceName 接続プロパティを取得または設定します。

#### [DisableMultiRowFetch プロパティ \[161 ページ\]](#)

DisableMultiRowFetch 接続プロパティを取得または設定します。

#### [Elevate プロパティ \[162 ページ\]](#)

Elevate 接続プロパティを取得または設定します。

#### [EncryptedPassword プロパティ \[162 ページ\]](#)

EncryptedPassword 接続プロパティを取得または設定します。

#### [Encryption プロパティ \[162 ページ\]](#)

Encryption 接続プロパティを取得または設定します。

#### [Enlist プロパティ \[163 ページ\]](#)

Enlist 接続プロパティを取得または設定します。

#### [FileDataSourceName プロパティ \[163 ページ\]](#)

FileDataSourceName 接続プロパティを取得または設定します。

#### [ForceStart プロパティ \[163 ページ\]](#)

ForceStart 接続プロパティを取得または設定します。

#### [Host プロパティ \[164 ページ\]](#)

Host 接続プロパティを取得または設定します。

#### [IdleTimeout プロパティ \[164 ページ\]](#)

IdleTimeout 接続プロパティを取得または設定します。

#### [InitString プロパティ \[164 ページ\]](#)

InitString 接続プロパティを取得または設定します。

#### [Integrated プロパティ \[165 ページ\]](#)



Integrated 接続プロパティを取得または設定します。

[Kerberos プロパティ \[165 ページ\]](#)

Kerberos 接続プロパティを取得または設定します。

[Language プロパティ \[165 ページ\]](#)

Language 接続プロパティを取得または設定します。

[LazyClose プロパティ \[166 ページ\]](#)

LazyClose 接続プロパティを取得または設定します。

[LivenessTimeout プロパティ \[166 ページ\]](#)

LivenessTimeout 接続プロパティを取得または設定します。

[LogFile プロパティ \[166 ページ\]](#)

LogFile 接続プロパティを取得または設定します。

[MaxPoolSize プロパティ \[167 ページ\]](#)

MaxPoolSize 接続プロパティを取得または設定します。

[MinPoolSize プロパティ \[167 ページ\]](#)

MinPoolSize 接続プロパティを取得または設定します。

[NewPassword プロパティ \[167 ページ\]](#)

NewPassword 接続プロパティを取得または設定します。

[NodeType プロパティ \[168 ページ\]](#)

NodeType 接続プロパティを取得または設定します。

[Password プロパティ \[168 ページ\]](#)

Password 接続プロパティを取得または設定します。

[PersistSecurityInfo プロパティ \[168 ページ\]](#)

PersistSecurityInfo 接続プロパティを取得または設定します。

[Pooling プロパティ \[169 ページ\]](#)

Pooling 接続プロパティを取得または設定します。

[PrefetchBuffer プロパティ \[169 ページ\]](#)

PrefetchBuffer 接続プロパティを取得または設定します。

[PrefetchRows プロパティ \[169 ページ\]](#)

PrefetchRows 接続プロパティを取得または設定します。

[RetryConnectionTimeout プロパティ \[170 ページ\]](#)

RetryConnectionTimeout 接続プロパティを取得または設定します。

[ServerName プロパティ \[170 ページ\]](#)

ServerName 接続プロパティを取得または設定します。

[StartLine プロパティ \[170 ページ\]](#)

StartLine 接続プロパティを取得または設定します。

[Unconditional プロパティ \[171 ページ\]](#)

Unconditional 接続プロパティを取得または設定します。

[UserID プロパティ \[171 ページ\]](#)

UserID 接続プロパティを取得または設定します。

## 1.8.1 SAConnectionStringBuilder コンストラクタ

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public	<a href="#">SAConnectionStringBuilder()</a> [154 ページ]	SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。
public	<a href="#">SAConnectionStringBuilder(string)</a> [155 ページ]	SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

このセクションの内容:

[SAConnectionStringBuilder\(\) コンストラクタ \[154 ページ\]](#)

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

[SAConnectionStringBuilder\(string\) コンストラクタ \[155 ページ\]](#)

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

### 1.8.1.1 SAConnectionStringBuilder() コンストラクタ

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SAConnectionStringBuilder ()
```

##### C#

```
public SAConnectionStringBuilder ()
```

## 1.8.1.2 SAConnectionStringBuilder(string) コンストラクタ

SAConnectionStringBuilder クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Public Sub SAConnectionStringBuilder (ByVal connectionString As String)
```

#### C#

```
public SAConnectionStringBuilder (string connectionString)
```

### パラメータ

**connectionString** オブジェクトの内部接続情報の基礎。keyword=value のペアに解析された情報です。接続パラメータのリストについては、「[接続パラメータ](#)」を参照してください。

### 例

次の文は、hr という名前のデータベースサーバ上で動作している policies という名前のデータベースへの接続について SAConnectionString オブジェクトを初期化します。接続では、ユーザ ID admin とパスワード money を使用します。

```
SAConnectionStringBuilder conn = new SAConnectionStringBuilder(  
    "UID=admin;PWD=money;SERVER=hr;DBN=policies" );
```

## 1.8.2 AppInfo プロパティ

AppInfo 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property AppInfo As String
```

#### C#

```
public string AppInfo {get;set;}
```

## 1.8.3 AutoStart プロパティ

AutoStart 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property AutoStart As String
```

#### C#

```
public string AutoStart {get;set;}
```

## 1.8.4 AutoStop プロパティ

AutoStop 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property AutoStop As String
```

#### C#

```
public string AutoStop {get;set;}
```

## 1.8.5 Charset プロパティ

Charset 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Charset As String
```

#### C#

```
public string Charset {get;set;}
```

## 1.8.6 CommBufferSize プロパティ

CommBufferSize 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property CommBufferSize As Integer
```

#### C#

```
public int CommBufferSize {get;set;}
```

## 1.8.7 CommLinks プロパティ

CommLinks 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property CommLinks As String
```

#### C#

```
public string CommLinks {get;set;}
```

## 1.8.8 Compress プロパティ

Compress 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Compress As String
```

#### C#

```
public string Compress {get;set;}
```

## 1.8.9 CompressionThreshold プロパティ

CompressionThreshold 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property CompressionThreshold As Integer
```

#### C#

```
public int CompressionThreshold {get;set;}
```

## 1.8.10 ConnectionLifetime プロパティ

ConnectionLifetime 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionLifetime As Integer
```

#### C#

```
public int ConnectionLifetime {get;set;}
```

## 1.8.11 ConnectionName プロパティ

ConnectionName 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionName As String
```

#### C#

```
public string ConnectionName {get;set;}
```

## 1.8.12 ConnectionPool プロパティ

ConnectionPool 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionPool As String
```

#### C#

```
public string ConnectionPool {get;set;}
```

## 1.8.13 ConnectionReset プロパティ

ConnectionReset 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionReset As Boolean
```

#### C#

```
public bool ConnectionReset {get;set;}
```

## 1.8.14 ConnectionTimeout プロパティ

ConnectionTimeout 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ConnectionTimeout As Integer
```

#### C#

```
public int ConnectionTimeout {get;set;}
```

### 例

次の文は、ConnectionTimeout プロパティの値を表示します。

```
MessageBox.Show( connString.ConnectionTimeout.ToString() );
```

## 1.8.15 DatabaseFile プロパティ

DatabaseFile 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DatabaseFile As String
```

#### C#

```
public string DatabaseFile {get;set;}
```

## 1.8.16 DatabaseKey プロパティ

DatabaseKey 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DatabaseKey As String
```

#### C#

```
public string DatabaseKey {get;set;}
```

## 1.8.17 DatabaseName プロパティ

DatabaseName 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DatabaseName As String
```

#### C#

```
public string DatabaseName {get;set;}
```



## 1.8.18 DatabaseSwitches プロパティ

DatabaseSwitches 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DatabaseSwitches As String
```

#### C#

```
public string DatabaseSwitches {get;set;}
```

## 1.8.19 DataSourceName プロパティ

DataSourceName 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DataSourceName As String
```

#### C#

```
public string DataSourceName {get;set;}
```

## 1.8.20 DisableMultiRowFetch プロパティ

DisableMultiRowFetch 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property DisableMultiRowFetch As String
```

#### C#

```
public string DisableMultiRowFetch {get;set;}
```

## 1.8.21 Elevate プロパティ

Elevate 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Elevate As String
```

#### C#

```
public string Elevate {get;set;}
```

## 1.8.22 EncryptedPassword プロパティ

EncryptedPassword 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property EncryptedPassword As String
```

#### C#

```
public string EncryptedPassword {get;set;}
```

## 1.8.23 Encryption プロパティ

Encryption 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Encryption As String
```

#### C#

```
public string Encryption {get;set;}
```

## 1.8.24 Enlist プロパティ

Enlist 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Enlist As Boolean
```

#### C#

```
public bool Enlist {get;set;}
```

## 1.8.25 FileDataSourceName プロパティ

FileDataSourceName 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property FileDataSourceName As String
```

#### C#

```
public string FileDataSourceName {get;set;}
```

## 1.8.26 ForceStart プロパティ

ForceStart 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ForceStart As String
```

#### C#

```
public string ForceStart {get;set;}
```

## 1.8.27 Host プロパティ

Host 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Host As String
```

#### C#

```
public string Host {get;set;}
```

## 1.8.28 IdleTimeout プロパティ

IdleTimeout 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property IdleTimeout As Integer
```

#### C#

```
public int IdleTimeout {get;set;}
```

## 1.8.29 InitString プロパティ

InitString 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property InitString As String
```

#### C#

```
public string InitString {get;set;}
```

## 1.8.30 Integrated プロパティ

Integrated 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Integrated As String
```

#### C#

```
public string Integrated {get;set;}
```

## 1.8.31 Kerberos プロパティ

Kerberos 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Kerberos As String
```

#### C#

```
public string Kerberos {get;set;}
```

## 1.8.32 Language プロパティ

Language 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Language As String
```

#### C#

```
public string Language {get;set;}
```

## 1.8.33 LazyClose プロパティ

LazyClose 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property LazyClose As String
```

#### C#

```
public string LazyClose {get;set;}
```

## 1.8.34 LivenessTimeout プロパティ

LivenessTimeout 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property LivenessTimeout As Integer
```

#### C#

```
public int LivenessTimeout {get;set;}
```

## 1.8.35 LogFile プロパティ

LogFile 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property LogFile As String
```

#### C#

```
public string LogFile {get;set;}
```

## 1.8.36 MaxPoolSize プロパティ

MaxPoolSize 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property MaxPoolSize As Integer
```

#### C#

```
public int MaxPoolSize {get;set;}
```

## 1.8.37 MinPoolSize プロパティ

MinPoolSize 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property MinPoolSize As Integer
```

#### C#

```
public int MinPoolSize {get;set;}
```

## 1.8.38 NewPassword プロパティ

NewPassword 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property NewPassword As String
```

#### C#

```
public string NewPassword {get;set;}
```

## 1.8.39 NodeType プロパティ

NodeType 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property NodeType As String
```

#### C#

```
public string NodeType {get;set;}
```

## 1.8.40 Password プロパティ

Password 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Password As String
```

#### C#

```
public string Password {get;set;}
```

## 1.8.41 PersistSecurityInfo プロパティ

PersistSecurityInfo 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property PersistSecurityInfo As Boolean
```

#### C#

```
public bool PersistSecurityInfo {get;set;}
```



## 1.8.42 Pooling プロパティ

Pooling 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Pooling As Boolean
```

#### C#

```
public bool Pooling {get;set;}
```

## 1.8.43 PrefetchBuffer プロパティ

PrefetchBuffer 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property PrefetchBuffer As Integer
```

#### C#

```
public int PrefetchBuffer {get;set;}
```

## 1.8.44 PrefetchRows プロパティ

PrefetchRows 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property PrefetchRows As Integer
```

#### C#

```
public int PrefetchRows {get;set;}
```

備考

デフォルト値は 200 です。

## 1.8.45 RetryConnectionTimeout プロパティ

RetryConnectionTimeout 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property RetryConnectionTimeout As Integer
```

#### C#

```
public int RetryConnectionTimeout {get;set;}
```

## 1.8.46 ServerName プロパティ

ServerName 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ServerName As String
```

#### C#

```
public string ServerName {get;set;}
```

## 1.8.47 StartLine プロパティ

StartLine 接続プロパティを取得または設定します。

### 構文

#### Visual Basic

```
Public Property StartLine As String
```

C#

```
public string StartLine {get;set;}
```

## 1.8.48 Unconditional プロパティ

Unconditional 接続プロパティを取得または設定します。

📄 構文

Visual Basic

```
Public Property Unconditional As String
```

C#

```
public string Unconditional {get;set;}
```

## 1.8.49 UserID プロパティ

UserID 接続プロパティを取得または設定します。

📄 構文

Visual Basic

```
Public Property UserID As String
```

C#

```
public string UserID {get;set;}
```

## 1.9 SAConnectionStringBuilderBase クラス

SAConnectionStringBuilder クラスの基本クラスです。

📄 構文

Visual Basic

```
Public MustInherit Class SAConnectionStringBuilderBase Inherits  
System.Data.Common.DbConnectionStringBuilder
```

## C#

```
public abstract class SAConnectionStringBuilderBase :  
System.Data.Common.DbConnectionStringBuilder
```

## メンバー

SAConnectionStringBuilderBase のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変数とタイプ	メソッド	説明
public override bool	<a href="#">ContainsKey(string) [173 ページ]</a>	SAConnectionStringBuilder オブジェクトに特定のキーワードが含まれているかどうかを判断します。
public string	<a href="#">GetKeyword(string) [174 ページ]</a>	指定された SAConnectionStringBuilder プロパティのキーワードを取得します。
public bool	<a href="#">GetUseLongNameAsKeyword() [174 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。
public override bool	<a href="#">Remove(string) [175 ページ]</a>	指定されたキーが設定されたエントリを SAConnectionStringBuilder インスタンスから削除します。
public void	<a href="#">SetUseLongNameAsKeyword(bool) [176 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。
public override bool	<a href="#">ShouldSerialize(string) [176 ページ]</a>	指定されたキーが、この SAConnectionStringBuilder インスタンスに存在するかどうかを示します。
public override bool	<a href="#">TryGetValue(string, out object) [177 ページ]</a>	入力されたキーに対応する値を、この SAConnectionStringBuilder から取り出します。

### プロパティ

変数とタイプ	プロパティ	説明
public override ICollection	<a href="#">Keys [178 ページ]</a>	SAConnectionStringBuilder のキーが含まれた System.Collections.ICollection を取得します。
public override object	<a href="#">this[string keyword] [178 ページ]</a>	接続キーワードの値を取得または設定します。

このセクションの内容:

#### [ContainsKey\(string\) メソッド \[173 ページ\]](#)

SAConnectionStringBuilder オブジェクトに特定のキーワードが含まれているかどうかを判断します。

#### [GetKeyword\(string\) メソッド \[174 ページ\]](#)

指定された SAConnectionStringBuilder プロパティのキーワードを取得します。

#### [GetUseLongNameAsKeyword\(\) メソッド \[174 ページ\]](#)

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。

#### [Remove\(string\) メソッド \[175 ページ\]](#)

指定されたキーが設定されたエントリを SAConnectionStringBuilder インスタンスから削除します。

#### [SetUseLongNameAsKeyword\(bool\) メソッド \[176 ページ\]](#)

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。

#### [ShouldSerialize\(string\) メソッド \[176 ページ\]](#)

指定されたキーが、この SAConnectionStringBuilder インスタンスに存在するかどうかを示します。

#### [TryGetValue\(string, out object\) メソッド \[177 ページ\]](#)

入力されたキーに対応する値を、この SAConnectionStringBuilder から取り出します。

#### [Keys プロパティ \[178 ページ\]](#)

SAConnectionStringBuilder のキーが含まれた System.Collections.ICollection を取得します。

#### [this\[string keyword\] プロパティ \[178 ページ\]](#)

接続キーワードの値を取得または設定します。

## 1.9.1 ContainsKey(string) メソッド

SAConnectionStringBuilder オブジェクトに特定のキーワードが含まれているかどうかを判断します。

### 構文

#### Visual Basic

```
Public Overrides Function ContainsKey (ByVal keyword As String) As Boolean
```

#### C#

```
public override bool ContainsKey (string keyword)
```

### パラメータ

**keyword** SAConnectionStringBuilder 内で検索するキーワード。

### 戻り値

keyword に関連する値が設定されている場合は true、設定されていない場合は false。



例

次の文は、SAConnectionStringBuilder オブジェクトに UserID キーワードが含まれているかどうかを判断します。

```
connectString.ContainsKey("UserID")
```

## 1.9.2 GetKeyword(string) メソッド

指定された SAConnectionStringBuilder プロパティのキーワードを取得します。



構文

### Visual Basic

```
Public Function GetKeyword (ByVal propName As String) As String
```

### C#

```
public string GetKeyword (string propName)
```

### パラメータ

**propName** SAConnectionStringBuilder プロパティの名前。

### 戻り値

指定された SAConnectionStringBuilder プロパティのキーワード。

## 1.9.3 GetUseLongNameAsKeyword() メソッド

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を取得します。



構文

### Visual Basic

```
Public Function GetUseLongNameAsKeyword () As Boolean
```

### C#

```
public bool GetUseLongNameAsKeyword ()
```

## 戻り値

長い接続パラメータ名を使用して接続文字列を作成する場合は true、それ以外の場合は false。

## 備考

接続パラメータの名前には、長い名前と短い名前の 2 種類があります。たとえば、接続文字列に ODBC データソースの名前を指定する場合は、DataSourceName と DSN のいずれかを使用します。デフォルトでは、長い接続パラメータ名を使用して接続文字列を作成します。

## 関連情報

[SetUseLongNameAsKeyword\(bool\) メソッド \[176 ページ\]](#)

## 1.9.4 Remove(string) メソッド

指定されたキーが設定されたエントリを SAConnectionStringBuilder インスタンスから削除します。

### 構文

#### Visual Basic

```
Public Overrides Function Remove (ByVal keyword As String) As Boolean
```

#### C#

```
public override bool Remove (string keyword)
```

## パラメータ

**keyword** この SAConnectionStringBuilder 内の接続文字列から削除するキー/値のペアのキー。

## 戻り値

接続文字列内のキーが削除された場合は true、キーが存在しなかった場合は false。

## 1.9.5 SetUseLongNameAsKeyword(bool) メソッド

長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。

### 構文

#### Visual Basic

```
Public Sub SetUseLongNameAsKeyword (ByVal useLongNameAsKeyword As Boolean)
```

#### C#

```
public void SetUseLongNameAsKeyword (bool useLongNameAsKeyword)
```

### パラメータ

**useLongNameAsKeyword** 長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値。

### 備考

デフォルトでは、長い接続パラメータ名が使用されます。

### 関連情報

[GetUseLongNameAsKeyword\(\) メソッド \[174 ページ\]](#)

## 1.9.6 ShouldSerialize(string) メソッド

指定されたキーが、この SAConnectionStringBuilder インスタンスに存在するかどうかを示します。

### 構文

#### Visual Basic

```
Public Overrides Function ShouldSerialize (ByVal keyword As String) As Boolean
```

#### C#

```
public override bool ShouldSerialize (string keyword)
```



## パラメータ

**keyword** SAConnectionStringBuilder 内で検索するキー。

## 戻り値

SAConnectionStringBuilder に指定されたキーが設定されたエントリが含まれている場合は true、それ以外の場合は false。

## 1.9.7 TryGetValue(string, out object) メソッド

入力されたキーに対応する値を、この SAConnectionStringBuilder から取り出します。

### 構文

#### Visual Basic

```
Public Overrides Function TryGetValue (
    ByVal keyword As String,
    ByVal value As Object
) As Boolean
```

#### C#

```
public override bool TryGetValue (
    string keyword,
    out object value
)
```

## パラメータ

**keyword** 取り出す項目のキー。

**value** キーワードに対応する値。

## 戻り値

キーワードが接続文字列内にある場合は true、それ以外は false。

## 1.9.8 Keys プロパティ

SACConnectionStringBuilder のキーが含まれた System.Collections.ICollection を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Keys As ICollection
```

#### C#

```
public override ICollection Keys {get;}
```

### 戻り値

SACConnectionStringBuilder のキーが含まれた System.Collections.ICollection。

## 1.9.9 this[string keyword] プロパティ

接続キーワードの値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property Item (ByVal keyword As String) As Object
```

#### C#

```
public override object this[string keyword] {get;set;}
```

### 備考

指定された接続キーワードの値を表すオブジェクト。

キーワードまたは型が無効の場合は、例外が発生します。パラメータ値は大文字と小文字を区別します。

値の設定で NULL が渡されると、値がクリアされます。

## 1.10 SACredential クラス

SACredential は、データベースサーバ認証を使用してログインのパスワードを指定する、よりセキュアな方法を提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SACredential
```

#### C#

```
public sealed class SACredential
```

### メンバー

SACredential のすべてのメンバー (継承されたメンバーも含みます) を次に示します。

#### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SACredential(string, SecureString) [180 ページ]</a>	SACredential オブジェクトを初期化します。

#### プロパティ

変更子とタイプ	プロパティ	説明
public SecureString	<a href="#">Password [180 ページ]</a>	SACredential オブジェクトのパスワードを返します。
public string	<a href="#">UserId [181 ページ]</a>	SACredential オブジェクトのユーザ ID を返します。

### 備考

SACredential は、データベースサーバ認証で使用される ユーザ ID とパスワードで構成されています。SACredential オブジェクトのパスワードは、System.Security.SecureString 型です。SACredential を継承することはできません。

このセクションの内容:

[SACredential\(string, SecureString\) コンストラクタ \[180 ページ\]](#)

SACredential オブジェクトを初期化します。

[Password プロパティ \[180 ページ\]](#)

SACredential オブジェクトのパスワードを返します。

[UserId プロパティ \[181 ページ\]](#)

SACredential オブジェクトのユーザ ID を返します。

## 1.10.1 SACredential(string, SecureString) コンストラクタ

SACredential オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SACredential (
    ByVal userId As String,
    ByVal password As SecureString
)
```

#### C#

```
public SACredential (
    string userId,
    SecureString password
)
```

### パラメータ

**userId** ユーザ ID。

**password** パスワード。読み取り専用としてマークされた System.Security.SecureString 値。読み書き System.Security.SecureString パラメータを渡すと、System.ArgumentException が発行されます。

### 関連情報

[SACredential クラス \[118 ページ\]](#)

## 1.10.2 Password プロパティ

SACredential オブジェクトのパスワードを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Password As SecureString
```

**C#**

```
public SecureString Password {get;}
```

戻り値

SACredential オブジェクトのパスワード。

### 1.10.3 UserId プロパティ

SACredential オブジェクトのユーザ ID を返します。

 構文

**Visual Basic**

```
Public ReadOnly Property UserId As String
```

**C#**

```
public string UserId {get;}
```

戻り値

SACredential オブジェクトのユーザ ID。

## 1.11 SDataAdapter クラス

System.Data.DataSet に入力したりデータベースを更新したりするために使用する一連のコマンドとデータベース接続を表します。

 構文

**Visual Basic**

```
Public NotInheritable Class SDataAdapter Inherits  
System.Data.Common.DbDataAdapter Implements System.ICloneable
```

## C#

```
public sealed class SDataAdapter : System.Data.Common.DbDataAdapter,  
System.ICloneable
```

## メンバー

SDataAdapter のすべてのメンバー (継承されたメンバーも含まれます) を以下に示します。

### コンストラクタ

変数とタイプ	コンストラクタ	説明
public	<a href="#">SDataAdapter [185 ページ]</a>	SDataAdapter オブジェクトを初期化します。

### メソッド

変数とタイプ	メソッド	説明
protected override void	<a href="#">ClearBatch() [189 ページ]</a>	バッチからすべての SCommand オブジェクトを削除します。
protected override RowUpdatedEventArgs	<a href="#">CreateRowUpdatedEvent(DataRow, IDbCommand, StatementType, DataTableMapping) [189 ページ]</a>	System.Data.Common.RowUpdatedEventArgs クラスの新しいインスタンスを初期化します。
protected override RowUpdatingEventArgs	<a href="#">CreateRowUpdatingEvent(DataRow, IDbCommand, StatementType, DataTableMapping) [190 ページ]</a>	System.Data.Common.RowUpdatingEventArgs クラスの新しいインスタンスを初期化します。
protected override void	<a href="#">Dispose(bool) [191 ページ]</a>	SDataAdapter オブジェクトによって使用されるアンマネージリソースを解放するほか、必要に応じてマネージリソースを解放します。
protected override int	<a href="#">Fill [192 ページ]</a>	データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。
protected override DataTable[]	<a href="#">FillSchema [195 ページ]</a>	System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。
public new SAParameter[]	<a href="#">GetFillParameters() [198 ページ]</a>	SELECT 文の実行時に自分が設定したパラメータを返します。
protected override void	<a href="#">InitializeBatching() [198 ページ]</a>	SDataAdapter オブジェクトのためのバッチを初期化します。
protected override void	<a href="#">OnRowUpdated(RowUpdatedEventArgs) [199 ページ]</a>	.NET Framework データプロバイダの RowUpdated イベントを発生します。

変更子とタイプ	メソッド	説明
protected override void	<a href="#">OnRowUpdating(RowUpdatingEventArgs) [199 ページ]</a>	.NET Framework データプロバイダの RowUpdating イベントを発生します。
protected override void	<a href="#">TerminateBatching() [200 ページ]</a>	SADataAdapter オブジェクトのためのバッチを終了します。
protected override int	<a href="#">Update(DataRow[], DataTableMapping) [200 ページ]</a>	DataSet に加えられた変更内容で、データベース内のテーブルを更新します。

## プロパティ

変更子とタイプ	プロパティ	説明
public new SACommand	<a href="#">DeleteCommand [201 ページ]</a>	DataSet で削除されたローに該当するデータベース内のローを削除するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。
public new SACommand	<a href="#">InsertCommand [202 ページ]</a>	DataSet に挿入されたローに該当するローをデータベースに追加するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。
public new SACommand	<a href="#">SelectCommand [202 ページ]</a>	Fill または FillSchema の実行時に、DataSet にコピーするための結果セットをデータベースから取得するために使用される SACommand を指定します。
public new DataTableMappingCollection	<a href="#">TableMappings [203 ページ]</a>	ソーステーブルと DataTable 間のマスタマッピングを提供するコレクションを指定します。
public override int	<a href="#">UpdateBatchSize [204 ページ]</a>	データベースサーバとの各往復の中で処理されるローの数を取得または設定します。
public new SACommand	<a href="#">UpdateCommand [204 ページ]</a>	DataSet で更新されたローに該当するデータベース内のローを更新するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

## イベント

変更子とタイプ	イベント	説明
public SRowUpdatedEventHandler	<a href="#">RowUpdated [205 ページ]</a>	データソースに対してコマンドが実行された後の更新時に発生します。
public SRowUpdatingEventHandler	<a href="#">RowUpdating [205 ページ]</a>	データソースに対してコマンドが実行される前の更新時に発生します。

## 備考

System.Data.DataSet は、データをオフラインで処理するための方法を提供します。SDataAdapter は、DataSet を一連の SQL 文に関連付けるためのメソッドを提供します。

**実装:** IDbDataAdapter、IDataAdapter、ICloneable

詳細については、「データのアクセスとデータの操作」を参照してください。

このセクションの内容:

### [SDataAdapter コンストラクタ \[185 ページ\]](#)

SDataAdapter オブジェクトを初期化します。

### [ClearBatch\(\) メソッド \[189 ページ\]](#)

バッチからすべての SACommand オブジェクトを削除します。

### [CreateRowUpdatedEvent\(DataRow, IDbCommand, StatementType, DataTableMapping\) メソッド \[189 ページ\]](#)

System.Data.Common.RowUpdatedEventArgs クラスの新しいインスタンスを初期化します。

### [CreateRowUpdatingEvent\(DataRow, IDbCommand, StatementType, DataTableMapping\) メソッド \[190 ページ\]](#)

System.Data.Common.RowUpdatingEventArgs クラスの新しいインスタンスを初期化します。

### [Dispose\(bool\) メソッド \[191 ページ\]](#)

SDataAdapter オブジェクトによって使用されるアンマネージリソースを解放するほか、必要に応じてマネージリソースを解放します。

### [Fill メソッド \[192 ページ\]](#)

データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。

### [FillSchema メソッド \[195 ページ\]](#)

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

### [GetFillParameters\(\) メソッド \[198 ページ\]](#)

SELECT 文の実行時に自分が設定したパラメータを返します。

### [InitializeBatching\(\) メソッド \[198 ページ\]](#)

SDataAdapter オブジェクトのためのバッチを初期化します。

### [OnRowUpdated\(RowUpdatedEventArgs\) メソッド \[199 ページ\]](#)

.NET Framework データプロバイダの RowUpdated イベントを発生します。

### [OnRowUpdating\(RowUpdatingEventArgs\) メソッド \[199 ページ\]](#)

.NET Framework データプロバイダの RowUpdating イベントを発生します。

### [TerminateBatching\(\) メソッド \[200 ページ\]](#)

SDataAdapter オブジェクトのためのバッチを終了します。

### [Update\(DataRow\[\], DataTableMapping\) メソッド \[200 ページ\]](#)

DataSet に加えられた変更内容で、データベース内のテーブルを更新します。

### [DeleteCommand プロパティ \[201 ページ\]](#)

DataSet で削除されたローに該当するデータベース内のローを削除するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。



#### [InsertCommand プロパティ \[202 ページ\]](#)

DataSet に挿入されたローに該当するローをデータベースに追加するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

#### [SelectCommand プロパティ \[202 ページ\]](#)

Fill または FillSchema の実行時に、DataSet にコピーするための結果セットをデータベースから取得するために使用される SACommand を指定します。

#### [TableMappings プロパティ \[203 ページ\]](#)

ソーステーブルと DataTable 間のマスタマッピングを提供するコレクションを指定します。

#### [UpdateBatchSize プロパティ \[204 ページ\]](#)

データベースサーバとの各往復の中で処理されるローの数を取得または設定します。

#### [UpdateCommand プロパティ \[204 ページ\]](#)

DataSet で更新されたローに該当するデータベース内のローを更新するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

#### [RowUpdated イベント \[205 ページ\]](#)

データソースに対してコマンドが実行された後の更新時に発生します。

#### [RowUpdating イベント \[205 ページ\]](#)

データソースに対してコマンドが実行される前の更新時に発生します。

## 1.11.1 SDataAdapter コンストラクタ

SDataAdapter オブジェクトを初期化します。

### オーバロードリスト

変数とタイプ	オーバーロード名	説明
public	<a href="#">SDataAdapter() [186 ページ]</a>	SDataAdapter オブジェクトを初期化します。
public	<a href="#">SDataAdapter(SACommand) [186 ページ]</a>	SDataAdapter オブジェクトを、指定した SELECT 文で初期化します。
public	<a href="#">SDataAdapter(string, SAConnection) [187 ページ]</a>	SDataAdapter オブジェクトを、指定した SELECT 文および接続で初期化します。
public	<a href="#">SDataAdapter(string, string) [188 ページ]</a>	SDataAdapter オブジェクトを、指定した SELECT 文および接続文字列で初期化します。

このセクションの内容:

#### [SDataAdapter\(\) コンストラクタ \[186 ページ\]](#)

SDataAdapter オブジェクトを初期化します。

[SDataAdapter\(SACommand\) コンストラクタ \[186 ページ\]](#)

SDataAdapter オブジェクトを、指定した SELECT 文で初期化します。

[SDataAdapter\(string, SAConnection\) コンストラクタ \[187 ページ\]](#)

SDataAdapter オブジェクトを、指定した SELECT 文および接続で初期化します。

[SDataAdapter\(string, string\) コンストラクタ \[188 ページ\]](#)

SDataAdapter オブジェクトを、指定した SELECT 文および接続文字列で初期化します。

### 1.11.1.1 SDataAdapter() コンストラクタ

SDataAdapter オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SDataAdapter ()
```

##### C#

```
public SDataAdapter ()
```

#### 関連情報

[SDataAdapter\(SACommand\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(string, SAConnection\) コンストラクタ \[187 ページ\]](#)

[SDataAdapter\(string, string\) コンストラクタ \[188 ページ\]](#)

### 1.11.1.2 SDataAdapter(SACommand) コンストラクタ

SDataAdapter オブジェクトを、指定した SELECT 文で初期化します。

#### 構文

##### Visual Basic

```
Public Sub SDataAdapter (ByVal selectCommand As SACommand)
```

##### C#

```
public SDataAdapter (SACommand selectCommand)
```

## パラメータ

**selectCommand** System.Data.DataSet に配置するためのレコードをデータソースから選択するために System.Data.Common.DbDataAdapter.Fill(System.Data.DataSet) の実行時に使用される SACommand オブジェクト。

## 関連情報

[SDataAdapter\(\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(string, SAConnection\) コンストラクタ \[187 ページ\]](#)

[SDataAdapter\(string, string\) コンストラクタ \[188 ページ\]](#)

### 1.11.1.3 SDataAdapter(string, SAConnection) コンストラクタ

SDataAdapter オブジェクトを、指定した SELECT 文および接続で初期化します。

#### 構文

##### Visual Basic

```
Public Sub SDataAdapter (
    ByVal selectCommandText As String,
    ByVal selectConnection As SAConnection
)
```

##### C#

```
public SDataAdapter (
    string selectCommandText,
    SAConnection selectConnection
)
```

## パラメータ

**selectCommandText** SDataAdapter オブジェクトの SDataAdapter.SelectCommand プロパティを設定するために使用される SELECT 文。

**selectConnection** データベースへの接続を定義する SAConnection オブジェクト。

## 関連情報

[SDataAdapter\(\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(SACommand\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(string, string\) コンストラクタ \[188 ページ\]](#)

[SelectCommand プロパティ \[202 ページ\]](#)

[SAConnection クラス \[118 ページ\]](#)

### 1.11.1.4 SDataAdapter(string, string) コンストラクタ

SDataAdapter オブジェクトを、指定した SELECT 文および接続文字列で初期化します。

#### 構文

##### Visual Basic

```
Public Sub SDataAdapter (  
    ByVal selectCommandText As String,  
    ByVal selectConnectionString As String  
)
```

##### C#

```
public SDataAdapter (  
    string selectCommandText,  
    string selectConnectionString  
)
```

#### パラメータ

**selectCommandText** SDataAdapter オブジェクトの SDataAdapter.SelectCommand プロパティを設定するために使用される SELECT 文。

**selectConnectionString** データベースの接続文字列。

#### 関連情報

[SDataAdapter\(\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(SACommand\) コンストラクタ \[186 ページ\]](#)

[SDataAdapter\(string, SAConnection\) コンストラクタ \[187 ページ\]](#)

[SelectCommand プロパティ \[202 ページ\]](#)

## 1.11.2 ClearBatch() メソッド

バッチからすべての SACommand オブジェクトを削除します。

### 構文

#### Visual Basic

```
Protected Overrides Sub ClearBatch ()
```

#### C#

```
protected override void ClearBatch ()
```

### 関連情報

[SACommand クラス \[53 ページ\]](#)

## 1.11.3 CreateRowUpdatedEvent(DataRow, IDbCommand, StatementType, DataTableMapping) メソッド

System.Data.Common.RowUpdatedEventArgs クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Protected Overrides Function CreateRowUpdatedEvent (  
    ByVal dataRow As DataRow,  
    ByVal command As IDbCommand,  
    ByVal statementType As StatementType,  
    ByVal tableMapping As DataTableMapping  
    ) As RowUpdatedEventArgs
```

#### C#

```
protected override RowUpdatedEventArgs CreateRowUpdatedEvent (  
    DataRow dataRow,  
    IDbCommand command,  
    StatementType statementType,  
    DataTableMapping tableMapping  
    )
```

## パラメータ

**dataRow** データソースの更新に使用する System.Data.DataRow。

**command** System.Data.IDataAdapter.Update(System.Data.DataSet) 中に実行される System.Data.IDbCommand。

**statementType** コマンドが UPDATE、INSERT、DELETE、または SELECT 文かどうか。

**tableMapping** System.Data.Common.DataTableMapping オブジェクト。

## 戻り値

System.Data.Common.RowUpdatedEventArgs クラスの新しいインスタンス。

### 1.11.4 CreateRowUpdatingEvent(DataRow, IDbCommand, StatementType, DataTableMapping) メソッド

System.Data.Common.RowUpdatingEventArgs クラスの新しいインスタンスを初期化します。

#### 構文

##### Visual Basic

```
Protected Overrides Function CreateRowUpdatingEvent (  
    ByVal dataRow As DataRow,  
    ByVal command As IDbCommand,  
    ByVal statementType As StatementType,  
    ByVal tableMapping As DataTableMapping  
) As RowUpdatingEventArgs
```

##### C#

```
protected override RowUpdatingEventArgs CreateRowUpdatingEvent (  
    DataRow dataRow,  
    IDbCommand command,  
    StatementType statementType,  
    DataTableMapping tableMapping  
)
```

## パラメータ

**dataRow** データソースの更新に使用する System.Data.DataRow。

**command** System.Data.IDataAdapter.Update(System.Data.DataSet) 中に実行される System.Data.IDbCommand。

**statementType** コマンドが UPDATE、INSERT、DELETE、または SELECT 文かどうか。

**tableMapping** System.Data.Common.DataTableMapping オブジェクト。

戻り値

System.Data.Common.RowUpdatingEventArgs クラスの新しいインスタンス。

### 1.11.5 Dispose(bool) メソッド

SDataAdapter オブジェクトによって使用されるアンマネージリソースを解放するほか、必要に応じてマネージリソースを解放します。

#### 構文

##### Visual Basic

```
Protected Overrides Sub Dispose (ByVal disposing As Boolean)
```

##### C#

```
protected override void Dispose (bool disposing)
```

パラメータ

**disposing** マネージリソースとアンマネージリソースの両方を解放する場合は true、アンマネージリソースのみを解放する場合は false。

関連情報

[SDataAdapter クラス \[181 ページ\]](#)

## 1.11.6 Fill メソッド

データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
protected override int	<a href="#">Fill(DataSet, int, int, string, IDbCommand, CommandBehavior)</a> [192 ページ]	データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。
protected override int	<a href="#">Fill(DataTable[], int, int, IDbCommand, CommandBehavior)</a> [194 ページ]	System.Data.DataSet および System.Data.DataTable 名を使用して、System.Data.DataSet で指定された範囲にあるローを追加または再表示し、データソース内のローと一致させます。

このセクションの内容:

[Fill\(DataSet, int, int, string, IDbCommand, CommandBehavior\) メソッド](#) [192 ページ]

データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。

[Fill\(DataTable\[\], int, int, IDbCommand, CommandBehavior\) メソッド](#) [194 ページ]

System.Data.DataSet および System.Data.DataTable 名を使用して、System.Data.DataSet で指定された範囲にあるローを追加または再表示し、データソース内のローと一致させます。

### 1.11.6.1 Fill(DataSet, int, int, string, IDbCommand, CommandBehavior) メソッド

データベースからのデータで、System.Data.DataSet または System.Data.DataTable オブジェクトのローを追加または再表示します。

#### 構文

##### Visual Basic

```
Protected Overrides Function Fill (  
    ByVal dataSet As DataSet,  
    ByVal startRecord As Integer,  
    ByVal maxRecords As Integer,  
    ByVal srcTable As String,  
    ByVal command As IDbCommand,  
    ByVal behavior As CommandBehavior  
) As Integer
```



## C#

```
protected override int Fill (  
    DataSet dataSet,  
    int startRecord,  
    int maxRecords,  
    string srcTable,  
    IDbCommand command,  
    CommandBehavior behavior  
)
```

## パラメータ

**dataSet** レコードと (必要に応じて) スキーマを使用して設定する System.Data.DataSet。

**startRecord** 先頭の 0 から始まるレコード番号。

**maxRecords** System.Data.DataSet に読み込むレコードの最大数。

**srcTable** テーブルマッピングに使用するソーステーブルの名前。

**command** データソースからローを取得するために使用される SQL SELECT 文。

**behavior** CommandBehavior 値の 1 つ。

## 戻り値

System.Data.DataSet で正常に追加または再表示されたローの数。

## 備考

startRecord 引数を使用して、DataSet にコピーされるレコードの数を制限した場合でも、SDataAdapter クエリ内のすべてのレコードがデータベースからクライアントにフェッチされます。結果セットが大きい場合は、パフォーマンスに重大な影響を与える可能性があります。

別の方法として、読み込み専用、前方専用の結果セットで十分であれば、SDataReader (場合によっては SQL 文 (ExecuteNonQuery) とともに) を使用して修正を行います。その他にも、必要な結果のみを返すストアードプロシージャを作成するという方法もあります。

SelectCommand がローを返さない場合、DataSet にテーブルが追加されず、例外も発生しません。

詳細については、「データのアクセスとデータの操作」を参照してください。

## 1.11.6.2 Fill(DataTable[], int, int, IDbCommand, CommandBehavior) メソッド

System.Data.DataSet および System.Data.DataTable 名を使用して、System.Data.DataSet で指定された範囲にあるローを追加または再表示し、データソース内のローと一致させます。

### 構文

#### Visual Basic

```
Protected Overrides Function Fill (  
    ByVal dataTables As DataTable(),  
    ByVal startRecord As Integer,  
    ByVal maxRecords As Integer,  
    ByVal command As IDbCommand,  
    ByVal behavior As CommandBehavior  
    ) As Integer
```

#### C#

```
protected override int Fill (  
    DataTable[] dataTables,  
    int startRecord,  
    int maxRecords,  
    IDbCommand command,  
    CommandBehavior behavior  
    )
```

### パラメータ

**dataTables** データソースから表示する System.Data.DataTable オブジェクト。

**startRecord** 0 から始まる先頭のレコード番号。

**maxRecords** 取得されるレコードの最大数。

**command** System.Data.DataTable オブジェクトを表示するために実行される System.Data.IDbCommand。

**behavior** System.Data.CommandBehavior 値の 1 つ。

### 戻り値

データテーブルに追加または再表示されるローの数。

### 備考

startRecord 引数を使用して、DataSet にコピーされるレコードの数を制限した場合でも、SDataAdapter クエリ内のすべてのレコードがデータベースからクライアントにフェッチされます。結果セットが大きい場合は、パフォーマンスに重大な影響を与える可能性があります。

別の方法として、読み込み専用、前方専用の結果セットで十分であれば、SADataReader (場合によっては SQL 文 (ExecuteNonQuery) とともに) を使用して修正を行います。その他にも、必要な結果のみを返すストアプロシージャを作成するという方法もあります。

SelectCommand がローを返さない場合、DataSet にテーブルが追加されず、例外も発生しません。

詳細については、「データのアクセスとデータの操作」を参照してください。

## 1.11.7 FillSchema メソッド

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
protected override DataTable[]	<a href="#">FillSchema(DataSet, SchemaType, IDbCommand, string, CommandBehavior) [196 ページ]</a>	System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。
protected override DataTable	<a href="#">FillSchema(DataTable, SchemaType, IDbCommand, CommandBehavior) [197 ページ]</a>	System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

このセクションの内容:

#### [FillSchema\(DataSet, SchemaType, IDbCommand, string, CommandBehavior\) メソッド \[196 ページ\]](#)

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

#### [FillSchema\(DataTable, SchemaType, IDbCommand, CommandBehavior\) メソッド \[197 ページ\]](#)

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

## 1.11.7.1 FillSchema(DataSet, SchemaType, IDbCommand, string, CommandBehavior) メソッド

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

### 構文

#### Visual Basic

```
Protected Overrides Function FillSchema (  
    ByVal dataSet As DataSet,  
    ByVal schemaType As SchemaType,  
    ByVal command As IDbCommand,  
    ByVal srcTable As String,  
    ByVal behavior As CommandBehavior  
) As DataTable()
```

#### C#

```
protected override DataTable[] FillSchema (  
    DataSet dataSet,  
    SchemaType schemaType,  
    IDbCommand command,  
    string srcTable,  
    CommandBehavior behavior  
)
```

### パラメータ

**dataSet** スキーマを使用して設定する System.Data.DataSet。

**schemaType** スキーマを挿入する方法を指定する System.Data.SchemaType 値の 1 つ。

**command** データソースからローを取得するために使用される SQL SELECT 文。

**srcTable** テーブルマッピングに使用するソーステーブルの名前。

**behavior** System.Data.CommandBehavior 値の 1 つ。

### 戻り値

System.Data.DataSet に追加された System.Data.DataTable オブジェクトのコレクションへの参照。

### 備考

詳細については、System.Data.IDataAdapter.FillSchema および「データのアクセスとデータの操作」を参照してください。

## 1.11.7.2 FillSchema(DataTable, SchemaType, IDbCommand, CommandBehavior) メソッド

System.Data.DataTable を System.Data.DataSet に追加し、スキーマがデータソースのスキーマと一致するように設定します。

### 構文

#### Visual Basic

```
Protected Overrides Function FillSchema (  
    ByVal dataTable As DataTable,  
    ByVal schemaType As SchemaType,  
    ByVal command As IDbCommand,  
    ByVal behavior As CommandBehavior  
) As DataTable
```

#### C#

```
protected override DataTable FillSchema (  
    DataTable dataTable,  
    SchemaType schemaType,  
    IDbCommand command,  
    CommandBehavior behavior  
)
```

### パラメータ

**dataTable** スキーマを使用して設定する System.Data.DataTable。

**schemaType** スキーマを挿入する方法を指定する System.Data.SchemaType 値の 1 つ。

**command** データソースからローを取得するために使用される SQL SELECT 文。

**behavior** System.Data.CommandBehavior 値の 1 つ。

### 戻り値

スキーマが含まれている System.Data.DataTable オブジェクトへの参照。

### 備考

詳細については、

System.Data.Common.DbDataAdapter.FillSchema(System.Data.DataTable, System.Data.SchemaType) および「データのアクセスとデータの操作」を参照してください。

## 1.11.8 GetFillParameters() メソッド

SELECT 文の実行時に自分が設定したパラメータを返します。

### 構文

#### Visual Basic

```
Public Shadows Function GetFillParameters () As SAPParameter()
```

#### C#

```
public new SAPParameter[] GetFillParameters ()
```

### 戻り値

ユーザによって設定されたパラメータが含まれる IDataParameter オブジェクトの配列。

## 1.11.9 InitializeBatching() メソッド

SDataAdapter オブジェクトのためのバッチを初期化します。

### 構文

#### Visual Basic

```
Protected Overrides Sub InitializeBatching ()
```

#### C#

```
protected override void InitializeBatching ()
```

### 関連情報

[SDataAdapter クラス \[181 ページ\]](#)

## 1.11.10 OnRowUpdated(RowUpdatedEventArgs) メソッド

.NET Framework データプロバイダの RowUpdated イベントを発生します。

### 構文

#### Visual Basic

```
Protected Overrides Sub OnRowUpdated (ByVal value As RowUpdatedEventArgs)
```

#### C#

```
protected override void OnRowUpdated (RowUpdatedEventArgs value)
```

### パラメータ

**value** イベントデータが含まれる System.Data.Common.RowUpdatedEventArgs。

## 1.11.11 OnRowUpdating(RowUpdatingEventArgs) メソッド

.NET Framework データプロバイダの RowUpdating イベントを発生します。

### 構文

#### Visual Basic

```
Protected Overrides Sub OnRowUpdating (ByVal value As RowUpdatingEventArgs)
```

#### C#

```
protected override void OnRowUpdating (RowUpdatingEventArgs value)
```

### パラメータ

**value** イベントデータが含まれる System.Data.Common.RowUpdatingEventArgs。

## 1.11.12 TerminateBatching() メソッド

SDataAdapter オブジェクトのためのバッチを終了します。

### 構文

#### Visual Basic

```
Protected Overrides Sub TerminateBatching ()
```

#### C#

```
protected override void TerminateBatching ()
```

### 関連情報

[SDataAdapter クラス \[181 ページ\]](#)

## 1.11.13 Update(DataRow[], DataTableMapping) メソッド

DataSet に加えられた変更内容で、データベース内のテーブルを更新します。

### 構文

#### Visual Basic

```
Protected Overrides Function Update (
    ByVal dataRows As DataRow(),
    ByVal tableMapping As DataTableMapping
) As Integer
```

#### C#

```
protected override int Update (
    DataRow[] dataRows,
    DataTableMapping tableMapping
)
```

### パラメータ

**dataRows** 更新の元になる System.Data.DataRow の配列。

**tableMapping** 使用する System.Data.IDataAdapter.TableMappings コレクション。



## 戻り値

System.Data.DataRow 配列から正常に更新されたローの数。

## 備考

データセット内の挿入、更新、または削除された各ローの InsertCommand、UpdateCommand、および DeleteCommand を使用して、Update が実行されます。

詳細については、「データのアクセスとデータの操作」を参照してください。

## 関連情報

[DeleteCommand プロパティ \[201 ページ\]](#)

[InsertCommand プロパティ \[202 ページ\]](#)

[UpdateCommand プロパティ \[204 ページ\]](#)

## 1.11.14 DeleteCommand プロパティ

DataSet で削除されたローに該当するデータベース内のローを削除するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

### 構文

#### Visual Basic

```
Public Shadows Property DeleteCommand As SACommand
```

#### C#

```
public new SACommand DeleteCommand {get;set;}
```

## 備考

Update の実行時にこのプロパティが設定されておらず、DataSet にプライマリーキー情報がある場合、SelectCommand を設定して SACommandBuilder を使用すると、DeleteCommand を自動的に生成できます。この場合、SACommandBuilder は、設定されていない追加コマンドを生成します。この生成論理には、SelectCommand に表示されるキーカラム情報が必要です。

DeleteCommand が既存の SACommand オブジェクトに割り当てられる場合、SACommand オブジェクトのクローンは作成されません。DeleteCommand は、既存の SACommand への参照を保持します。

## 関連情報

[SelectCommand プロパティ \[202 ページ\]](#)

### 1.11.15 InsertCommand プロパティ

DataSet に挿入されたローに該当するローをデータベースに追加するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

#### 構文

##### Visual Basic

```
Public Shadows Property InsertCommand As SACommand
```

##### C#

```
public new SACommand InsertCommand {get;set;}
```

## 備考

SACommandBuilder には、InsertCommand を生成するためのキーカラムは必要ありません。

InsertCommand が既存の SACommand オブジェクトに割り当てられる場合、SACommand オブジェクトのクローンは作成されません。InsertCommand は、既存の SACommand への参照を保持します。

このコマンドがローを返す場合、SACommand オブジェクトの UpdatedRowSource プロパティの設定方法によっては、これらのローが DataSet に追加されることがあります。

### 1.11.16 SelectCommand プロパティ

Fill または FillSchema の実行時に、DataSet にコピーするための結果セットをデータベースから取得するために使用される SACommand を指定します。

#### 構文

##### Visual Basic

```
Public Shadows Property SelectCommand As SACommand
```

## C#

```
public new SACommand SelectCommand {get;set;}
```

## 備考

SelectCommand が以前に作成された SACommand オブジェクトに割り当てられる場合、SACommand オブジェクトのクローンは作成されません。SelectCommand は、以前に作成された SACommand オブジェクトへの参照を保持します。

SelectCommand がローを返さない場合、DataSet にテーブルが追加されず、例外も発生しません。

SELECT 文は、SDataAdapter コンストラクタにも指定できます。

## 1.11.17 TableMappings プロパティ

ソーステーブルと DataTable 間のマスタマッピングを提供するコレクションを指定します。

### 構文

#### Visual Basic

```
Public ReadOnly Shadows Property TableMappings As DataTableMappingCollection
```

#### C#

```
public new DataTableMappingCollection TableMappings {get;}
```

## 備考

デフォルト値は空のコレクションです。

変更を調整する場合、SDataAdapter は DataTableMappingCollection コレクションを使用して、データソースによって使用されるカラム名を、DataSet によって使用されるカラム名に関連付けます。

## 1.11.18 UpdateBatchSize プロパティ

データベースサーバとの各往復の中で処理されるローの数を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property UpdateBatchSize As Integer
```

#### C#

```
public override int UpdateBatchSize {get;set;}
```

### 備考

デフォルト値は 1 です。

1 より大きい値を設定すると、SADDataAdapter.Update は、すべての挿入文をバッチで実行します。削除と更新はこれまでのように順次実行されますが、挿入は、UpdateBatchSize の値のサイズのバッチで、後で実行されます。0 を設定すると、Update は挿入文を 1 つのバッチで送信します。

1 より大きい値を設定すると、SADDataAdapter.Fill は、すべての挿入文をバッチで実行します。削除と更新はこれまでのように順次実行されますが、挿入は、UpdateBatchSize の値のサイズのバッチで、後で実行されます。

0 を設定すると、Fill は挿入文を 1 つのバッチで送信します。

0 未満の値を設定すると、エラーになります。

UpdateBatchSize が 1 以外の値に設定され、InsertCommand プロパティが INSERT 文以外のものに設定されている場合、Fill を呼び出すときに例外がスローされます。

この動作は、SqlDataAdapter とは異なります。SqlDataAdapter は、すべての種類のコマンドをバッチ処理します。

## 1.11.19 UpdateCommand プロパティ

DataSet で更新されたローに該当するデータベース内のローを更新するために、Update メソッドが呼び出されたときにデータベースに対して実行される SACommand オブジェクトを指定します。

### 構文

#### Visual Basic

```
Public Shadows Property UpdateCommand As SACommand
```

#### C#

```
public new SACommand UpdateCommand {get;set;}
```

## 備考

Update の実行時に、このプロパティが設定されておらず、SelectCommand にプライマリキー情報がある場合、SelectCommand プロパティを設定して SACommandBuilder を使用すると、UpdateCommand を自動的に生成できます。次に、設定していない追加コマンドが SACommandBuilder によって生成されます。この生成論理には、SelectCommand に表示されるキーカラム情報が必要です。

UpdateCommand が以前に作成された SACommand オブジェクトに割り当てられる場合、SACommand オブジェクトのクローンは作成されません。UpdateCommand は、以前に作成された SACommand オブジェクトへの参照を保持します。

このコマンドを実行するとローが返される場合、SACommand オブジェクトの UpdatedRowSource プロパティの設定方法によっては、これらのローが DataSet とマージされることがあります。

## 1.11.20 RowUpdated イベント

データソースに対してコマンドが実行された後の更新時に発生します。

### 構文

#### Visual Basic

```
Public Event RowUpdated As SARowUpdatedEventHandler
```

#### C#

```
public SARowUpdatedEventHandler RowUpdated;
```

## 備考

更新が試みられると、イベントが発生します。

イベントハンドラは、このイベントに関するデータが含まれるタイプ SARowUpdatedEventArgs の引数を受け取ります。

詳細については、.NET Framework のマニュアルの OleDbDataAdapter.RowUpdated イベントを参照してください。

## 1.11.21 RowUpdating イベント

データソースに対してコマンドが実行される前の更新時に発生します。

### 構文

#### Visual Basic

```
Public Event RowUpdating As SARowUpdatingEventHandler
```

## C#

```
public SARowUpdatingEventHandler RowUpdating;
```

## 備考

更新が試みられると、イベントが発生します。

イベントハンドラは、このイベントに関するデータが含まれるタイプ `SARowUpdatingEventArgs` の引数を受け取ります。

詳細については、.NET Framework のマニュアルの `OleDbDataAdapter.RowUpdating` イベントを参照してください。

## 1.12 SDataReader クラス

クエリまたはストアプロシージャからの読み込み専用、前方専用の結果セットです。

### 構文

#### Visual Basic

```
Public NotInheritable Class SDataReader Inherits  
System.Data.Common.DbDataReader Implements System.ComponentModel.IListSource
```

#### C#

```
public sealed class SDataReader : System.Data.Common.DbDataReader,  
System.ComponentModel.IListSource
```

## メンバー

`SDataReader` のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変更子とタイプ	メソッド	説明
public override void	<a href="#">Close()</a> [211 ページ]	<code>SDataReader</code> を閉じます。
public override bool	<a href="#">GetBoolean(int)</a> [212 ページ]	指定されたカラムの値をブール値として返します。
public override byte	<a href="#">GetByte(int)</a> [213 ページ]	指定されたカラムの値をバイトとして返します。

変数とタイプ	メソッド	説明
public override unsafe long	<a href="#">GetBytes(int, long, byte[], int, int) [213 ページ]</a>	指定されたカラムオフセットからバイトのストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。
public override char	<a href="#">GetChar(int) [214 ページ]</a>	指定されたカラムの値を文字として返します。
public override unsafe long	<a href="#">GetChars(int, long, char[], int, int) [215 ページ]</a>	指定されたカラムオフセットから文字のストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。
public new IDataReader	<a href="#">GetData(int) [217 ページ]</a>	このメソッドはサポートされていません。
public override string	<a href="#">GetDataTypeName(int) [217 ページ]</a>	ソースデータ型の名前を返します。
public override DateTime	<a href="#">GetDateTime(int) [218 ページ]</a>	指定されたカラムの値を DateTime オブジェクトとして返します。
public DateTimeOffset	<a href="#">GetDateTimeOffset(int) [219 ページ]</a>	指定されたカラムの値を DateTimeOffset オブジェクトとして返します。
public override decimal	<a href="#">GetDecimal(int) [220 ページ]</a>	指定されたカラムの値を Decimal オブジェクトとして返します。
public override double	<a href="#">GetDouble(int) [221 ページ]</a>	指定されたカラムの値を倍精度の浮動小数点数として返します。
public override IEnumerator	<a href="#">GetEnumerator() [222 ページ]</a>	SADaReader オブジェクトで反復処理する System.Collections.IEnumerator を返します。
public override Type	<a href="#">GetFieldType(int) [222 ページ]</a>	オブジェクトのデータ型である Type を返します。
public override float	<a href="#">GetFloat(int) [223 ページ]</a>	指定されたカラムの値を単精度の浮動小数点数として返します。
public override Guid	<a href="#">GetGuid(int) [224 ページ]</a>	指定されたカラムの値をグローバル意識別子 (GUID) として返します。
public override short	<a href="#">GetInt16(int) [225 ページ]</a>	指定されたカラムの値を 16 ビット符号付き整数として返します。
public override int	<a href="#">GetInt32(int) [225 ページ]</a>	指定されたカラムの値を 32 ビット符号付き整数として返します。
public override long	<a href="#">GetInt64(int) [226 ページ]</a>	指定されたカラムの値を 64 ビット符号付き整数として返します。
public override string	<a href="#">GetName(int) [227 ページ]</a>	指定されたカラムの名前を返します。
public override int	<a href="#">GetOrdinal(string) [227 ページ]</a>	指定されたカラム名に対応するカラムの順序を返します。
public override unsafe DataTable	<a href="#">GetSchemaTable() [228 ページ]</a>	SADaReader のカラムメタデータが記述された DataTable を返します。
public override string	<a href="#">GetString(int) [230 ページ]</a>	指定されたカラムの値を文字列として返します。

変数とタイプ	メソッド	説明
public TimeSpan	<a href="#">GetTimeSpan(int) [231 ページ]</a>	指定されたカラムの値を TimeSpan オブジェクトとして返します。
public ushort	<a href="#">GetUInt16(int) [232 ページ]</a>	指定されたカラムの値を 16 ビット符号なし整数として返します。
public uint	<a href="#">GetUInt32(int) [233 ページ]</a>	指定されたカラムの値を 32 ビット符号なし整数として返します。
public ulong	<a href="#">GetUInt64(int) [233 ページ]</a>	指定されたカラムの値を 64 ビット符号なし整数として返します。
public override object	<a href="#">GetValue [234 ページ]</a>	指定されたカラムの値を Object として返します。
public override unsafe int	<a href="#">GetValues(object[]) [236 ページ]</a>	現在のローのすべてのカラムを取得します。
public override bool	<a href="#">IsDBNull(int) [237 ページ]</a>	カラムに NULL 値が含まれるかどうかを示す値を返します。
public void	<a href="#">myDispose() [238 ページ]</a>	オブジェクトに関連付けられているリソースを解放します。
public override bool	<a href="#">NextResult() [238 ページ]</a>	複数の結果セットを返すクエリを処理するとき、SADaReader を次の結果セットに進めます。
public override unsafe bool	<a href="#">Read() [239 ページ]</a>	結果セットの次のローを読み込み、SADaReader をこのローに移動します。

## プロパティ

変数とタイプ	プロパティ	説明
public override int	<a href="#">Depth [240 ページ]</a>	現在のローのネストの深さを示す値を取得します。
public override int	<a href="#">FieldCount [240 ページ]</a>	結果セット内のカラム数を取得します。
public override bool	<a href="#">HasRows [241 ページ]</a>	SADaReader に 1 つ以上のローが含まれるかどうかを示す値を取得します。
public override bool	<a href="#">IsClosed [241 ページ]</a>	SADaReader が閉じているかどうかを示す値を取得します。
public override int	<a href="#">RecordsAffected [242 ページ]</a>	SQL 文の実行によって変更、挿入、または削除されたローの数。
public override object	<a href="#">this [242 ページ]</a>	カラムの値をネイティブフォーマットで返します。

## 備考

SADaReader にはコンストラクタがありません。SADaReader オブジェクトを取得するには、SACommand を実行します。

```
SACommand cmd = new SACommand(
```



```
"SELECT EmployeeID FROM Employees", conn );
SADataReader reader = cmd.ExecuteReader();
```

SADataReader では前方へのみ移動できます。結果を操作するためにより柔軟なオブジェクトが必要な場合は、SADataAdapter を使用します。

SADataReader は必要に応じてローを取得しますが、SADataAdapter の場合、オブジェクトに対してアクションを実行する前に、結果セットのすべてのローを取得する必要があります。結果セットのサイズが大きい場合、この違いのために SADataReader の方が応答時間が速くなります。

**実装:** IDataReader、IDisposable、IDataRecord、IListSource

詳細については、「データのアクセスとデータの操作」を参照してください。

このセクションの内容:

[Close\(\) メソッド \[211 ページ\]](#)

SADataReader を閉じます。

[GetBoolean\(int\) メソッド \[212 ページ\]](#)

指定されたカラムの値をブール値として返します。

[GetByte\(int\) メソッド \[213 ページ\]](#)

指定されたカラムの値をバイトとして返します。

[GetBytes\(int, long, byte\[\], int, int\) メソッド \[213 ページ\]](#)

指定されたカラムオフセットからバイトのストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。

[GetChar\(int\) メソッド \[214 ページ\]](#)

指定されたカラムの値を文字として返します。

[GetChars\(int, long, char\[\], int, int\) メソッド \[215 ページ\]](#)

指定されたカラムオフセットから文字のストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。

[GetData\(int\) メソッド \[217 ページ\]](#)

このメソッドはサポートされていません。

[GetDataTypeName\(int\) メソッド \[217 ページ\]](#)

ソースデータ型の名前を返します。

[GetDateTime\(int\) メソッド \[218 ページ\]](#)

指定されたカラムの値を DateTime オブジェクトとして返します。

[GetDateTimeOffset\(int\) メソッド \[219 ページ\]](#)

指定されたカラムの値を DateTimeOffset オブジェクトとして返します。

[GetDecimal\(int\) メソッド \[220 ページ\]](#)

指定されたカラムの値を Decimal オブジェクトとして返します。

[GetDouble\(int\) メソッド \[221 ページ\]](#)

指定されたカラムの値を倍精度の浮動小数点数として返します。

[GetEnumerator\(\) メソッド \[222 ページ\]](#)

SADataReader オブジェクトで反復処理する System.Collections.IEnumerator を返します。

[GetFieldType\(int\) メソッド \[222 ページ\]](#)

オブジェクトのデータ型である Type を返します。

#### [GetFloat\(int\) メソッド \[223 ページ\]](#)

指定されたカラムの値を単精度の浮動小数点数として返します。

#### [GetGuid\(int\) メソッド \[224 ページ\]](#)

指定されたカラムの値をグローバル一意識別子 (GUID) として返します。

#### [GetInt16\(int\) メソッド \[225 ページ\]](#)

指定されたカラムの値を 16 ビット符号付き整数として返します。

#### [GetInt32\(int\) メソッド \[225 ページ\]](#)

指定されたカラムの値を 32 ビット符号付き整数として返します。

#### [GetInt64\(int\) メソッド \[226 ページ\]](#)

指定されたカラムの値を 64 ビット符号付き整数として返します。

#### [GetName\(int\) メソッド \[227 ページ\]](#)

指定されたカラムの名前を返します。

#### [GetOrdinal\(string\) メソッド \[227 ページ\]](#)

指定されたカラム名に対応するカラムの順序を返します。

#### [GetSchemaTable\(\) メソッド \[228 ページ\]](#)

SADaReader のカラムメタデータが記述された DataTable を返します。

#### [GetString\(int\) メソッド \[230 ページ\]](#)

指定されたカラムの値を文字列として返します。

#### [GetTimeSpan\(int\) メソッド \[231 ページ\]](#)

指定されたカラムの値を TimeSpan オブジェクトとして返します。

#### [GetUInt16\(int\) メソッド \[232 ページ\]](#)

指定されたカラムの値を 16 ビット符号なし整数として返します。

#### [GetUInt32\(int\) メソッド \[233 ページ\]](#)

指定されたカラムの値を 32 ビット符号なし整数として返します。

#### [GetUInt64\(int\) メソッド \[233 ページ\]](#)

指定されたカラムの値を 64 ビット符号なし整数として返します。

#### [GetValue メソッド \[234 ページ\]](#)

指定されたカラムの値を Object として返します。

#### [GetValues\(object\[\]\) メソッド \[236 ページ\]](#)

現在のローのすべてのカラムを取得します。

#### [IsDBNull\(int\) メソッド \[237 ページ\]](#)

カラムに NULL 値が含まれるかどうかを示す値を返します。

#### [myDispose\(\) メソッド \[238 ページ\]](#)

オブジェクトに関連付けられているリソースを解放します。

#### [NextResult\(\) メソッド \[238 ページ\]](#)

複数の結果セットを返すクエリを処理するとき、SADaReader を次の結果セットに進めます。

#### [Read\(\) メソッド \[239 ページ\]](#)

結果セットの次のローを読み込み、SADaReader をこのローに移動します。

#### [Depth プロパティ \[240 ページ\]](#)

現在のローのネストの深さを示す値を取得します。

#### [FieldCount プロパティ \[240 ページ\]](#)

結果セット内のカラム数を取得します。

#### [HasRows プロパティ \[241 ページ\]](#)

SADaReader に 1 つ以上のローが含まれるかどうかを示す値を取得します。

#### [IsClosed プロパティ \[241 ページ\]](#)

SADaReader が閉じているかどうかを示す値を取得します。

#### [RecordsAffected プロパティ \[242 ページ\]](#)

SQL 文の実行によって変更、挿入、または削除されたローの数。

#### [this プロパティ \[242 ページ\]](#)

カラムの値をネイティブフォーマットで返します。

## 関連情報

[ExecuteReader\(\) メソッド \[77 ページ\]](#)

### 1.12.1 Close() メソッド

SADaReader を閉じます。

#### 構文

##### Visual Basic

```
Public Overrides Sub Close ()
```

##### C#

```
public override void Close ()
```

## 備考

SADaReader を使用し終わったら、Close メソッドを明示的に呼び出してください。

オートコミットモードで実行している場合、SADaReader を閉じる関連動作として、COMMIT が発行されます。

## 1.12.2 GetBoolean(int) メソッド

指定されたカラムの値をブール値として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetBoolean (ByVal ordinal As Integer) As Boolean
```

#### C#

```
public override bool GetBoolean (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

カラムの値。

### 備考

変換は行われなため、取り出されるデータはすでにブール値である必要があります。

### 関連情報

[GetOrdinal\(string\) メソッド \[227 ページ\]](#)

[GetFieldType\(int\) メソッド \[222 ページ\]](#)

### 1.12.3 GetByte(int) メソッド

指定されたカラムの値をバイトとして返します。

#### 構文

##### Visual Basic

```
Public Overrides Function GetByte (ByVal ordinal As Integer) As Byte
```

##### C#

```
public override byte GetByte (int ordinal)
```

#### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

#### 戻り値

カラムの値。

#### 備考

変換は行われなため、取り出されるデータはすでにバイトである必要があります。

### 1.12.4 GetBytes(int, long, byte[], int, int) メソッド

指定されたカラムオフセットからバイトのストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。

#### 構文

##### Visual Basic

```
Public Overrides Function GetBytes (
    ByVal ordinal As Integer,
    ByVal dataIndex As Long,
    ByVal buffer As Byte(),
    ByVal bufferSize As Integer,
    ByVal length As Integer
) As Long
```

C#

```
public override unsafe long GetBytes (
    int ordinal,
    long dataIndex,
    byte[] buffer,
    int bufferSize,
    int length
)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

**dataIndex** バイトの読み込みを開始するカラム値内のインデックス。

**buffer** データの格納先となる配列。

**bufferIndex** データのコピーを開始する配列内のインデックス。

**length** 指定したバッファにコピーする最大長。

## 戻り値

読み込まれたバイト数。

## 備考

GetBytes は、フィールド内で使用可能なバイト数を返します。ほとんどの場合、これは正確なフィールド長です。ただし、GetBytes を使用してフィールドからバイトがすでに取得されている場合、返される数値が実際の長さより小さくなる可能性があります。これはたとえば、SADeveloper がサイズの大きいデータ構造体をバッファに読み込む場合などです。

NULL 参照 (Visual Basic の場合は Nothing) であるバッファを渡すと、GetBytes はフィールドの長さをバイト数で返します。

変換は行われなため、取り出されるデータはすでに byte 配列である必要があります。

## 1.12.5 GetChar(int) メソッド

指定されたカラムの値を文字として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetChar (ByVal ordinal As Integer) As Char
```

## C#

```
public override char GetChar (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

カラムの値。

## 備考

変換は行われなため、取り出されるデータはすでに文字である必要があります。

SADataReader.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

## 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.6 GetChars(int, long, char[], int, int) メソッド

指定されたカラムオフセットから文字のストリームを特定のバッファオフセットから始まる配列としてバッファに読み込みます。

### 構文

#### Visual Basic

```
Public Overrides Function GetChars (  
    ByVal ordinal As Integer,  
    ByVal dataIndex As Long,  
    ByVal buffer As Char(),  
    ByVal bufferSize As Integer,  
    ByVal length As Integer  
    ) As Long
```

## C#

```
public override unsafe long GetChars (
    int ordinal,
    long dataIndex,
    char[] buffer,
    int bufferIndex,
    int length
)
```

## パラメータ

**ordinal** カラムを表す、0 から始まる序数。

**dataIndex** 読み込みオペレーションを開始するロー内のインデックス。

**buffer** データのコピー先のバッファ。

**bufferIndex** 読み込みオペレーションを開始するバッファのインデックス。

**length** 読み込む文字数。

## 戻り値

実際に読み込まれた文字数。

## 備考

GetChars は、フィールド内で使用可能な文字数を返します。ほとんどの場合、これは正確なフィールド長です。ただし、GetChars を使用してフィールドから文字がすでに取得されている場合、返される数値が実際の長さより小さくなる可能性があります。これはたとえば、SADDataReader がサイズの大きいデータ構造体をバッファに読み込む場合などです。

NULL 参照 (Visual Basic の場合は Nothing) であるバッファを渡すと、GetChars はフィールドの長さを文字数として返しません。

変換は行われなため、取り出されるデータはすでに文字配列である必要があります。

BLOB の処理方法については、「データのアクセスとデータの操作」を参照してください。



## 1.12.7 GetData(int) メソッド

このメソッドはサポートされていません。

### 構文

#### Visual Basic

```
Public Shadows Function GetData (ByVal i As Integer) As IDataReader
```

#### C#

```
public new IDataReader GetData (int i)
```

### 備考

このメソッドを呼び出すと、`InvalidOperationException` がスローされます。

## 1.12.8 GetDataTypeName(int) メソッド

ソースデータ型の名前を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetDataTypeName (ByVal index As Integer) As String
```

#### C#

```
public override string GetDataTypeName (int index)
```

### パラメータ

**index** カラムを表す、0 から始まる序数。

### 戻り値

バックエンドデータ型の名前。

## 1.12.9 GetDateTime(int) メソッド

指定されたカラムの値を DateTime オブジェクトとして返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetDateTime (ByVal ordinal As Integer) As Date
```

#### C#

```
public override DateTime GetDateTime (int ordinal)
```

### パラメータ

**ordinal** カラムを表す、0 から始まる序数。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われなため、取り出されるデータはすでに DateTime オブジェクトである必要があります。

SADeveloper.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

### 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.10 GetDateTimeOffset(int) メソッド

指定されたカラムの値を DateTimeOffset オブジェクトとして返します。

### 構文

#### Visual Basic

```
Public Function GetDateTimeOffset (ByVal ordinal As Integer) As DateTimeOffset
```

#### C#

```
public DateTimeOffset GetDateTimeOffset (int ordinal)
```

### パラメータ

**ordinal** カラムを表す、0 から始まる序数。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われなため、取り出されるデータはすでに DateTimeOffset オブジェクトである必要があります。

SADataReader.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

### 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.11 GetDecimal(int) メソッド

指定されたカラムの値を Decimal オブジェクトとして返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetDecimal (ByVal ordinal As Integer) As Decimal
```

#### C#

```
public override decimal GetDecimal (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われなため、取り出されるデータはすでに Decimal オブジェクトである必要があります。

SADeveloper.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

### 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.12 GetDouble(int) メソッド

指定されたカラムの値を倍精度の浮動小数点数として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetDouble (ByVal ordinal As Integer) As Double
```

#### C#

```
public override double GetDouble (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われなかったため、取り出されるデータはすでに倍精度の浮動小数点数である必要があります。  
SADaReader.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

### 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.13 GetEnumerator() メソッド

SADaReader オブジェクトで反復処理する System.Collections.IEnumerator を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetEnumerator () As System.Collections.IEnumerator
```

#### C#

```
public override IEnumerator GetEnumerator ()
```

### 戻り値

SADaReader オブジェクトの System.Collections.IEnumerator。

### 関連情報

[SADaReader クラス \[206 ページ\]](#)

## 1.12.14 GetFieldType(int) メソッド

オブジェクトのデータ型である Type を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetFieldType (ByVal index As Integer) As Type
```

#### C#

```
public override Type GetFieldType (int index)
```

### パラメータ

**index** カラムを表す、0 から始まる序数。

## 戻り値

オブジェクトのデータ型である Type。

### 1.12.15 GetFloat(int) メソッド

指定されたカラムの値を単精度の浮動小数点数として返します。

#### 構文

##### Visual Basic

```
Public Overrides Function GetFloat (ByVal ordinal As Integer) As Single
```

##### C#

```
public override float GetFloat (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

変換は行われなかったため、取り出されるデータはすでに単精度の浮動小数点数である必要があります。  
SADeveloper.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

## 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.16 GetGuid(int) メソッド

指定されたカラムの値をグローバル一意識別子 (GUID) として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetGuid (ByVal ordinal As Integer) As Guid
```

#### C#

```
public override Guid GetGuid (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

### 備考

取り出されるデータは、すでにグローバル一意識別子またはバイナリ (16) である必要があります。

SADataReader.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

### 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)



## 1.12.17 GetInt16(int) メソッド

指定されたカラムの値を 16 ビット符号付き整数として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetInt16 (ByVal ordinal As Integer) As Short
```

#### C#

```
public override short GetInt16 (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われないため、取り出されるデータはすでに 16 ビット符号付き整数である必要があります。

## 1.12.18 GetInt32(int) メソッド

指定されたカラムの値を 32 ビット符号付き整数として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetInt32 (ByVal ordinal As Integer) As Integer
```

#### C#

```
public override int GetInt32 (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

変換は行われなため、取り出されるデータはすでに 32 ビット符号付き整数である必要があります。

## 1.12.19 GetInt64(int) メソッド

指定されたカラムの値を 64 ビット符号付き整数として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetInt64 (ByVal ordinal As Integer) As Long
```

#### C#

```
public override long GetInt64 (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

変換は行われないため、取り出されるデータはすでに 64 ビット符号付き整数である必要があります。

### 1.12.20 GetName(int) メソッド

指定されたカラムの名前を返します。

#### 構文

##### Visual Basic

```
Public Overrides Function GetName (ByVal index As Integer) As String
```

##### C#

```
public override string GetName (int index)
```

## パラメータ

**index** カラムの 0 から始まるインデックス。

## 戻り値

指定されたカラムの名前。

### 1.12.21 GetOrdinal(string) メソッド

指定されたカラム名に対応するカラムの順序を返します。

#### 構文

##### Visual Basic

```
Public Overrides Function GetOrdinal (ByVal name As String) As Integer
```

##### C#

```
public override int GetOrdinal (string name)
```

## パラメータ

**name** カラムの名前。

## 戻り値

0 から始まるカラムの順序。

## 備考

GetOrdinal は、最初に大文字と小文字を区別したルックアップを実行します。このルックアップが失敗した場合、2 回目は大文字と小文字を区別しないでルックアップを実行します。

GetOrdinal は、日本語のかな幅を区別しません。

順序ベースのルックアップの方が名前ベースのルックアップより効率的であるため、ループ内で GetOrdinal を呼び出すのは非効率です。GetOrdinal を 1 回呼び出し、結果を整数変数に割り当ててループ内で使用すると、時間を節約できます。

## 1.12.22 GetSchemaTable() メソッド

SADaReader のカラムメタデータが記述された DataTable を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetSchemaTable () As DataTable
```

#### C#

```
public override unsafe DataTable GetSchemaTable ()
```

## 戻り値

カラムメタデータが記述された DataTable。

## 備考

このメソッドは、各カラムに関するメタデータを次の順で返します。

DataTable カラム	説明
ColumnName	カラムの名前。カラムに名前がない場合は NULL 参照 (Visual Basic の Nothing)。SQL クエリでカラムのエイリアスが使用されている場合は、そのエイリアスが返されます。結果セットでは、すべてのカラムに名前があるとは限らず、すべてのカラム名がユニークであるとも限りません。
ColumnOrdinal	カラムの ID。値の範囲は、[0, FieldCount -1] です。
ColumnSize	サイズ指定されたカラムの場合、カラムの値の最大長。その他のカラムの場合、これは、そのデータ型のバイト単位のサイズです。
NumericPrecision	数値カラムの精度。カラムが数値型ではない場合は DBNull。
NumericScale	数値カラムの位取り。カラムが数値型ではない場合は DBNull。
IsUnique	カラムが取得元のテーブル (BaseTableName) でユニークな非計算カラムである場合は true。
IsKey	カラムが、結果セットのユニークキーからとも取得された結果セットの一連のカラムのいずれかである場合は true。IsKey が true に設定されたカラムのセットは、結果セット内のローをユニークに識別する最小限のセットである必要はありません。
BaseServerName	SADaReader が使用するデータベースサーバの名前。
BaseCatalogName	カラムが含まれているデータベース内のカタログの名前。値は常に DBNull です。
BaseColumnName	データベースのテーブル BaseTableName にあるカラムの元の名前。カラムが計算される場合、およびこの情報を特定できない場合は DBNull です。
BaseSchemaName	カラムが含まれているデータベース内のスキーマの名前。
BaseTableName	カラムが含まれているデータベースのテーブルの名前。カラムが計算される場合と、この情報を特定できない場合は、DBNull です。
DataType	この型のカラムに最適な .NET データ型。
AllowDBNull	カラムが NULL 入力可である場合は true、NULL 入力不可である場合またはこの情報を特定できない場合は false。
ProviderType	カラム型
IsAliased	カラム名がエイリアスの場合は true、エイリアスでない場合は false。
IsExpression	カラムが式の場合は true、カラム値の場合は false。
IsIdentity	カラムが identity カラムである場合は true、identity カラムでない場合は false。
IsAutoIncrement	カラムがオートインクリメントカラムまたはグローバルオートインクリメントカラムである場合は true、それ以外のカラムである場合 (または、この情報を特定できない場合) は false。

DataTable カラム	説明
IsRowVersion	書き込みできない永続的なロー識別子がカラムに含まれており、ローを識別する以外は意味を持たない値がカラムにある場合は true。
IsHidden	カラムが非表示の場合は true、表示される場合は false。
IsLong	カラムが long varchar、long nvarchar、または long binary の場合は true、それ以外の場合は false。
IsReadOnly	カラムが読み込み専用である場合は true、カラムが修正可能であるか、カラムのアクセス権を特定できない場合は false。

これらのカラムの詳細については、.NET Framework のマニュアルの SqlDataReader.GetSchemaTable を参照してください。

詳細については、「データのアクセスとデータの操作」を参照してください。

## 1.12.23 GetString(int) メソッド

指定されたカラムの値を文字列として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetString (ByVal ordinal As Integer) As String
```

#### C#

```
public override string GetString (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

## 備考

変換は行われなかったため、取り出されるデータはすでに文字列である必要があります。

SADeveloper.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

## 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

## 1.12.24 GetTimeSpan(int) メソッド

指定されたカラムの値を TimeSpan オブジェクトとして返します。

### 構文

#### Visual Basic

```
Public Function GetTimeSpan (ByVal ordinal As Integer) As TimeSpan
```

#### C#

```
public TimeSpan GetTimeSpan (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

カラムは、TIME データ型である必要があります。データは、TimeSpan に変換されます。TimeSpan の Days プロパティは常に 0 に設定されます。

SADeveloper.IsDBNull メソッドを呼び出して NULL 値を確認してから、このメソッドを呼び出します。

詳細については、「データのアクセスとデータの操作」を参照してください。

## 関連情報

[IsDBNull\(int\) メソッド \[237 ページ\]](#)

### 1.12.25 GetUInt16(int) メソッド

指定されたカラムの値を 16 ビット符号なし整数として返します。

#### 構文

##### Visual Basic

```
Public Function GetUInt16 (ByVal ordinal As Integer) As UShort
```

##### C#

```
public ushort GetUInt16 (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

変換は行われなため、取り出されるデータはすでに 16 ビット符号なし整数である必要があります。



## 1.12.26 GetUInt32(int) メソッド

指定されたカラムの値を 32 ビット符号なし整数として返します。

### 構文

#### Visual Basic

```
Public Function GetUInt32 (ByVal ordinal As Integer) As UInteger
```

#### C#

```
public uint GetUInt32 (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

指定されたカラムの値。

### 備考

変換は行われないため、取り出されるデータはすでに 32 ビット符号なし整数である必要があります。

## 1.12.27 GetUInt64(int) メソッド

指定されたカラムの値を 64 ビット符号なし整数として返します。

### 構文

#### Visual Basic

```
Public Function GetUInt64 (ByVal ordinal As Integer) As ULong
```

#### C#

```
public ulong GetUInt64 (int ordinal)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

## 戻り値

指定されたカラムの値。

## 備考

変換は行われなため、取り出されるデータはすでに 64 ビット符号なし整数である必要があります。

## 1.12.28 GetValue メソッド

指定されたカラムの値を Object として返します。

## オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override object	<a href="#">GetValue(int) [235 ページ]</a>	指定されたカラムの値を Object として返します。
public object	<a href="#">GetValue(int, long, int) [235 ページ]</a>	指定したカラムの値の部分文字列を Object として返します。

このセクションの内容:

### [GetValue\(int\) メソッド \[235 ページ\]](#)

指定されたカラムの値を Object として返します。

### [GetValue\(int, long, int\) メソッド \[235 ページ\]](#)

指定したカラムの値の部分文字列を Object として返します。

## 1.12.28.1 GetValue(int) メソッド

指定されたカラムの値を Object として返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetValue (ByVal ordinal As Integer) As Object
```

#### C#

```
public override object GetValue (int ordinal)
```

### パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

### 戻り値

オブジェクトとして返される指定されたカラムの値。

### 備考

このメソッドは、NULL データベースカラムに対して DBNull を返します。

## 1.12.28.2 GetValue(int, long, int) メソッド

指定したカラムの値の部分文字列を Object として返します。

### 構文

#### Visual Basic

```
Public Function GetValue (
    ByVal ordinal As Integer,
    ByVal index As Long,
    ByVal length As Integer
) As Object
```

C#

```
public object GetValue (  
    int ordinal,  
    long index,  
    int length  
)
```

## パラメータ

**ordinal** 値を取得するカラムを表す序数。番号は 0 から始まります。

**index** 取得される値の部分文字列の、0 から始まるインデックス。

**length** 取得される値の部分文字列の長さ。

## 戻り値

部分文字列の値はオブジェクトとして返されます。

## 備考

このメソッドは、NULL データベースカラムに対して DBNull を返します。

## 1.12.29 GetValues(object[]) メソッド

現在のローのすべてのカラムを取得します。

### 構文

#### Visual Basic

```
Public Overrides Function GetValues (ByVal values As Object()) As Integer
```

#### C#

```
public override unsafe int GetValues (object[] values)
```

## パラメータ

**values** 結果セットのロー全体を保持するオブジェクトの配列。

## 戻り値

配列内のオブジェクトの数。

## 備考

ほとんどのアプリケーションについて、GetValues メソッドは、各カラムを個々に取り出すのではなく、すべてのカラムを取り出す効率的な方法を提供します。

結果のローに含まれるカラムの数より少ないカラムが含まれる Object 配列を渡すことができます。Object 配列が保持するデータ量のみが配列にコピーされます。また、結果のローに含まれるカラムの数より長い Object 配列を渡すこともできます。

このメソッドは、NULL データベースカラムに対して DBNull を返します。

## 1.12.30 IsDBNull(int) メソッド

カラムに NULL 値が含まれるかどうかを示す値を返します。

### 構文

#### Visual Basic

```
Public Overrides Function IsDBNull (ByVal ordinal As Integer) As Boolean
```

#### C#

```
public override bool IsDBNull (int ordinal)
```

## パラメータ

**ordinal** カラムを表す、0 から始まる序数。

## 戻り値

指定されたカラム値が DBNull と等しい場合は true、等しくない場合は false を返します。

## 備考

入力された取得メソッド (GetByte、GetChar など) を呼び出す前に、このメソッドを呼び出して NULL カラム値を確認し、例外が発生しないようにします。

### 1.12.31 myDispose() メソッド

オブジェクトに関連付けられているリソースを解放します。

#### 構文

##### Visual Basic

```
Public Sub myDispose ()
```

##### C#

```
public void myDispose ()
```

### 1.12.32 NextResult() メソッド

複数の結果セットを返すクエリを処理するとき、SADataReader を次の結果セットに進めます。

#### 構文

##### Visual Basic

```
Public Overrides Function NextResult () As Boolean
```

##### C#

```
public override bool NextResult ()
```

## 戻り値

さらに結果セットがある場合は true、そうでない場合は false。

## 備考

バッチ SQL 文またはストアプロシージャを実行して生成できる複数の結果セットを処理するために使用されます。デフォルトでは、データリーダーは最初の結果セットの位置にあります。

### 1.12.33 Read() メソッド

結果セットの次のローを読み込み、SADeader をこのローに移動します。

#### 構文

##### Visual Basic

```
Public Overrides Function Read () As Boolean
```

##### C#

```
public override unsafe bool Read ()
```

## 戻り値

さらにローがある場合は true、そうでない場合は false。

## 備考

SADeader のデフォルト位置は、最初のレコードより前です。任意のデータにアクセスするには Read を呼び出します。

#### 例

次のコードは、結果の単一カラムの値をリストボックスに設定します。

```
while( reader.Read() ) {  
    listResults.Items.Add( reader.GetValue( 0 ).ToString() );  
}  
listResults.EndUpdate();  
reader.Close();
```

## 1.12.34 Depth プロパティ

現在のローのネストの深さを示す値を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Depth As Integer
```

#### C#

```
public override int Depth {get;}
```

### 備考

最も外側のテーブルの深さは 0 です。

現在のローのネストの深さ。

## 1.12.35 FieldCount プロパティ

結果セット内のカラム数を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property FieldCount As Integer
```

#### C#

```
public override int FieldCount {get;}
```

### 備考

現在のレコード内のカラム数。



## 1.12.36 HasRows プロパティ

SADaReader に 1 つ以上のローが含まれるかどうかを示す値を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property HasRows As Boolean
```

#### C#

```
public override bool HasRows {get;}
```

### 備考

SADaReader に 1 つ以上のローが含まれる場合は true、含まれない場合は false です。

## 1.12.37 IsClosed プロパティ

SADaReader が閉じているかどうかを示す値を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property IsClosed As Boolean
```

#### C#

```
public override bool IsClosed {get;}
```

### 備考

SADaReader が閉じられている場合は true、閉じられていない場合は false です。

SADaReader が閉じられた後に使用できるプロパティは、IsClosed と RecordsAffected のみです。

## 1.12.38 RecordsAffected プロパティ

SQL 文の実行によって変更、挿入、または削除されたローの数。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property RecordsAffected As Integer
```

#### C#

```
public override int RecordsAffected {get;}
```

### 備考

変更、挿入、または削除されたローの数。この値は、ローが影響されなかったり文が失敗した場合は 0、SELECT 文の場合は -1 です。

変更、挿入、または削除されたローの数。この値は、ローが影響されなかったり文が失敗した場合は 0、SELECT 文の場合は -1 です。

このプロパティの値は累積されます。たとえば、2 つのレコードがバッチモードで挿入された場合、RecordsAffected の値は 2 になります。

SADaReader が閉じられた後に使用できるプロパティは、IsClosed と RecordsAffected のみです。

## 1.12.39 this プロパティ

カラムの値をネイティブフォーマットで返します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override object	<a href="#">this[int index] [243 ページ]</a>	カラムの値をネイティブフォーマットで返します。
public override object	<a href="#">this[string name] [243 ページ]</a>	カラムの値をネイティブフォーマットで返します。

このセクションの内容:

[this\[int index\] プロパティ \[243 ページ\]](#)

カラムの値をネイティブフォーマットで返します。

[this\[string name\] プロパティ \[243 ページ\]](#)

カラムの値をネイティブフォーマットで返します。

### 1.12.39.1 this[int index] プロパティ

カラムの値をネイティブフォーマットで返します。

#### 構文

##### Visual Basic

```
Public ReadOnly Overrides Property Item (ByVal index As Integer) As Object
```

##### C#

```
public override object this[int index] {get;}
```

#### 備考

C# では、このプロパティは `SADatReader` クラスのインデクサです。

### 1.12.39.2 this[string name] プロパティ

カラムの値をネイティブフォーマットで返します。

#### 構文

##### Visual Basic

```
Public ReadOnly Overrides Property Item (ByVal name As String) As Object
```

##### C#

```
public override object this[string name] {get;}
```

#### 備考

C# では、このプロパティは `SADatReader` クラスのインデクサです。

## 1.13 SDataSourceEnumerator クラス

ローカルネットワーク内で有効なデータベースサーバのすべてのインスタンスを列挙するメカニズムを提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SDataSourceEnumerator Inherits  
System.Data.Common.DbDataSourceEnumerator
```

#### C#

```
public sealed class SDataSourceEnumerator :  
System.Data.Common.DbDataSourceEnumerator
```

### メンバー

SDataSourceEnumerator のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### メソッド

変更子とタイプ	メソッド	説明
public unsafe override DataTable	<a href="#">GetDataSources() [245 ページ]</a>	参照可能なすべてのデータベースサーバに関する情報が含まれる DataTable を取得します。

#### プロパティ

変更子とタイプ	プロパティ	説明
public SDataSourceEnumerator	<a href="#">Instance [245 ページ]</a>	SDataSourceEnumerator のインスタンスを取得します。これを使用すると、すべての表示可能なデータベースサーバを取得できます。

### 備考

SDataSourceEnumerator にはコンストラクタがありません。

このセクションの内容:

[GetDataSources\(\) メソッド \[245 ページ\]](#)

参照可能なすべてのデータベースサーバに関する情報が含まれる DataTable を取得します。

[Instance プロパティ \[245 ページ\]](#)

SADDataSourceEnumerator のインスタンスを取得します。これを使用すると、すべての表示可能なデータベースサーバを取得できます。

## 1.13.1 GetDataSources() メソッド

参照可能なすべてのデータベースサーバに関する情報が含まれる DataTable を取得します。

### 構文

#### Visual Basic

```
Public Overrides Function GetDataSources () As DataTable
```

#### C#

```
public unsafe override DataTable GetDataSources ()
```

### 備考

返されるテーブルは、ServerName、IPAddress、PortNumber、および DataBaseNames という 4 つのカラムで構成されています。テーブルには、有効なデータベースサーバごとに 1 つのローがあります。

### 例

次のコードは、有効な各データベースサーバに関する情報を DataTable に設定します。

```
DataTable servers = SADDataSourceEnumerator.Instance.GetDataSources();
```

## 1.13.2 Instance プロパティ

SADDataSourceEnumerator のインスタンスを取得します。これを使用すると、すべての表示可能なデータベースサーバを取得できます。

### 構文

#### Visual Basic

```
Public Shared ReadOnly Property Instance As SADDataSourceEnumerator
```

#### C#

```
public SADDataSourceEnumerator Instance {get;}
```

## 1.14 SADefault クラス

デフォルト値が設定されたパラメータを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SADefault
```

#### C#

```
public sealed class SADefault
```

### メンバー

SADefault のすべてのメンバー (継承されたメンバーも含まれます) を以下に示します。

#### 変数

変更子とタイプ	変数	説明
public static readonly SADefault	Value	デフォルトパラメータの値を取得します。 このフィールドは読み込み専用で静的です。

### 備考

SADefault にはコンストラクタがありません。

```
SAParameter parm = new SAParameter();  
parm.Value = SADefault.Value;
```

## 1.15 SAError クラス

データソースによって返された警告またはエラーに関する情報を収集します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAError
```

## C#

```
public sealed class SAError
```

## メンバー

SAError のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変更子とタイプ	メソッド	説明
public override string	<a href="#">ToString() [248 ページ]</a>	エラーメッセージの完全なテキストです。

### プロパティ

変更子とタイプ	プロパティ	説明
public string	<a href="#">Message [248 ページ]</a>	エラーの簡単な説明を返します。
public int	<a href="#">NativeError [248 ページ]</a>	データベース固有のエラー情報を返します。
public string	<a href="#">Source [249 ページ]</a>	エラーを生成したプロバイダの名前を返します。
public string	<a href="#">SqlState [249 ページ]</a>	ANSI SQL 規格に準拠する 5 文字の SQLSTATE です。

## 備考

SAError にはコンストラクタがありません。

エラー処理については、「[.NET エラー処理](#)」を参照してください。

このセクションの内容:

#### [ToString\(\) メソッド \[248 ページ\]](#)

エラーメッセージの完全なテキストです。

#### [Message プロパティ \[248 ページ\]](#)

エラーの簡単な説明を返します。

#### [NativeError プロパティ \[248 ページ\]](#)

データベース固有のエラー情報を返します。

#### [Source プロパティ \[249 ページ\]](#)

エラーを生成したプロバイダの名前を返します。

#### [SqlState プロパティ \[249 ページ\]](#)

ANSI SQL 規格に準拠する 5 文字の SQLSTATE です。

## 1.15.1 ToString() メソッド

エラーメッセージの完全なテキストです。

### 構文

#### Visual Basic

```
Public Overrides Function ToString () As String
```

#### C#

```
public override string ToString ()
```

### 例

戻り値は、*SAError*: の形式の文字列で、後ろにメッセージが続きます。次に例を示します。

```
SAError:UserId or Password not valid.
```

## 1.15.2 Message プロパティ

エラーの簡単な説明を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Message As String
```

#### C#

```
public string Message {get;}
```

## 1.15.3 NativeError プロパティ

データベース固有のエラー情報を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property NativeError As Integer
```

#### C#

```
public int NativeError {get;}
```



## 1.15.4 Source プロパティ

エラーを生成したプロバイダの名前を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Source As String
```

#### C#

```
public string Source {get;}
```

## 1.15.5 SqlState プロパティ

ANSI SQL 規格に準拠する 5 文字の SQLSTATE です。

### 構文

#### Visual Basic

```
Public ReadOnly Property SqlState As String
```

#### C#

```
public string SqlState {get;}
```

## 1.16 SAErrorCollection クラス

.NET データプロバイダによって生成されたすべてのエラーを収集します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAErrorCollection Implements  
System.Collections.ICollection, System.Collections.IEnumerable
```

#### C#

```
public sealed class SAErrorCollection : System.Collections.ICollection,  
System.Collections.IEnumerable
```

## メンバー

SAErrorCollection のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変更子とタイプ	メソッド	説明
public void	<a href="#">CopyTo(Array, int) [251 ページ]</a>	配列内の特定のインデックスから開始して SAErrorCollection の要素を配列にコピーします。
public IEnumerator	<a href="#">GetEnumerator() [251 ページ]</a>	SAErrorCollection で反復処理する列挙子を返します。

### プロパティ

変更子とタイプ	プロパティ	説明
public int	<a href="#">Count [252 ページ]</a>	コレクション内のエラーの数を返します。
public SAError	<a href="#">this[int index] [252 ページ]</a>	指定されたインデックス位置のエラーを返します。

## 備考

SAErrorCollection にはコンストラクタがありません。通常、SAErrorCollection は SAException.Errors プロパティから取得されます。

実装: ICollection、IEnumerable

エラー処理については、「[.NET エラー処理](#)」を参照してください。

このセクションの内容:

#### [CopyTo\(Array, int\) メソッド \[251 ページ\]](#)

配列内の特定のインデックスから開始して SAErrorCollection の要素を配列にコピーします。

#### [GetEnumerator\(\) メソッド \[251 ページ\]](#)

SAErrorCollection で反復処理する列挙子を返します。

#### [Count プロパティ \[252 ページ\]](#)

コレクション内のエラーの数を返します。

#### [this\[int index\] プロパティ \[252 ページ\]](#)

指定されたインデックス位置のエラーを返します。

## 関連情報

[Errors プロパティ \[255 ページ\]](#)

## 1.16.1 CopyTo(Array, int) メソッド

配列内の特定のインデックスから開始して SAErrorCollection の要素を配列にコピーします。

### 構文

#### Visual Basic

```
Public Sub CopyTo (  
    ByVal array As Array,  
    ByVal index As Integer  
)
```

#### C#

```
public void CopyTo (  
    Array array,  
    int index  
)
```

### パラメータ

**array** 要素のコピー先となる配列。

**index** 配列の開始インデックス。

## 1.16.2 GetEnumerator() メソッド

SAErrorCollection で反復処理する列挙子を返します。

### 構文

#### Visual Basic

```
Public Function GetEnumerator () As System.Collections.IEnumerator
```

#### C#

```
public IEnumerator GetEnumerator ()
```

### 戻り値

SAErrorCollection の System.Collections.IEnumerator。

## 1.16.3 Count プロパティ

コレクション内のエラーの数を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Count As Integer
```

#### C#

```
public int Count {get;}
```

## 1.16.4 this[int index] プロパティ

指定されたインデックス位置のエラーを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Item (ByVal index As Integer) As SAError
```

#### C#

```
public SAError this[int index] {get;}
```

### 備考

指定されたインデックス位置のエラーが含まれる SAError オブジェクト。

### 関連情報

[SAError クラス \[246 ページ\]](#)

## 1.17 SAException クラス

データベースサーバが警告またはエラーを返したときにスローされる例外です。

### 構文

#### Visual Basic

```
Public Class SAException Inherits System.Exception
```

#### C#

```
public class SAException : System.Exception
```

### メンバー

SAException のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### メソッド

変数とタイプ	メソッド	説明
public override void	<a href="#">GetObjectData(SerializationInfo, StreamingContext) [254 ページ]</a>	SerializationInfo に例外に関する情報を設定します。

#### プロパティ

変数とタイプ	プロパティ	説明
public SAErrorCollection	<a href="#">Errors [255 ページ]</a>	1つまたは複数の SAError オブジェクトのコレクションを返します。
public override string	<a href="#">Message [255 ページ]</a>	エラーが記述されたテキストを返します。
public int	<a href="#">NativeError [256 ページ]</a>	データベース固有のエラー情報を返します。
public override string	<a href="#">Source [256 ページ]</a>	エラーを生成したプロバイダの名前を返します。

### 備考

SAException にはコンストラクタがありません。通常、SAException オブジェクトは catch 内で宣言されます。次に例を示します。

```
...
catch( SAException ex )
{
    MessageBox.Show( ex.Errors[0].Message, "Error" );
}
```

エラー処理については、「.NET エラー処理」を参照してください。

このセクションの内容:

[GetObjectData\(SerializationInfo, StreamingContext\) メソッド \[254 ページ\]](#)

SerializationInfo に例外に関する情報を設定します。

[Errors プロパティ \[255 ページ\]](#)

1 つまたは複数の SAError オブジェクトのコレクションを返します。

[Message プロパティ \[255 ページ\]](#)

エラーが記述されたテキストを返します。

[NativeError プロパティ \[256 ページ\]](#)

データベース固有のエラー情報を返します。

[Source プロパティ \[256 ページ\]](#)

エラーを生成したプロバイダの名前を返します。

## 1.17.1 GetObjectData(SerializationInfo, StreamingContext) メソッド

SerializationInfo に例外に関する情報を設定します。

### 構文

#### Visual Basic

```
Public Overrides Sub GetObjectData (  
    ByVal info As SerializationInfo,  
    ByVal context As StreamingContext  
)
```

#### C#

```
public override void GetObjectData (  
    SerializationInfo info,  
    StreamingContext context  
)
```

### パラメータ

**info** スローされた例外に関する直列化形式のオブジェクトデータを保持する SerializationInfo。

**context** ソースまたは送信先に関するコンテキスト情報が含まれる StreamingContext。

## 備考

Exception.GetObjectData を上書きします。

## 1.17.2 Errors プロパティ

1つまたは複数の SAError オブジェクトのコレクションを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Errors As SAErrorCollection
```

#### C#

```
public SAErrorCollection Errors {get;}
```

## 備考

SAErrorCollection オブジェクトには常に少なくとも1つの SAError オブジェクトのインスタンスがあります。

## 関連情報

[SAErrorCollection クラス \[249 ページ\]](#)

[SAError クラス \[246 ページ\]](#)

## 1.17.3 Message プロパティ

エラーが記述されたテキストを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Message As String
```

#### C#

```
public override string Message {get;}
```

## 備考

このメソッドは、Errors コレクション内のすべての SAEError オブジェクトのすべての Message プロパティの連結が含まれる単一文字列を返します。最後のメッセージを除く各メッセージの後ろには復帰文字があります。

## 関連情報

[SAError クラス \[246 ページ\]](#)

## 1.17.4 NativeError プロパティ

データベース固有のエラー情報を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property NativeError As Integer
```

#### C#

```
public int NativeError {get;}
```

## 1.17.5 Source プロパティ

エラーを生成したプロバイダの名前を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Source As String
```

#### C#

```
public override string Source {get;}
```



## 1.18 SAFactory クラス

データソースクラスの Sap.Data.SQAnywhere プロバイダの実装のインスタンスを作成するメソッドのセットを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAFactory Inherits  
System.Data.Common.DbProviderFactory
```

#### C#

```
public sealed class SAFactory : System.Data.Common.DbProviderFactory
```

### メンバー

SAFactory のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### 変数

変数とタイプ	変数	説明
public static readonly SAFactory	Instance	SAFactory クラスのシングルトンインスタンスを表します。  SAFactory はシングルトンクラスです。つまり、このクラスのインスタンスとして存在できるのは、このインスタンスのみです。  通常、このフィールドを直接使用することはありません。代わりに、SAFactory のこのインスタンスへの参照を取得するには、System.Data.Common.DbProviderFactories.GetFactory(String) を使用します。例については、SAFactory の説明を参照してください。

#### メソッド

変数とタイプ	メソッド	説明
public override DbCommand	<a href="#">CreateCommand() [259 ページ]</a>	厳密に型指定された System.Data.Common.DbCommand インスタンスを返します。
public override DbCommandBuilder	<a href="#">CreateCommandBuilder() [260 ページ]</a>	厳密に型指定された System.Data.Common.DbCommandBuilder インスタンスを返します。

変更子とタイプ	メソッド	説明
public override DbConnection	<a href="#">CreateConnection()</a> [260 ページ]	厳密に型指定された System.Data.Common.DbConnection インスタンスを返します。
public override DbConnectionStringBuilder	<a href="#">CreateConnectionStringBuilder()</a> [261 ページ]	厳密に型指定された System.Data.Common.DbConnectionStringBuilder インスタンスを返します。
public override DbDataAdapter	<a href="#">CreateDataAdapter()</a> [262 ページ]	厳密に型指定された System.Data.Common.DbDataAdapter インスタンスを返します。
public override DbDataSourceEnumerator	<a href="#">CreateDataSourceEnumerator()</a> [262 ページ]	厳密に型指定された System.Data.Common.DbDataSourceEnumerator インスタンスを返します。
public override DbParameter	<a href="#">CreateParameter()</a> [263 ページ]	厳密に型指定された System.Data.Common.DbParameter インスタンスを返します。
public override CodeAccessPermission	<a href="#">CreatePermission(PermissionState)</a> [263 ページ]	厳密に型指定された CodeAccessPermission インスタンスを返します。

## プロパティ

変更子とタイプ	プロパティ	説明
public override bool	<a href="#">CanCreateDataSourceEnumerator</a> [264 ページ]	常に true を返します。これは、SADataSourceEnumerator オブジェクトを作成できることを示しています。

## 備考

SAFactory にはコンストラクタがありません。

DbProviderFactories と DbProviderFactory により、プロバイダに依存しないコードを簡単に作成できるようになりました。GetFactory に渡されるプロバイダの不変名として Sap.Data.SQLiteAnywhere を指定します。次に例を示します。

```
' Visual Basic
Dim factory As DbProviderFactory = _
    DbProviderFactories.GetFactory( "Sap.Data.SQLiteAnywhere" )
Dim conn As DbConnection = _
    factory.CreateConnection()
// C#
DbProviderFactory factory =
    DbProviderFactories.GetFactory("Sap.Data.SQLiteAnywhere" );
DbConnection conn = factory.CreateConnection();
```

この例の中の conn は、SAConnection オブジェクトとして作成されます。

ADO.NET におけるプロバイダファクトリと汎用プログラミングについては、[Generic Coding with the ADO.NET 2.0 Base Classes and Factories](#) を参照してください。

このセクションの内容:

[CreateCommand\(\) メソッド \[259 ページ\]](#)

厳密に型指定された System.Data.Common.DbCommand インスタンスを返します。

[CreateCommandBuilder\(\) メソッド \[260 ページ\]](#)

厳密に型指定された System.Data.Common.DbCommandBuilder インスタンスを返します。

[CreateConnection\(\) メソッド \[260 ページ\]](#)

厳密に型指定された System.Data.Common.DbConnection インスタンスを返します。

[CreateConnectionStringBuilder\(\) メソッド \[261 ページ\]](#)

厳密に型指定された System.Data.Common.DbConnectionStringBuilder インスタンスを返します。

[CreateDataAdapter\(\) メソッド \[262 ページ\]](#)

厳密に型指定された System.Data.Common.DbDataAdapter インスタンスを返します。

[CreateDataSourceEnumerator\(\) メソッド \[262 ページ\]](#)

厳密に型指定された System.Data.Common.DbDataSourceEnumerator インスタンスを返します。

[CreateParameter\(\) メソッド \[263 ページ\]](#)

厳密に型指定された System.Data.Common.DbParameter インスタンスを返します。

[CreatePermission\(PermissionState\) メソッド \[263 ページ\]](#)

厳密に型指定された CodeAccessPermission インスタンスを返します。

[CanCreateDataSourceEnumerator プロパティ \[264 ページ\]](#)

常に true を返します。これは、SADDataSourceEnumerator オブジェクトを作成できることを示しています。

## 1.18.1 CreateCommand() メソッド

厳密に型指定された System.Data.Common.DbCommand インスタンスを返します。

### 構文

#### Visual Basic

```
Public Overrides Function CreateCommand () As DbCommand
```

#### C#

```
public override DbCommand CreateCommand ()
```

### 戻り値

DbCommand として型指定された新しい SACommand オブジェクト。

## 関連情報

[SACommand クラス \[53 ページ\]](#)

### 1.18.2 CreateCommandBuilder() メソッド

厳密に型指定された System.Data.Common.DbCommandBuilder インスタンスを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function CreateCommandBuilder () As DbCommandBuilder
```

##### C#

```
public override DbCommandBuilder CreateCommandBuilder ()
```

## 戻り値

DbCommandBuilder として型指定された新しい SACommandBuilder オブジェクト。

## 関連情報

[SACommandBuilder クラス \[91 ページ\]](#)

### 1.18.3 CreateConnection() メソッド

厳密に型指定された System.Data.Common.DbConnection インスタンスを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function CreateConnection () As DbConnection
```

##### C#

```
public override DbConnection CreateConnection ()
```

## 戻り値

DbConnection として型指定された新しい SAcConnection オブジェクト。

## 関連情報

[SAcConnection クラス \[118 ページ\]](#)

### 1.18.4 CreateConnectionStringBuilder() メソッド

厳密に型指定された System.Data.Common.DbConnectionStringBuilder インスタンスを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function CreateConnectionStringBuilder () As  
    DbConnectionStringBuilder
```

##### C#

```
public override DbConnectionStringBuilder CreateConnectionStringBuilder ()
```

## 戻り値

DbConnectionStringBuilder として型指定された新しい SAcConnectionStringBuilder オブジェクト。

## 関連情報

[SAcConnectionStringBuilder クラス \[147 ページ\]](#)

## 1.18.5 CreateDataAdapter() メソッド

厳密に型指定された System.Data.Common.DbDataAdapter インスタンスを返します。

### 構文

#### Visual Basic

```
Public Overrides Function CreateDataAdapter () As DbDataAdapter
```

#### C#

```
public override DbDataAdapter CreateDataAdapter ()
```

### 戻り値

DbDataAdapter として型指定された新しい SDataAdapter オブジェクト。

### 関連情報

[SDataAdapter クラス \[181 ページ\]](#)

## 1.18.6 CreateDataSourceEnumerator() メソッド

厳密に型指定された System.Data.Common.DbDataSourceEnumerator インスタンスを返します。

### 構文

#### Visual Basic

```
Public Overrides Function CreateDataSourceEnumerator () As  
DbDataSourceEnumerator
```

#### C#

```
public override DbDataSourceEnumerator CreateDataSourceEnumerator ()
```

### 戻り値

DbDataSourceEnumerator として型指定された新しい SDataSourceEnumerator オブジェクト。

## 関連情報

[SADataSourceEnumerator クラス \[244 ページ\]](#)

### 1.18.7 CreateParameter() メソッド

厳密に型指定された System.Data.Common.DbParameter インスタンスを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function CreateParameter () As DbParameter
```

##### C#

```
public override DbParameter CreateParameter ()
```

## 戻り値

DbParameter として型指定された新しい SAPParameter オブジェクト。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

### 1.18.8 CreatePermission(PermissionState) メソッド

厳密に型指定された CodeAccessPermission インスタンスを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function CreatePermission (ByVal state As PermissionState) As CodeAccessPermission
```

##### C#

```
public override CodeAccessPermission CreatePermission (PermissionState state)
```

## パラメータ

**state** System.Security.Permissions.PermissionState 列挙のメンバー。

## 戻り値

CodeAccessPermission として型指定された新しい SAPermission オブジェクト。

## 関連情報

[SACommand クラス \[53 ページ\]](#)

## 1.18.9 CanCreateDataSourceEnumerator プロパティ

常に true を返します。これは、SADataSourceEnumerator オブジェクトを作成できることを示しています。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property CanCreateDataSourceEnumerator As Boolean
```

#### C#

```
public override bool CanCreateDataSourceEnumerator {get;}
```

## 戻り値

true を返します。これは、SADataSourceEnumerator オブジェクトを作成できることを示しています。

## 関連情報

[SADataSourceEnumerator クラス \[244 ページ\]](#)

[SACommand クラス \[53 ページ\]](#)



## 1.19 SAInfoMessageEventArgs クラス

InfoMessage イベントのデータを提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAInfoMessageEventArgs Inherits System.EventArgs
```

#### C#

```
public sealed class SAInfoMessageEventArgs : System.EventArgs
```

### メンバー

SAInfoMessageEventArgs のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### メソッド

変更子とタイプ	メソッド	説明
public override string	<a href="#">ToString() [266 ページ]</a>	InfoMessage イベントの文字列表現を取り出します。

#### プロパティ

変更子とタイプ	プロパティ	説明
public SAErrorCollection	<a href="#">Errors [266 ページ]</a>	データソースから送信されたメッセージのコレクションを返します。
public string	<a href="#">Message [267 ページ]</a>	データソースから送信されたエラーの完全なテキストを返します。
public SAMessageType	<a href="#">MessageType [267 ページ]</a>	メッセージのタイプを返します。
public int	<a href="#">NativeError [268 ページ]</a>	データベースによって返される SQLCODE を返します。
public string	<a href="#">Source [268 ページ]</a>	.NET データプロバイダの名前を返します。

### 備考

SAInfoMessageEventArgs にはコンストラクタがありません。

このセクションの内容:

[ToString\(\) メソッド \[266 ページ\]](#)

InfoMessage イベントの文字列表現を取り出します。

#### [Errors プロパティ \[266 ページ\]](#)

データソースから送信されたメッセージのコレクションを返します。

#### [Message プロパティ \[267 ページ\]](#)

データソースから送信されたエラーの完全なテキストを返します。

#### [MessageType プロパティ \[267 ページ\]](#)

メッセージのタイプを返します。

#### [NativeError プロパティ \[268 ページ\]](#)

データベースによって返される SQLCODE を返します。

#### [Source プロパティ \[268 ページ\]](#)

.NET データプロバイダの名前を返します。

## 1.19.1 ToString() メソッド

InfoMessage イベントの文字列表現を取り出します。

### 構文

#### Visual Basic

```
Public Overrides Function ToString () As String
```

#### C#

```
public override string ToString ()
```

### 戻り値

InfoMessage イベントを表す文字列。

## 1.19.2 Errors プロパティ

データソースから送信されたメッセージのコレクションを返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Errors As SAErrorCollection
```

**C#**

```
public SAErrorCollection Errors {get;}
```

### 1.19.3 Message プロパティ

データソースから送信されたエラーの完全なテキストを返します。

 構文

**Visual Basic**

```
Public ReadOnly Property Message As String
```

**C#**

```
public string Message {get;}
```

### 1.19.4 MessageType プロパティ

メッセージのタイプを返します。

 構文

**Visual Basic**

```
Public ReadOnly Property MessageType As SAMessageType
```

**C#**

```
public SAMessageType MessageType {get;}
```

**備考**

次のいずれかを指定してください。Action、Info、Status、または Warning。

## 1.19.5 NativeError プロパティ

データベースによって返される SQLCODE を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property NativeError As Integer
```

#### C#

```
public int NativeError {get;}
```

## 1.19.6 Source プロパティ

.NET データプロバイダの名前を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Property Source As String
```

#### C#

```
public string Source {get;}
```

## 1.20 SAMetaDataCollectionNames クラス

メタデータコレクションを取得する `SACConnection.GetSchema(string)` メソッドで使用する定数のリストを提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAMetaDataCollectionNames
```

#### C#

```
public sealed class SAMetaDataCollectionNames
```

## メンバー

SAMetaDataCollectionNames のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### 変数

変更子とタイプ	変数	説明
public static readonly string	Columns	Columns コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	DataSourceInformation	DataSourceInformation コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	DataTypes	DataTypes コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	ForeignKeys	ForeignKeys コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	IndexColumns	IndexColumns コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Indexes	Indexes コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	MetaDataCollections	MetaDataCollections コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	ProcedureParameters	ProcedureParameters コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Procedures	Procedures コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	ReservedWords	ReservedWords コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Restrictions	Restrictions コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Tables	Tables コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Users	Users コレクションを示す SAConnection.GetSchema(string) メソッドで使用する定数を提供します。

変更子とタイプ	変数	説明
public static readonly string	UserDefinedTypes	UserDefinedTypes コレクションを示す SACConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	ViewColumns	ViewColumns コレクションを示す SACConnection.GetSchema(string) メソッドで使用する定数を提供します。
public static readonly string	Views	Views コレクションを示す SACConnection.GetSchema(string) メソッドで使用する定数を提供します。

## 備考

このフィールドは定数で、読み込み専用です。

## 関連情報

[GetSchema\(string\) メソッド \[135 ページ\]](#)

## 1.21 SAPparameter クラス

SACCommand のパラメータと必要に応じて DataSet カラムへのマッピングを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAPparameter Inherits
System.Data.Common.DbParameter Implements System.ICloneable
```

#### C#

```
public sealed class SAPparameter : System.Data.Common.DbParameter,
System.ICloneable
```

## メンバー

SAPparameter のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

## コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SAPParameter [273 ページ]</a>	値として NULL (Visual Basic の場合は Nothing) を使用して、SAPParameter オブジェクトを初期化します。

## メソッド

変更子とタイプ	メソッド	説明
public override void	<a href="#">ResetDbType() [278 ページ]</a>	この SAPParameter に関連付けられている型 (DbType および SADBType の値) をリセットします。
public override string	<a href="#">ToString() [279 ページ]</a>	ParameterName が含まれる文字列を返します。

## プロパティ

変更子とタイプ	プロパティ	説明
public override DbType	<a href="#">DbType [279 ページ]</a>	パラメータの DbType を取得または設定します。
public override ParameterDirection	<a href="#">Direction [280 ページ]</a>	パラメータが入力専用、出力専用、双方向性、またはストアドプロシージャ戻り値パラメータのいずれであるかを示す値を取得または設定します。
public override bool	<a href="#">IsNullable [280 ページ]</a>	パラメータが NULL 値を受け入れるかどうかを示す値を取得または設定します。
public int	<a href="#">Offset [281 ページ]</a>	Value プロパティのオフセットを取得または設定します。
public override string	<a href="#">ParameterName [281 ページ]</a>	SAPParameter の名前を取得または設定します。
public byte	<a href="#">Precision [282 ページ]</a>	Value プロパティを表すために使用される最大桁数を取得または設定します。
public SADBType	<a href="#">SADBType [282 ページ]</a>	パラメータの SADBType です。
public byte	<a href="#">Scale [283 ページ]</a>	Value が解析される小数点の桁の数を取得または設定します。
public override int	<a href="#">Size [283 ページ]</a>	カラム内のデータの最大サイズ (バイト単位) を取得または設定します。
public override string	<a href="#">SourceColumn [284 ページ]</a>	DataSet にマッピングされ、値をロードしたり返したりするときに使用するソースカラムの名前を取得または設定します。
public override bool	<a href="#">SourceColumnNullMapping [285 ページ]</a>	ソースカラムが NULL 入力可かどうかを示す値を取得または設定します。
public override DataRowVersion	<a href="#">SourceVersion [285 ページ]</a>	Value をロードするときに使用する DataRowVersion を取得または設定します。

変更子とタイプ	プロパティ	説明
public override object	Value [286 ページ]	パラメータの値を取得または設定します。

## 備考

実装: IDbDataParameter、IDataParameter、ICloneable

このセクションの内容:

### [SqlParameter コンストラクタ \[273 ページ\]](#)

値として NULL (Visual Basic の場合は Nothing) を使用して、SqlParameter オブジェクトを初期化します。

### [ResetDbType\(\) メソッド \[278 ページ\]](#)

この SqlParameter に関連付けられている型 (DbType および SADBType の値) をリセットします。

### [ToString\(\) メソッド \[279 ページ\]](#)

ParameterName が含まれる文字列を返します。

### [DbType プロパティ \[279 ページ\]](#)

パラメータの DbType を取得または設定します。

### [Direction プロパティ \[280 ページ\]](#)

パラメータが入力専用、出力専用、双方向性、またはストアドプロシージャ戻り値パラメータのいずれであるかを示す値を取得または設定します。

### [IsNullable プロパティ \[280 ページ\]](#)

パラメータが NULL 値を受け入れるかどうかを示す値を取得または設定します。

### [Offset プロパティ \[281 ページ\]](#)

Value プロパティのオフセットを取得または設定します。

### [ParameterName プロパティ \[281 ページ\]](#)

SqlParameter の名前を取得または設定します。

### [Precision プロパティ \[282 ページ\]](#)

Value プロパティを表すために使用される最大桁数を取得または設定します。

### [SADBType プロパティ \[282 ページ\]](#)

パラメータの SADBType です。

### [Scale プロパティ \[283 ページ\]](#)

Value が解析される小数点の桁の数を取得または設定します。

### [Size プロパティ \[283 ページ\]](#)

コラム内のデータの最大サイズ (バイト単位) を取得または設定します。

### [SourceColumn プロパティ \[284 ページ\]](#)

DataSet にマッピングされ、値をロードしたり返したりするときに使用するソースコラムの名前を取得または設定します。

### [SourceColumnNullMapping プロパティ \[285 ページ\]](#)

ソースコラムが NULL 入力可かどうかを示す値を取得または設定します。



[SourceVersion プロパティ \[285 ページ\]](#)

Value をロードするときに使用する DataRowVersion を取得または設定します。

[Value プロパティ \[286 ページ\]](#)

パラメータの値を取得または設定します。

## 1.21.1 SAPParameter コンストラクタ

値として NULL (Visual Basic の場合は Nothing) を使用して、SAPParameter オブジェクトを初期化します。

### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public	<a href="#">SAPParameter() [274 ページ]</a>	値として NULL (Visual Basic の場合は Nothing) を使用して、SAPParameter オブジェクトを初期化します。
public	<a href="#">SAPParameter(string, SADBType) [274 ページ]</a>	SAPParameter オブジェクトを、指定したパラメータ名、データ型と長さで初期化します。
public	<a href="#">SAPParameter(string, SADBType, int) [275 ページ]</a>	SAPParameter オブジェクトを、指定されたパラメータ名、データ型、サイズで初期化します。
public	<a href="#">SAPParameter(string, SADBType, int, ParameterDirection, bool, byte, byte, string, DataRowVersion, object) [276 ページ]</a>	指定されたパラメータ名、データ型、サイズ、方向、null 入力属性、数値精度、数値の位取り、ソースカラム、ソースのバージョン、値で、SAPParameter オブジェクトを初期化します。
public	<a href="#">SAPParameter(string, SADBType, int, string) [277 ページ]</a>	SAPParameter オブジェクトを、指定されたパラメータ名、データ型、サイズ、ソースカラムで初期化します。
public	<a href="#">SAPParameter(string, object) [278 ページ]</a>	SAPParameter オブジェクトを、指定したパラメータ名と値で初期化します。

このセクションの内容:

[SAPParameter\(\) コンストラクタ \[274 ページ\]](#)

値として NULL (Visual Basic の場合は Nothing) を使用して、SAPParameter オブジェクトを初期化します。

[SAPParameter\(string, SADBType\) コンストラクタ \[274 ページ\]](#)

SAPParameter オブジェクトを、指定したパラメータ名、データ型と長さで初期化します。

[SAPParameter\(string, SADBType, int\) コンストラクタ \[275 ページ\]](#)

SAPParameter オブジェクトを、指定されたパラメータ名、データ型、サイズで初期化します。

[SAPParameter\(string, SADBType, int, ParameterDirection, bool, byte, byte, string, DataRowVersion, object\) コンストラクタ \[276 ページ\]](#)

指定されたパラメータ名、データ型、サイズ、方向、null 入力属性、数値精度、数値の位取り、ソースカラム、ソースのバージョン、値で、SAPParameter オブジェクトを初期化します。

#### [SAPParameter\(string, SADBType, int, string\) コンストラクタ \[277 ページ\]](#)

SAPParameter オブジェクトを、指定されたパラメータ名、データ型、サイズ、ソースカラムで初期化します。

#### [SAPParameter\(string, object\) コンストラクタ \[278 ページ\]](#)

SAPParameter オブジェクトを、指定したパラメータ名と値で初期化します。

### 1.21.1.1 SAPParameter() コンストラクタ

値として NULL (Visual Basic の場合は Nothing) を使用して、SAPParameter オブジェクトを初期化します。

#### 構文

##### Visual Basic

```
Public Sub SAPParameter ()
```

##### C#

```
public SAPParameter ()
```

### 1.21.1.2 SAPParameter(string, SADBType) コンストラクタ

SAPParameter オブジェクトを、指定したパラメータ名、データ型と長さで初期化します。

#### 構文

##### Visual Basic

```
Public Sub SAPParameter (  
    ByVal parameterName As String,  
    ByVal dbType As SADBType  
)
```

##### C#

```
public SAPParameter (  
    string parameterName,  
    SADBType dbType  
)
```

## パラメータ

**parameterName** パラメータの名前。

**dbType** SADBType 値の 1 つ。

#### 例

次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 関連情報

[SADBType プロパティ \[282 ページ\]](#)

### 1.21.1.3 SAPParameter(string, SADBType, int) コンストラクタ

SAPParameter オブジェクトを、指定されたパラメータ名、データ型、サイズで初期化します。

#### 構文

##### Visual Basic

```
Public Sub SAPParameter (  
    ByVal parameterName As String,  
    ByVal dbType As SADBType,  
    ByVal size As Integer  
)
```

##### C#

```
public SAPParameter (  
    string parameterName,  
    SADBType dbType,  
    int size  
)
```

## パラメータ

**parameterName** パラメータの名前。

**dbType** SADBType 値の 1 つ。

**size** パラメータの長さ。

#### 例

次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 1.21.1.4 SAParameter(string, SADBType, int, ParameterDirection, bool, byte, byte, string, DataRowVersion, object) コンストラクタ

指定されたパラメータ名、データ型、サイズ、方向、null 入力属性、数値精度、数値の位取り、ソースカラム、ソースのバージョン、値で、SAParameter オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SAParameter (  
    ByVal parameterName As String,  
    ByVal dbType As SADBType,  
    ByVal size As Integer,  
    ByVal direction As ParameterDirection,  
    ByVal isNullable As Boolean,  
    ByVal precision As Byte,  
    ByVal scale As Byte,  
    ByVal sourceColumn As String,  
    ByVal sourceVersion As DataRowVersion,  
    ByVal value As Object  
)
```

#### C#

```
public SAParameter (  
    string parameterName,  
    SADBType dbType,  
    int size,  
    ParameterDirection direction,  
    bool isNullable,  
    byte precision,  
    byte scale,  
    string sourceColumn,  
    DataRowVersion sourceVersion,  
    object value  
)
```

### パラメータ

**parameterName** パラメータの名前。

**dbType** SADBType 値の 1 つ。

**size** パラメータの長さ。

**direction** ParameterDirection 値の 1 つ。

**isNullable** フィールドの値を NULL にできる場合は true、できない場合は false。

**precision** Value の解決に適用する、小数点以上と以下両方の総桁数。

**scale** Value の解決に適用する、小数点以下の桁数。

**sourceColumn** マッピング対象のソースカラムの名前。

**sourceVersion** DataRowVersion 値の 1 つ。

**value** パラメータの値である Object。

## 例

パラメータの名前。次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 1.21.1.5 SAParameter(string, SADBType, int, string) コンストラクタ

SAParameter オブジェクトを、指定されたパラメータ名、データ型、サイズ、ソースカラムで初期化します。

### 構文

#### Visual Basic

```
Public Sub SAParameter (  
    ByVal parameterName As String,  
    ByVal dbType As SADBType,  
    ByVal size As Integer,  
    ByVal sourceColumn As String  
)
```

#### C#

```
public SAParameter (  
    string parameterName,  
    SADBType dbType,  
    int size,  
    string sourceColumn  
)
```

## パラメータ

**parameterName** パラメータの名前。

**dbType** SADBType 値の 1 つ。

**size** パラメータの長さ。

**sourceColumn** マッピング対象のソースカラムの名前。

## 例

次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 1.21.1.6 SAParameter(string, object) コンストラクタ

SAParameter オブジェクトを、指定したパラメータ名と値で初期化します。

### 構文

#### Visual Basic

```
Public Sub SAParameter (  
    ByVal parameterName As String,  
    ByVal value As Object  
)
```

#### C#

```
public SAParameter (  
    string parameterName,  
    object value  
)
```

### パラメータ

**parameterName** パラメータの名前。

**value** パラメータの値である Object。

### 備考

このコンストラクタの使用はお奨めしません。これは、他のデータプロバイダとの互換性のために用意されています。

### 例

次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 1.21.2 ResetDbType() メソッド

この SAParameter に関連付けられている型 (DbType および SADBType の値) をリセットします。

### 構文

#### Visual Basic

```
Public Overrides Sub ResetDbType ()
```

**C#**

```
public override void ResetDbType ()
```

### 1.21.3 ToString() メソッド

ParameterName が含まれる文字列を返します。

 構文

**Visual Basic**

```
Public Overrides Function ToString () As String
```

**C#**

```
public override string ToString ()
```

戻り値

パラメータの名前。

### 1.21.4 DbType プロパティ

パラメータの DbType を取得または設定します。

 構文

**Visual Basic**

```
Public Overrides Property DbType As DbType
```

**C#**

```
public override DbType DbType {get;set;}
```

備考

SADbType と DbType はリンクされます。DbType を設定すると、サポートされている SADbType に SADbType を変更します。

この値は、SADbType 列挙のメンバーにする必要があります。

## 1.21.5 Direction プロパティ

パラメータが入力専用、出力専用、双方向性、またはストアードプロシージャ戻り値パラメータのいずれであるかを示す値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property Direction As ParameterDirection
```

#### C#

```
public override ParameterDirection Direction {get;set;}
```

### 備考

ParameterDirection 値の 1 つ。

ParameterDirection が出力である場合、関連付けられている SACommand を実行しても値は返らず、SAParameter には NULL 値が含まれます。最後の結果セットの最後のローが読み込まれると、Output、InputOut、ReturnValue パラメータが更新されます。

## 1.21.6 IsNullable プロパティ

パラメータが NULL 値を受け入れるかどうかを示す値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property IsNullable As Boolean
```

#### C#

```
public override bool IsNullable {get;set;}
```



## 備考

NULL 値が受け入れられる場合、この値は true です。受け入れられない場合は false です。デフォルトは false です。NULL 値は DBNull クラスを使用して処理されます。

### 1.21.7 Offset プロパティ

Value プロパティのオフセットを取得または設定します。

#### 構文

##### Visual Basic

```
Public Property Offset As Integer
```

##### C#

```
public int Offset {get;set;}
```

## 備考

値のオフセット。デフォルトは 0 です。

### 1.21.8 ParameterName プロパティ

SAPParameter の名前を取得または設定します。

#### 構文

##### Visual Basic

```
Public Overrides Property ParameterName As String
```

##### C#

```
public override string ParameterName {get;set;}
```

## 備考

デフォルトは、空の文字列です。

## 例

次の例では、パラメータ名は "param" です。

```
SELECT * FROM Customers WHERE ID = :param
```

## 1.21.9 Precision プロパティ

Value プロパティを表すために使用される最大桁数を取得または設定します。

### 構文

#### Visual Basic

```
Public Property Precision As Byte
```

#### C#

```
public byte Precision {get;set;}
```

### 備考

このプロパティの値は、Value プロパティを表すために使用される最大桁数です。デフォルト値は 0 です。これは、データプロバイダが Value プロパティの精度を設定することを示します。

Precision プロパティは、10 進数および数値入力パラメータに対してのみ使用されます。

## 1.21.10 SADBType プロパティ

パラメータの SADBType です。

### 構文

#### Visual Basic

```
Public Property SADBType As SADBType
```

#### C#

```
public SADBType SADBType {get;set;}
```

## 備考

SADbType と DbType はリンクされます。SADbType を設定すると、サポートされている DbType に DbType を変更します。

この値は、SADbType 列挙のメンバーにする必要があります。

### 1.21.11 Scale プロパティ

Value が解析される小数点の桁の数を取得または設定します。

#### 構文

##### Visual Basic

```
Public Property Scale As Byte
```

##### C#

```
public byte Scale {get;set;}
```

## 備考

Value が解析される小数点までの桁の数。デフォルトは 0 です。

Scale プロパティは、decimal と numeric 型の入カパラメータにのみ使用します。

### 1.21.12 Size プロパティ

カラム内のデータの最大サイズ (バイト単位) を取得または設定します。

#### 構文

##### Visual Basic

```
Public Overrides Property Size As Integer
```

##### C#

```
public override int Size {get;set;}
```

## 備考

このプロパティの値は、カラム内のデータの最大サイズ (バイト単位) です。デフォルト値はパラメータ値から推測されます。

このプロパティの値は、カラム内のデータの最大サイズ (バイト単位) です。デフォルト値はパラメータ値から推測されます。

Size プロパティは、バイナリと文字列型に対して使用されます。

可変長のデータ型の場合、Size プロパティは、データベースサーバに送信するデータの最大量を示します。たとえば、Size プロパティを使用して、データベースサーバに送信されるデータ量を文字列値の最初の 100 バイトに制限できます。

Size を明示的に設定しない場合、指定されたパラメータ値の実際のサイズから推測されます。固定幅のデータ型の場合、Size の値は無視されます。この値は情報用として取り出すことができ、プロバイダがパラメータの値をデータベースサーバに送信するときに使用する最大量を返します。

### 1.21.13 SourceColumn プロパティ

DataSet にマッピングされ、値をロードしたり返したりするときに使用するソースカラムの名前を取得または設定します。

#### 構文

##### Visual Basic

```
Public Overrides Property SourceColumn As String
```

##### C#

```
public override string SourceColumn {get;set;}
```

## 備考

DataSet にマッピングされ、値をロードしたり返したりするときに使用するソースカラムの名前を指定する文字列。

SourceColumn を空の文字列以外の値に設定すると、パラメータの値は SourceColumn 名を持つカラムから取り出されます。Direction を Input に設定すると、値は DataSet から取得されます。Direction を Output に設定すると、値はデータソースから取得されます。Direction が InputOutput の場合は Input と Output の両方です。

## 1.21.14 SourceColumnNullMapping プロパティ

ソースカラムが NULL 入力可かどうかを示す値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property SourceColumnNullMapping As Boolean
```

#### C#

```
public override bool SourceColumnNullMapping {get;set;}
```

### 備考

このプロパティによって、SACommandBuilder は、NULL 入力可のカラムに対して適切に Update 文を生成できます。

ソースカラムが NULL 入力可の場合は true、NULL 入力不可の場合は false が返されます。

## 1.21.15 SourceVersion プロパティ

Value をロードするときに使用する DataRowVersion を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property SourceVersion As DataRowVersion
```

#### C#

```
public override DataRowVersion SourceVersion {get;set;}
```

### 備考

Update オペレーション時に UpdateCommand によって使用され、パラメータ値を Current と Original のどちらに設定するかを決定します。このプロパティを使用してプライマリーキーを更新できます。このプロパティは、InsertCommand と DeleteCommand によって無視されます。このプロパティは、Item プロパティによって使用される DataRow のバージョン、または DataRow オブジェクトの GetChildRows メソッドに設定されます。

## 1.21.16 Value プロパティ

パラメータの値を取得または設定します。

### 構文

#### Visual Basic

```
Public Overrides Property Value As Object
```

#### C#

```
public override object Value {get;set;}
```

### 備考

パラメータの値を指定する Object。

入力パラメータの場合、この値は、データベースサーバに送信される SACommand のバウンド値です。取得および戻り値パラメータの場合、この値は、SADataReader が閉じられてから SACommand が完了したときに設定されます。

データベースサーバに NULL パラメータを送信する場合、NULL ではなく DBNull を指定してください。システム内では、NULL 値は値を持たない空のオブジェクトです。DBNull を使用して NULL 値を表します。

アプリケーションでデータベースタイプを指定する場合、.NET データプロバイダがデータをデータベースサーバに送信するときにバウンド値はこのタイプに変換されます。プロバイダは、IConvertible インタフェースをサポートしている場合、あらゆるタイプの値を変換しようとします。指定されたタイプと値の間に互換性がない場合、変換エラーが発生する可能性があります。

Value を設定して、DbType および SADbType プロパティの両方を推測できます。

Value プロパティは Update によって上書きされます。

## 1.22 SAParameterCollection クラス

SACommand オブジェクトのすべてのパラメータと必要に応じて DataSet カラムへのマッピングを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAParameterCollection Inherits  
System.Data.Common.DbParameterCollection
```

#### C#

```
public sealed class SAParameterCollection :  
System.Data.Common.DbParameterCollection
```

## メンバー

SAPParameterCollection のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### メソッド

変数とタイプ	メソッド	説明
public SAPParameter	<a href="#">Add [289 ページ]</a>	SAPParameter オブジェクトをこのコレクションに追加します。
public override void	<a href="#">AddRange [296 ページ]</a>	SAPParameterCollection の末尾に値の配列を追加します。
public SAPParameter	<a href="#">AddWithValue(string, object) [297 ページ]</a>	このコレクションの末尾に値を追加します。
public override void	<a href="#">Clear() [298 ページ]</a>	コレクションからすべての項目を削除します。
public override bool	<a href="#">Contains [298 ページ]</a>	コレクション内に SAPParameter オブジェクトがあるかどうかを示します。
public override void	<a href="#">CopyTo(Array, int) [300 ページ]</a>	SAPParameter オブジェクトを SAPParameterCollection から指定された配列にコピーします。
public override IEnumerator	<a href="#">GetEnumerator() [301 ページ]</a>	SAPParameterCollection で反復処理する列挙子を返します。
protected override DbParameter	<a href="#">GetParameter [302 ページ]</a>	SAPParameterCollection オブジェクトからのパラメータを返します。
public override int	<a href="#">IndexOf [304 ページ]</a>	コレクション内の SAPParameter オブジェクトのロケーションを返します。
public override void	<a href="#">Insert(int, object) [306 ページ]</a>	コレクション内の指定されたインデックス位置に SAPParameter オブジェクトを挿入します。
public override void	<a href="#">Remove(object) [306 ページ]</a>	指定された SAPParameter オブジェクトをコレクションから削除します。
public override void	<a href="#">RemoveAt [307 ページ]</a>	指定された SAPParameter オブジェクトをコレクションから削除します。
protected override void	<a href="#">SetParameter [308 ページ]</a>	SAPParameterCollection オブジェクトにパラメータを設定します。

### プロパティ

変数とタイプ	プロパティ	説明
public override int	<a href="#">Count [310 ページ]</a>	コレクション内の SAPParameter オブジェクトの数を返します。
public override bool	<a href="#">IsFixedSize [311 ページ]</a>	SAPParameterCollection のサイズが固定かどうかを示す値を取得します。
public override bool	<a href="#">IsReadOnly [311 ページ]</a>	SAPParameterCollection が読み込み専用かどうかを示す値を取得します。

変更子とタイプ	プロパティ	説明
public override bool	<a href="#">IsSynchronized [312 ページ]</a>	SAPParameterCollection オブジェクトが同期しているかどうかを示す値を取得します。
public override object	<a href="#">SyncRoot [312 ページ]</a>	SAPParameterCollection へのアクセスを同期するために使用するオブジェクトを取得します。
public new SAPParameter	<a href="#">this [313 ページ]</a>	指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

## 備考

SAPParameterCollection にはコンストラクタがありません。SAPParameterCollection オブジェクトは、SACommand オブジェクトの SACommand.Parameters プロパティから取得します。

このセクションの内容:

### [Add メソッド \[289 ページ\]](#)

SAPParameter オブジェクトをこのコレクションに追加します。

### [AddRange メソッド \[296 ページ\]](#)

SAPParameterCollection の末尾に値の配列を追加します。

### [AddWithValue\(string, object\) メソッド \[297 ページ\]](#)

このコレクションの末尾に値を追加します。

### [Clear\(\) メソッド \[298 ページ\]](#)

コレクションからすべての項目を削除します。

### [Contains メソッド \[298 ページ\]](#)

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

### [CopyTo\(Array, int\) メソッド \[300 ページ\]](#)

SAPParameter オブジェクトを SAPParameterCollection から指定された配列にコピーします。

### [GetEnumerator\(\) メソッド \[301 ページ\]](#)

SAPParameterCollection で反復処理する列挙子を返します。

### [GetParameter メソッド \[302 ページ\]](#)

SAPParameterCollection オブジェクトからのパラメータを返します。

### [IndexOf メソッド \[304 ページ\]](#)

コレクション内の SAPParameter オブジェクトのロケーションを返します。

### [Insert\(int, object\) メソッド \[306 ページ\]](#)

コレクション内の指定されたインデックス位置に SAPParameter オブジェクトを挿入します。

### [Remove\(object\) メソッド \[306 ページ\]](#)

指定された SAPParameter オブジェクトをコレクションから削除します。

### [RemoveAt メソッド \[307 ページ\]](#)



指定された SAPParameter オブジェクトをコレクションから削除します。

[SetParameter メソッド \[308 ページ\]](#)

SAPParameterCollection オブジェクトにパラメータを設定します。

[Count プロパティ \[310 ページ\]](#)

コレクション内の SAPParameter オブジェクトの数を返します。

[IsFixedSize プロパティ \[311 ページ\]](#)

SAPParameterCollection のサイズが固定かどうかを示す値を取得します。

[IsReadOnly プロパティ \[311 ページ\]](#)

SAPParameterCollection が読み込み専用かどうかを示す値を取得します。

[IsSynchronized プロパティ \[312 ページ\]](#)

SAPParameterCollection オブジェクトが同期しているかどうかを示す値を取得します。

[SyncRoot プロパティ \[312 ページ\]](#)

SAPParameterCollection へのアクセスを同期するために使用するオブジェクトを取得します。

[this プロパティ \[313 ページ\]](#)

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

## 関連情報

[SACommand クラス \[53 ページ\]](#)

[Parameters プロパティ \[89 ページ\]](#)

[SAPParameter クラス \[270 ページ\]](#)

### 1.22.1 Add メソッド

SAPParameter オブジェクトをこのコレクションに追加します。

#### オーバーロードリスト

変数とタイプ	オーバーロード名	説明
public SAPParameter	<a href="#">Add(SAPParameter) [291 ページ]</a>	SAPParameter オブジェクトをこのコレクションに追加します。
public override int	<a href="#">Add(object) [291 ページ]</a>	SAPParameter オブジェクトをこのコレクションに追加します。

変更子とタイプ	オーバーロード名	説明
public SAPParameter	<a href="#">Add(string, SADBType) [292 ページ]</a>	コレクションに対して指定されたパラメータ名とデータ型を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。
public SAPParameter	<a href="#">Add(string, SADBType, int) [293 ページ]</a>	コレクションに対して指定されたパラメータ名、データ型、長さを使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。
public SAPParameter	<a href="#">Add(string, SADBType, int, string) [294 ページ]</a>	コレクションに対して指定されたパラメータ名、データ型、長さ、ソースカラム名を使用して作成された SAPParameter オブジェクトをコレクションに追加します。
public SAPParameter	<a href="#">Add(string, object) [295 ページ]</a>	コレクションに対して指定されたパラメータ名と値を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

このセクションの内容:

[Add\(SAPParameter\) メソッド \[291 ページ\]](#)

SAPParameter オブジェクトをこのコレクションに追加します。

[Add\(object\) メソッド \[291 ページ\]](#)

SAPParameter オブジェクトをこのコレクションに追加します。

[Add\(string, SADBType\) メソッド \[292 ページ\]](#)

コレクションに対して指定されたパラメータ名とデータ型を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

[Add\(string, SADBType, int\) メソッド \[293 ページ\]](#)

コレクションに対して指定されたパラメータ名、データ型、長さを使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

[Add\(string, SADBType, int, string\) メソッド \[294 ページ\]](#)

コレクションに対して指定されたパラメータ名、データ型、長さ、ソースカラム名を使用して作成された SAPParameter オブジェクトをコレクションに追加します。

[Add\(string, object\) メソッド \[295 ページ\]](#)

コレクションに対して指定されたパラメータ名と値を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

## 1.22.1.1 Add(SAPparameter) メソッド

SAPparameter オブジェクトをこのコレクションに追加します。

### 構文

#### Visual Basic

```
Public Function Add (ByVal value As SAPparameter) As SAPparameter
```

#### C#

```
public SAPparameter Add (SAPparameter value)
```

### パラメータ

**value** コレクションに追加される SAPparameter オブジェクト。

### 戻り値

新しい SAPparameter オブジェクト。

## 1.22.1.2 Add(object) メソッド

SAPparameter オブジェクトをこのコレクションに追加します。

### 構文

#### Visual Basic

```
Public Overrides Function Add (ByVal value As Object) As Integer
```

#### C#

```
public override int Add (object value)
```

### パラメータ

**value** コレクションに追加される SAPparameter オブジェクト。

## 戻り値

新しい SAPParameter オブジェクトのインデックス。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

### 1.22.1.3 Add(string, SADBType) メソッド

コレクションに対して指定されたパラメータ名とデータ型を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (  
    ByVal parameterName As String,  
    ByVal saDbType As SADBType  
) As SAPParameter
```

##### C#

```
public SAPParameter Add (  
    string parameterName,  
    SADBType saDbType  
)
```

## パラメータ

**parameterName** パラメータの名前。

**saDbType** SADBType 値の 1 つ。

## 戻り値

新しい SAPParameter オブジェクト。

## 関連情報

[SADbType 列挙 \[348 ページ\]](#)

[Add\(SAPParameter\) メソッド \[291 ページ\]](#)

[Add\(string, object\) メソッド \[295 ページ\]](#)

### 1.22.1.4 Add(string, SADbType, int) メソッド

コレクションに対して指定されたパラメータ名、データ型、長さを使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (  
    ByVal parameterName As String,  
    ByVal saDbType As SADbType,  
    ByVal size As Integer  
) As SAPParameter
```

##### C#

```
public SAPParameter Add (  
    string parameterName,  
    SADbType saDbType,  
    int size  
)
```

## パラメータ

**parameterName** パラメータの名前。

**saDbType** SADbType 値の 1 つ。

**size** パラメータの長さ。

## 戻り値

新しい SAPParameter オブジェクト。

## 関連情報

[SADbType 列挙 \[348 ページ\]](#)

[Add\(SAPParameter\) メソッド \[291 ページ\]](#)

[Add\(string, object\) メソッド \[295 ページ\]](#)

### 1.22.1.5 Add(string, SADbType, int, string) メソッド

コレクションに対して指定されたパラメータ名、データ型、長さ、ソースカラム名を使用して作成された SAPParameter オブジェクトをコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (  
    ByVal parameterName As String,  
    ByVal saDbType As SADbType,  
    ByVal size As Integer,  
    ByVal sourceColumn As String  
) As SAPParameter
```

##### C#

```
public SAPParameter Add (  
    string parameterName,  
    SADbType saDbType,  
    int size,  
    string sourceColumn  
)
```

## パラメータ

**parameterName** パラメータの名前。

**saDbType** SADbType 値の 1 つ。

**size** カラムの長さ。

**sourceColumn** マッピング対象のソースカラムの名前。

## 戻り値

新しい SAPParameter オブジェクト。

## 関連情報

[SADbType 列挙 \[348 ページ\]](#)

[Add\(SAPParameter\) メソッド \[291 ページ\]](#)

[Add\(string, object\) メソッド \[295 ページ\]](#)

### 1.22.1.6 Add(string, object) メソッド

コレクションに対して指定されたパラメータ名と値を使用して作成された SAPParameter オブジェクトをこのコレクションに追加します。

#### 構文

##### Visual Basic

```
Public Function Add (  
    ByVal parameterName As String,  
    ByVal value As Object  
) As SAPParameter
```

##### C#

```
public SAPParameter Add (  
    string parameterName,  
    object value  
)
```

## パラメータ

**parameterName** パラメータの名前。

**value** コレクションに追加するパラメータの値。

## 戻り値

新しい SAPParameter オブジェクト。

## 備考

定数 0 と 0.0 の特別な処理と、オーバーロードされたメソッドの解決方法のため、このメソッドを使用するときは、定数値を必要なオブジェクトタイプに明示的にキャストします。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

### 1.22.2 AddRange メソッド

SAPParameterCollection の末尾に値の配列を追加します。

#### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override void	<a href="#">AddRange(Array) [296 ページ]</a>	SAPParameterCollection の末尾に値の配列を追加します。
public void	<a href="#">AddRange(SAPParameter[]) [297 ページ]</a>	SAPParameterCollection の末尾に値の配列を追加します。

このセクションの内容:

[AddRange\(Array\) メソッド \[296 ページ\]](#)

SAPParameterCollection の末尾に値の配列を追加します。

[AddRange\(SAPParameter\[\]\) メソッド \[297 ページ\]](#)

SAPParameterCollection の末尾に値の配列を追加します。

#### 1.22.2.1 AddRange(Array) メソッド

SAPParameterCollection の末尾に値の配列を追加します。

##### 構文

###### Visual Basic

```
Public Overrides Sub AddRange (ByVal values As Array)
```

###### C#

```
public override void AddRange (Array values)
```



## パラメータ

**values** 追加する値。

### 1.22.2.2 AddRange(SAPparameter[]) メソッド

SAPparameterCollection の末尾に値の配列を追加します。

#### 構文

##### Visual Basic

```
Public Sub AddRange (ByVal values As SAPparameter())
```

##### C#

```
public void AddRange (SAPparameter[] values)
```

## パラメータ

**values** このコレクションの末尾に追加する SAPparameter オブジェクトの配列。

### 1.22.3 AddWithValue(string, object) メソッド

このコレクションの末尾に値を追加します。

#### 構文

##### Visual Basic

```
Public Function AddWithValue (  
    ByVal parameterName As String,  
    ByVal value As Object  
) As SAPparameter
```

##### C#

```
public SAPparameter AddWithValue (  
    string parameterName,  
    object value  
)
```

## パラメータ

**parameterName** パラメータの名前。

**value** 追加される値。

## 戻り値

新しい SAPParameter オブジェクト。

### 1.22.4 Clear() メソッド

コレクションからすべての項目を削除します。

#### 構文

##### Visual Basic

```
Public Overrides Sub Clear ()
```

##### C#

```
public override void Clear ()
```

### 1.22.5 Contains メソッド

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

## オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override bool	<a href="#">Contains(object) [299 ページ]</a>	コレクション内に SAPParameter オブジェクトがあるかどうかを示します。
public override bool	<a href="#">Contains(string) [300 ページ]</a>	コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

このセクションの内容:

[Contains\(object\) メソッド \[299 ページ\]](#)

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

[Contains\(string\) メソッド \[300 ページ\]](#)

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

## 1.22.5.1 Contains(object) メソッド

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

### 構文

#### Visual Basic

```
Public Overrides Function Contains (ByVal value As Object) As Boolean
```

#### C#

```
public override bool Contains (object value)
```

### パラメータ

**value** 検索する SAPParameter オブジェクト。

### 戻り値

コレクションに、その SAPParameter オブジェクトがある場合は true、ない場合は false。

### 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[Contains\(string\) メソッド \[300 ページ\]](#)

## 1.22.5.2 Contains(string) メソッド

コレクション内に SAPParameter オブジェクトがあるかどうかを示します。

### 構文

#### Visual Basic

```
Public Overrides Function Contains (ByVal value As String) As Boolean
```

#### C#

```
public override bool Contains (string value)
```

### パラメータ

**value** 検索対象のパラメータの名前。

### 戻り値

コレクションに、その SAPParameter オブジェクトがある場合は true、ない場合は false。

### 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[Contains\(object\) メソッド \[299 ページ\]](#)

## 1.22.6 CopyTo(Array, int) メソッド

SAPParameter オブジェクトを SAPParameterCollection から指定された配列にコピーします。

### 構文

#### Visual Basic

```
Public Overrides Sub CopyTo (  
    ByVal array As Array,  
    ByVal index As Integer  
)
```

## C#

```
public override void CopyTo (  
    Array array,  
    int index  
)
```

## パラメータ

**array** SAPParameter オブジェクトのコピー先の配列。

**index** 配列の開始インデックス。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.7 GetEnumerator() メソッド

SAPParameterCollection で反復処理する列挙子を返します。

### 構文

#### Visual Basic

```
Public Overrides Function GetEnumerator () As System.Collections.IEnumerator
```

#### C#

```
public override IEnumerator GetEnumerator ()
```

## 戻り値

SAPParameterCollection オブジェクトの System.Collections.IEnumerator。

## 関連情報

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.8 GetParameter メソッド

SAPParameterCollection オブジェクトからのパラメータを返します。

### オーバロードリスト

変更子とタイプ	オーバーロード名	説明
protected override DbParameter	<a href="#">GetParameter(int) [302 ページ]</a>	SAPParameterCollection オブジェクトからのパラメータを返します。
protected override DbParameter	<a href="#">GetParameter(string) [303 ページ]</a>	SAPParameterCollection オブジェクトからのパラメータを返します。

このセクションの内容:

#### [GetParameter\(int\) メソッド \[302 ページ\]](#)

SAPParameterCollection オブジェクトからのパラメータを返します。

#### [GetParameter\(string\) メソッド \[303 ページ\]](#)

SAPParameterCollection オブジェクトからのパラメータを返します。

### 1.22.8.1 GetParameter(int) メソッド

SAPParameterCollection オブジェクトからのパラメータを返します。

#### 構文

##### Visual Basic

```
Protected Overrides Function GetParameter (ByVal index As Integer) As DbParameter
```

##### C#

```
protected override DbParameter GetParameter (int index)
```

### パラメータ

**index** コレクション内のパラメータの 0 から始まるインデックス。

## 戻り値

SAPParameterCollection オブジェクトからの System.Data.Common.DbParameter。

## 関連情報

[SAPParameterCollection クラス \[286 ページ\]](#)

### 1.22.8.2 GetParameter(string) メソッド

SAPParameterCollection オブジェクトからのパラメータを返します。

#### 構文

##### Visual Basic

```
Protected Overrides Function GetParameter (ByVal parameterName As String) As DbParameter
```

##### C#

```
protected override DbParameter GetParameter (string parameterName)
```

## パラメータ

**parameterName** 検索するパラメータの名前。

## 戻り値

SAPParameterCollection オブジェクトからの System.Data.Common.DbParameter。

## 関連情報

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.9 IndexOf メソッド

コレクション内の SAParameter オブジェクトのロケーションを返します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override int	<a href="#">IndexOf(object) [304 ページ]</a>	コレクション内の SAParameter オブジェクトのロケーションを返します。
public override int	<a href="#">IndexOf(string) [305 ページ]</a>	コレクション内の SAParameter オブジェクトのロケーションを返します。

このセクションの内容:

#### [IndexOf\(object\) メソッド \[304 ページ\]](#)

コレクション内の SAParameter オブジェクトのロケーションを返します。

#### [IndexOf\(string\) メソッド \[305 ページ\]](#)

コレクション内の SAParameter オブジェクトのロケーションを返します。

### 1.22.9.1 IndexOf(object) メソッド

コレクション内の SAParameter オブジェクトのロケーションを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function IndexOf (ByVal value As Object) As Integer
```

##### C#

```
public override int IndexOf (object value)
```

### パラメータ

**value** 検索する SAParameter オブジェクト。



## 戻り値

コレクション内の SAPParameter オブジェクトの 0 から始まるロケーション。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[IndexOf\(string\) メソッド \[305 ページ\]](#)

### 1.22.9.2 IndexOf(string) メソッド

コレクション内の SAPParameter オブジェクトのロケーションを返します。

#### 構文

##### Visual Basic

```
Public Overrides Function IndexOf (ByVal parameterName As String) As Integer
```

##### C#

```
public override int IndexOf (string parameterName)
```

## パラメータ

**parameterName** 検索するパラメータの名前。

## 戻り値

コレクション内の SAPParameter オブジェクトの 0 から始まるインデックス。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[IndexOf\(object\) メソッド \[304 ページ\]](#)

## 1.22.10 Insert(int, object) メソッド

コレクション内の指定されたインデックス位置に SAPParameter オブジェクトを挿入します。

### 構文

#### Visual Basic

```
Public Overrides Sub Insert (  
    ByVal index As Integer,  
    ByVal value As Object  
)
```

#### C#

```
public override void Insert (  
    int index,  
    object value  
)
```

### パラメータ

**index** パラメータの挿入先となる、コレクション内の 0 から始まるインデックス。

**value** コレクションに追加される SAPParameter オブジェクト。

## 1.22.11 Remove(object) メソッド

指定された SAPParameter オブジェクトをコレクションから削除します。

### 構文

#### Visual Basic

```
Public Overrides Sub Remove (ByVal value As Object)
```

#### C#

```
public override void Remove (object value)
```

### パラメータ

**value** コレクションから削除する SAPParameter オブジェクト。

## 1.22.12 RemoveAt メソッド

指定された SAParameter オブジェクトをコレクションから削除します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override void	<a href="#">RemoveAt(int) [307 ページ]</a>	指定された SAParameter オブジェクトをコレクションから削除します。
public override void	<a href="#">RemoveAt(string) [308 ページ]</a>	指定された SAParameter オブジェクトをコレクションから削除します。

このセクションの内容:

#### [RemoveAt\(int\) メソッド \[307 ページ\]](#)

指定された SAParameter オブジェクトをコレクションから削除します。

#### [RemoveAt\(string\) メソッド \[308 ページ\]](#)

指定された SAParameter オブジェクトをコレクションから削除します。

### 1.22.12.1 RemoveAt(int) メソッド

指定された SAParameter オブジェクトをコレクションから削除します。

#### 構文

##### Visual Basic

```
Public Overrides Sub RemoveAt (ByVal index As Integer)
```

##### C#

```
public override void RemoveAt (int index)
```

### パラメータ

**index** 削除するパラメータの 0 から始まるインデックス。

## 関連情報

[RemoveAt\(string\) メソッド \[308 ページ\]](#)

### 1.22.12.2 RemoveAt(string) メソッド

指定された SAPParameter オブジェクトをコレクションから削除します。

#### 構文

##### Visual Basic

```
Public Overrides Sub RemoveAt (ByVal parameterName As String)
```

##### C#

```
public override void RemoveAt (string parameterName)
```

## パラメータ

**parameterName** 削除する SAPParameter オブジェクトの名前。

## 関連情報

[RemoveAt\(int\) メソッド \[307 ページ\]](#)

### 1.22.13 SetParameter メソッド

SAPParameterCollection オブジェクトにパラメータを設定します。

## オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
protected override void	<a href="#">SetParameter(int, DbParameter) [309 ページ]</a>	SAPParameterCollection オブジェクトにパラメータを設定します。

変数とタイプ	オーバーロード名	説明
protected override void	<a href="#">SetParameter(string, DbParameter) [310 ページ]</a>	SAPParameterCollection オブジェクトにパラメータを設定します。

このセクションの内容:

[SetParameter\(int, DbParameter\) メソッド \[309 ページ\]](#)

SAPParameterCollection オブジェクトにパラメータを設定します。

[SetParameter\(string, DbParameter\) メソッド \[310 ページ\]](#)

SAPParameterCollection オブジェクトにパラメータを設定します。

## 1.22.13.1 SetParameter(int, DbParameter) メソッド

SAPParameterCollection オブジェクトにパラメータを設定します。

### 構文

#### Visual Basic

```
Protected Overrides Sub SetParameter (
    ByVal index As Integer,
    ByVal value As DbParameter
)
```

#### C#

```
protected override void SetParameter (
    int index,
    DbParameter value
)
```

### パラメータ

**index** 設定するパラメータの 0 から始まるインデックス。

**value** SAPParameterCollection オブジェクトに挿入される System.Data.Common.DbParameter。

### 関連情報

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.13.2 SetParameter(string, DbParameter) メソッド

SAParameterCollection オブジェクトにパラメータを設定します。

### 構文

#### Visual Basic

```
Protected Overrides Sub SetParameter (  
    ByVal parameterName As String,  
    ByVal value As DbParameter  
)
```

#### C#

```
protected override void SetParameter (  
    string parameterName,  
    DbParameter value  
)
```

### パラメータ

**parameterName** 設定するパラメータの名前。

**value** SAParameterCollection オブジェクトに挿入される System.Data.Common.DbParameter。

### 関連情報

[SAParameterCollection クラス \[286 ページ\]](#)

## 1.22.14 Count プロパティ

コレクション内の SAParameter オブジェクトの数を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property Count As Integer
```

#### C#

```
public override int Count {get;}
```

## 備考

コレクション内の SAPParameter オブジェクトの数。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.15 IsFixedSize プロパティ

SAPParameterCollection のサイズが固定かどうかを示す値を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property IsFixedSize As Boolean
```

#### C#

```
public override bool IsFixedSize {get;}
```

## 備考

コレクションのサイズが固定の場合は true、そうでない場合は false。

## 1.22.16 IsReadOnly プロパティ

SAPParameterCollection が読み込み専用かどうかを示す値を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property IsReadOnly As Boolean
```

#### C#

```
public override bool IsReadOnly {get;}
```

## 備考

コレクションが読み込み専用の場合は true、そうでない場合は false。

### 1.22.17 IsSynchronized プロパティ

SAPParameterCollection オブジェクトが同期しているかどうかを示す値を取得します。

#### 構文

##### Visual Basic

```
Public ReadOnly Overrides Property IsSynchronized As Boolean
```

##### C#

```
public override bool IsSynchronized {get;}
```

## 備考

コレクションが同期している場合は true、そうでない場合は false。

### 1.22.18 SyncRoot プロパティ

SAPParameterCollection へのアクセスを同期するために使用するオブジェクトを取得します。

#### 構文

##### Visual Basic

```
Public ReadOnly Overrides Property SyncRoot As Object
```

##### C#

```
public override object SyncRoot {get;}
```



## 1.22.19 this プロパティ

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public new SAPParameter	<a href="#">this[int index] [313 ページ]</a>	指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。
public new SAPParameter	<a href="#">this[string parameterName] [314 ページ]</a>	指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

このセクションの内容:

#### [this\[int index\] プロパティ \[313 ページ\]](#)

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

#### [this\[string parameterName\] プロパティ \[314 ページ\]](#)

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

### 1.22.19.1 this[int index] プロパティ

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

#### 構文

##### Visual Basic

```
Public Shadows Property Item (ByVal index As Integer) As SAPParameter
```

##### C#

```
public new SAPParameter this[int index] {get;set;}
```

### 戻り値

指定されたインデックス位置の SAPParameter。

## 備考

SAPParameter オブジェクト。

C# では、このプロパティは SAPParameterCollection オブジェクトのインデクサです。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[SAPParameterCollection クラス \[286 ページ\]](#)

## 1.22.19.2 this[string parameterName] プロパティ

指定されたインデックス位置の SAPParameter オブジェクトを取得または設定します。

### 構文

#### Visual Basic

```
Public Shadows Property Item (ByVal parameterName As String) As SAPParameter
```

#### C#

```
public new SAPParameter this[string parameterName] {get;set;}
```

## 戻り値

指定された名前の SAPParameter オブジェクト。

## 備考

SAPParameter オブジェクト。

C# では、このプロパティは SAPParameterCollection オブジェクトのインデクサです。

## 関連情報

[SAPParameter クラス \[270 ページ\]](#)

[SAPParameterCollection クラス \[286 ページ\]](#)

[GetOrdinal\(string\) メソッド \[227 ページ\]](#)

[GetValue\(int\) メソッド \[235 ページ\]](#)

[GetFieldType\(int\) メソッド \[222 ページ\]](#)

## 1.23 SAPermission クラス

ユーザがデータソースへのアクセスに適したセキュリティレベルを持っていることを .NET データプロバイダが確認できるようにします。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAPermission Inherits  
System.Data.Common.DBDataPermission
```

#### C#

```
public sealed class SAPermission : System.Data.Common.DBDataPermission
```

### メンバー

SAPermission のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SAPermission(PermissionState) [316 ページ]</a>	SAPermission クラスの新しいインスタンスを初期化します。

#### メソッド

変更子とタイプ	メソッド	説明
protected override DBDataPermission	<a href="#">CreateInstance() [316 ページ]</a>	SAPermission クラスの新しいインスタンスを作成します。

このセクションの内容:

[SAPermission\(PermissionState\) コンストラクタ \[316 ページ\]](#)

SAPermission クラスの新しいインスタンスを初期化します。

[CreateInstance\(\) メソッド \[316 ページ\]](#)

SAPermission クラスの新しいインスタンスを作成します。

## 1.23.1 SAPermission(PermissionState) コンストラクタ

SAPermission クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Public Sub SAPermission (ByVal state As PermissionState)
```

#### C#

```
public SAPermission (PermissionState state)
```

### パラメータ

**state** PermissionState 値の 1 つ。

## 1.23.2 CreateInstance() メソッド

SAPermission クラスの新しいインスタンスを作成します。

### 構文

#### Visual Basic

```
Protected Overrides Function CreateInstance () As DBDataPermission
```

#### C#

```
protected override DBDataPermission CreateInstance ()
```

### 戻り値

新しい SAPermission オブジェクト。

## 1.24 SAPermissionAttribute クラス

カスタムセキュリティ属性にセキュリティアクションを関連付けます。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAPermissionAttribute Inherits  
System.Data.Common.DBDataPermissionAttribute
```

#### C#

```
public sealed class SAPermissionAttribute :  
System.Data.Common.DBDataPermissionAttribute
```

### メンバー

SAPermissionAttribute のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SAPermissionAttribute(SecurityAction)</a> [318 ページ]	SAPermissionAttribute クラスの新しいインスタンスを初期化します。

#### メソッド

変更子とタイプ	メソッド	説明
public override IPermission	<a href="#">CreatePermission()</a> [318 ページ]	属性プロパティに応じて設定された SAPermission オブジェクトを返します。

このセクションの内容:

#### [SAPermissionAttribute\(SecurityAction\) コンストラクタ](#) [318 ページ]

SAPermissionAttribute クラスの新しいインスタンスを初期化します。

#### [CreatePermission\(\) メソッド](#) [318 ページ]

属性プロパティに応じて設定された SAPermission オブジェクトを返します。

## 1.24.1 SAPermissionAttribute(SecurityAction) コンストラクタ

SAPermissionAttribute クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Public Sub SAPermissionAttribute (ByVal action As SecurityAction)
```

#### C#

```
public SAPermissionAttribute (SecurityAction action)
```

### パラメータ

**action** 宣言型セキュリティを使用して実行できるアクションを表す SecurityAction 値の 1 つ。

### 戻り値

An SAPermissionAttribute オブジェクト。

## 1.24.2 CreatePermission() メソッド

属性プロパティに応じて設定された SAPermission オブジェクトを返します。

### 構文

#### Visual Basic

```
Public Overrides Function CreatePermission () As IPermission
```

#### C#

```
public override IPermission CreatePermission ()
```

## 1.25 SARowsCopiedEventArgs クラス

SARowsCopiedEventHandler に渡される引数のセットを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SARowsCopiedEventArgs
```

#### C#

```
public sealed class SARowsCopiedEventArgs
```

### メンバー

SARowsCopiedEventArgs のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### コンストラクタ

変数とタイプ	コンストラクタ	説明
public	<a href="#">SARowsCopiedEventArgs(long) [320 ページ]</a>	SARowsCopiedEventArgs オブジェクトの新しいインスタンスを作成します。

#### プロパティ

変数とタイプ	プロパティ	説明
public bool	<a href="#">Abort [320 ページ]</a>	バルクコピーオペレーションをアボートするかどうかを示す値を取得または設定します。
public long	<a href="#">RowsCopied [320 ページ]</a>	現在のバルクコピーオペレーションでコピーされるローの数を取得します。

このセクションの内容:

#### [SARowsCopiedEventArgs\(long\) コンストラクタ \[320 ページ\]](#)

SARowsCopiedEventArgs オブジェクトの新しいインスタンスを作成します。

#### [Abort プロパティ \[320 ページ\]](#)

バルクコピーオペレーションをアボートするかどうかを示す値を取得または設定します。

#### [RowsCopied プロパティ \[320 ページ\]](#)

現在のバルクコピーオペレーションでコピーされるローの数を取得します。

## 1.25.1 SARowsCopiedEventArgs(long) コンストラクタ

SARowsCopiedEventArgs オブジェクトの新しいインスタンスを作成します。

### 構文

#### Visual Basic

```
Public Sub SARowsCopiedEventArgs (ByVal rowsCopied As Long)
```

#### C#

```
public SARowsCopiedEventArgs (long rowsCopied)
```

### パラメータ

**rowsCopied** 現在のバルクコピーオペレーションでコピーされるローの数を示す 64 ビット整数値。

## 1.25.2 Abort プロパティ

バルクコピーオペレーションをアボートするかどうかを示す値を取得または設定します。

### 構文

#### Visual Basic

```
Public Property Abort As Boolean
```

#### C#

```
public bool Abort {get;set;}
```

## 1.25.3 RowsCopied プロパティ

現在のバルクコピーオペレーションでコピーされるローの数を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Property RowsCopied As Long
```

#### C#

```
public long RowsCopied {get;}
```



## 1.26 SARowUpdatedEventArgs クラス

RowUpdated イベントのデータを提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SARowUpdatedEventArgs Inherits  
System.Data.Common.RowUpdatedEventArgs
```

#### C#

```
public sealed class SARowUpdatedEventArgs :  
System.Data.Common.RowUpdatedEventArgs
```

### メンバー

SARowUpdatedEventArgs のすべてのメンバー (継承されたメンバーも含まれます) を以下に示します。

#### コンストラクタ

変数とタイプ	コンストラクタ	説明
public	<a href="#">SARowUpdatedEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) [322 ページ]</a>	SARowUpdatedEventArgs クラスの新しいインスタンスを初期化します。

#### プロパティ

変数とタイプ	プロパティ	説明
public new SACommand	<a href="#">Command [322 ページ]</a>	DataAdapter.Update の呼び出し時に実行する SACommand を取得します。
public new int	<a href="#">RecordsAffected [323 ページ]</a>	SQL 文の実行によって変更、挿入、または削除されたローの数を返します。

このセクションの内容:

[SARowUpdatedEventArgs\(DataRow, IDbCommand, StatementType, DataTableMapping\) コンストラクタ \[322 ページ\]](#)

SARowUpdatedEventArgs クラスの新しいインスタンスを初期化します。

[Command プロパティ \[322 ページ\]](#)

DataAdapter.Update の呼び出し時に実行する SACommand を取得します。

[RecordsAffected プロパティ \[323 ページ\]](#)

SQL 文の実行によって変更、挿入、または削除されたローの数を返します。

## 1.26.1 SARowUpdatedEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) コンストラクタ

SARowUpdatedEventArgs クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Public Sub SARowUpdatedEventArgs (  
    ByVal row As DataRow,  
    ByVal command As IDbCommand,  
    ByVal statementType As StatementType,  
    ByVal tableMapping As DataTableMapping  
)
```

#### C#

```
public SARowUpdatedEventArgs (  
    DataRow row,  
    IDbCommand command,  
    StatementType statementType,  
    DataTableMapping tableMapping  
)
```

### パラメータ

**row** Update によって送信された DataRow。

**command** Update が呼び出されたときに実行された IDbCommand。

**statementType** 実行するクエリの種類を指定する StatementType 値のいずれか。

**tableMapping** Update を介して送信された DataTableMapping。

## 1.26.2 Command プロパティ

DataAdapter.Update の呼び出し時に実行する SACommand を取得します。

### 構文

#### Visual Basic

```
Public ReadOnly Shadows Property Command As SACommand
```

#### C#

```
public new SACommand Command {get;}
```

## 1.26.3 RecordsAffected プロパティ

SQL 文の実行によって変更、挿入、または削除されたローの数を返します。

### 構文

#### Visual Basic

```
Public ReadOnly Shadows Property RecordsAffected As Integer
```

#### C#

```
public new int RecordsAffected {get;}
```

### 備考

変更、挿入、または削除されたローの数。文が失敗したときにローが影響されなかった場合は 0、SELECT 文の場合は -1。

## 1.27 SRowUpdatingEventArgs クラス

RowUpdating イベントのデータを提供します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SRowUpdatingEventArgs Inherits  
System.Data.Common.RowUpdatingEventArgs
```

#### C#

```
public sealed class SRowUpdatingEventArgs :  
System.Data.Common.RowUpdatingEventArgs
```

### メンバー

SRowUpdatingEventArgs のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SARowUpdatingEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) [324 ページ]</a>	SARowUpdatingEventArgs クラスの新しいインスタンスを初期化します。

#### プロパティ

変更子とタイプ	プロパティ	説明
public new SACommand	<a href="#">Command [325 ページ]</a>	Update の実行時に実行する SACommand を指定します。

このセクションの内容:

[SARowUpdatingEventArgs\(DataRow, IDbCommand, StatementType, DataTableMapping\) コンストラクタ \[324 ページ\]](#)

SARowUpdatingEventArgs クラスの新しいインスタンスを初期化します。

[Command プロパティ \[325 ページ\]](#)

Update の実行時に実行する SACommand を指定します。

## 1.27.1 SARowUpdatingEventArgs(DataRow, IDbCommand, StatementType, DataTableMapping) コンストラクタ

SARowUpdatingEventArgs クラスの新しいインスタンスを初期化します。

### 構文

#### Visual Basic

```
Public Sub SARowUpdatingEventArgs (
    ByVal row As DataRow,
    ByVal command As IDbCommand,
    ByVal statementType As StatementType,
    ByVal tableMapping As DataTableMapping
)
```

#### C#

```
public SARowUpdatingEventArgs (
    DataRow row,
    IDbCommand command,
    StatementType statementType,
    DataTableMapping tableMapping
)
```

## パラメータ

**row** 更新する DataRow。

**command** 更新時に実行する IDbCommand。

**statementType** 実行するクエリの種類を指定する StatementType 値のいずれか。

**tableMapping** Update を介して送信された DataTableMapping。

## 1.27.2 Command プロパティ

Update の実行時に実行する SACCommand を指定します。

### 構文

#### Visual Basic

```
Public Shadows Property Command As SACCommand
```

#### C#

```
public new SACCommand Command {get;set;}
```

## 1.28 SAServerSideConnection クラス

データベースへのサーバ側の接続を表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SAServerSideConnection
```

#### C#

```
public sealed class SAServerSideConnection
```

## メンバー

SAServerSideConnection のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### プロパティ

変更子とタイプ	プロパティ	説明
public SAConnection	<a href="#">Connection [326 ページ]</a>	データベースサーバで SQL を実行するために必要な SAConnection オブジェクトを提供します。

## 備考

このクラスは CLR 外部環境からのサーバ側 SQL の実行をサポートします。データベースサーバから呼び出した CLR 関数またはメソッドは、データベースサーバにコールバックできます。データベースサーバとの接続はすでに確立されています。接続を確立するために SAConnection を使用する必要はありません。

このセクションの内容:

### [Connection プロパティ \[326 ページ\]](#)

データベースサーバで SQL を実行するために必要な SAConnection オブジェクトを提供します。

## 1.28.1 Connection プロパティ

データベースサーバで SQL を実行するために必要な SAConnection オブジェクトを提供します。

### 構文

#### Visual Basic

```
Public Shared ReadOnly Property Connection As SAConnection
```

#### C#

```
public SAConnection Connection {get;}
```

## 備考

データベースサーバとの接続はすでに確立されています。接続を確立するために SAConnection を使用する必要はありません。次のような文を使用して、接続オブジェクトを取得します。

```
SAConnection _conn = SASServerSideConnection.Connection;
```

### 例

次の例は、SAConnection オブジェクトを取得し、SQL 文を実行してテーブルを作成します。

```
using Sap.Data.SQLAnywhere;
using Sap.SQLAnywhere.Server;
SAConnection _conn = SASServerSideConnection.Connection;
```

```
SACCommand cmd = _conn.CreateCommand();
cmd.CommandText = "CREATE TABLE Tab( c1 int, c2 char(128), c3 smallint, c4
double, c5 numeric(30,6) )";
cmd.ExecuteNonQuery();
cmd.Dispose();
```

## 関連情報

[SACConnection クラス \[118 ページ\]](#)

## 1.29 SATcpOptionsBuilder クラス

SACConnection オブジェクトが使用する接続文字列の TCP オプション部分を作成および管理する簡単な方法を提供します。

### 📄 構文

#### Visual Basic

```
Public NotInheritable Class SATcpOptionsBuilder Inherits
SACConnectionStringBuilderBase
```

#### C#

```
public sealed class SATcpOptionsBuilder : SACConnectionStringBuilderBase
```

## メンバー

SATcpOptionsBuilder のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

### コンストラクタ

変更子とタイプ	コンストラクタ	説明
public	<a href="#">SATcpOptionsBuilder [330 ページ]</a>	SATcpOptionsBuilder オブジェクトを初期化します。

### メソッド

変更子とタイプ	メソッド	説明
public override string	<a href="#">ToString() [332 ページ]</a>	SATcpOptionsBuilder オブジェクトを文字列表現に変換します。

### プロパティ

変更子とタイプ	プロパティ	説明
public string	<a href="#">Broadcast [332 ページ]</a>	Broadcast オプションを取得または設定します。
public string	<a href="#">BroadcastListener [332 ページ]</a>	BroadcastListener オプションを取得または設定します。
public string	<a href="#">ClientPort [333 ページ]</a>	ClientPort オプションを取得または設定します。
public string	<a href="#">DoBroadcast [333 ページ]</a>	DoBroadcast オプションを取得または設定します。
public string	<a href="#">Host [333 ページ]</a>	Host オプションを取得または設定します。
public string	<a href="#">IPV6 [334 ページ]</a>	IPV6 オプションを取得または設定します。
public string	<a href="#">LDAP [334 ページ]</a>	LDAP オプションを取得または設定します。
public string	<a href="#">LocalOnly [334 ページ]</a>	LocalOnly オプションを取得または設定します。
public string	<a href="#">MyIP [335 ページ]</a>	MyIP オプションを取得または設定します。
public int	<a href="#">ReceiveBufferSize [335 ページ]</a>	ReceiveBufferSize オプションを取得または設定します。
public int	<a href="#">SendBufferSize [335 ページ]</a>	Send BufferSize オプションを取得または設定します。
public string	<a href="#">ServerPort [336 ページ]</a>	ServerPort オプションを取得または設定します。
public string	<a href="#">TDS [336 ページ]</a>	TDS オプションを取得または設定します。
public int	<a href="#">Timeout [336 ページ]</a>	Timeout オプションを取得または設定します。
public string	<a href="#">VerifyServerName [337 ページ]</a>	VerifyServerName オプションを取得または設定します。

#### SAConnectionStringBuilderBase から継承されたメンバー

変更子とタイプ	メンバー	説明
public override bool	<a href="#">ContainsKey(string) [173 ページ]</a>	SAConnectionStringBuilder オブジェクトに特定のキーワードが含まれているかどうかを判断します。
public string	<a href="#">GetKeyword(string) [174 ページ]</a>	指定された SAConnectionStringBuilder プロパティのキーワードを取得します。
public bool	<a href="#">GetUseLongNameAsKeyword() [174 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかを示す boolean 値を取得します。
public override ICollection	<a href="#">Keys [178 ページ]</a>	SAConnectionStringBuilder のキーが含まれた System.Collections.ICollection を取得します。



変数とタイプ	メンバー	説明
public override bool	<a href="#">Remove(string) [175 ページ]</a>	指定されたキーが設定されたエントリを SAConnectionStringBuilder インスタンスから削除します。
public void	<a href="#">SetUseLongNameAsKeyword(bool) [176 ページ]</a>	長い接続パラメータ名を接続文字列で使用するかどうかを示す boolean 値を設定します。
public override bool	<a href="#">ShouldSerialize(string) [176 ページ]</a>	指定されたキーが、この SAConnectionStringBuilder インスタンスに存在するかどうかを示します。
public override object	<a href="#">this[string keyword] [178 ページ]</a>	接続キーワードの値を取得または設定します。
public override bool	<a href="#">TryGetValue(string, out object) [177 ページ]</a>	入力されたキーに対応する値を、この SAConnectionStringBuilder から取り出します。

このセクションの内容:

[SATcpOptionsBuilder コンストラクタ \[330 ページ\]](#)

SATcpOptionsBuilder オブジェクトを初期化します。

[ToString\(\) メソッド \[332 ページ\]](#)

SATcpOptionsBuilder オブジェクトを文字列表現に変換します。

[Broadcast プロパティ \[332 ページ\]](#)

Broadcast オプションを取得または設定します。

[BroadcastListener プロパティ \[332 ページ\]](#)

BroadcastListener オプションを取得または設定します。

[ClientPort プロパティ \[333 ページ\]](#)

ClientPort オプションを取得または設定します。

[DoBroadcast プロパティ \[333 ページ\]](#)

DoBroadcast オプションを取得または設定します。

[Host プロパティ \[333 ページ\]](#)

Host オプションを取得または設定します。

[IPV6 プロパティ \[334 ページ\]](#)

IPV6 オプションを取得または設定します。

[LDAP プロパティ \[334 ページ\]](#)

LDAP オプションを取得または設定します。

[LocalOnly プロパティ \[334 ページ\]](#)

LocalOnly オプションを取得または設定します。

[MyIP プロパティ \[335 ページ\]](#)

MyIP オプションを取得または設定します。

[ReceiveBufferSize プロパティ \[335 ページ\]](#)

ReceiveBufferSize オプションを取得または設定します。

### [SendBufferSize プロパティ \[335 ページ\]](#)

Send BufferSize オプションを取得または設定します。

### [ServerPort プロパティ \[336 ページ\]](#)

ServerPort オプションを取得または設定します。

### [TDS プロパティ \[336 ページ\]](#)

TDS オプションを取得または設定します。

### [Timeout プロパティ \[336 ページ\]](#)

Timeout オプションを取得または設定します。

### [VerifyServerName プロパティ \[337 ページ\]](#)

VerifyServerName オプションを取得または設定します。

## 関連情報

[SAConnection クラス \[118 ページ\]](#)

## 1.29.1 SATcpOptionsBuilder コンストラクタ

SATcpOptionsBuilder オブジェクトを初期化します。

## オーバードリスト

変更子とタイプ	オーバード名	説明
public	<a href="#">SATcpOptionsBuilder() [331 ページ]</a>	SATcpOptionsBuilder オブジェクトを初期化します。
public	<a href="#">SATcpOptionsBuilder(string) [331 ページ]</a>	SATcpOptionsBuilder オブジェクトを初期化します。

このセクションの内容:

### [SATcpOptionsBuilder\(\) コンストラクタ \[331 ページ\]](#)

SATcpOptionsBuilder オブジェクトを初期化します。

### [SATcpOptionsBuilder\(string\) コンストラクタ \[331 ページ\]](#)

SATcpOptionsBuilder オブジェクトを初期化します。

## 1.29.1.1 SATcpOptionsBuilder() コンストラクタ

SATcpOptionsBuilder オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SATcpOptionsBuilder ()
```

#### C#

```
public SATcpOptionsBuilder ()
```

### 例

次の文は、SATcpOptionsBuilder オブジェクトを初期化します。

```
SATcpOptionsBuilder options = new SATcpOptionsBuilder ( );
```

## 1.29.1.2 SATcpOptionsBuilder(string) コンストラクタ

SATcpOptionsBuilder オブジェクトを初期化します。

### 構文

#### Visual Basic

```
Public Sub SATcpOptionsBuilder (ByVal options As String)
```

#### C#

```
public SATcpOptionsBuilder (string options)
```

### パラメータ

**options** データベースサーバ TCP 接続パラメータオプション文字列。接続パラメータのリストについては、「接続パラメータ」を参照してください。

### 例

次の文は、SATcpOptionsBuilder オブジェクトを初期化します。

```
SATcpOptionsBuilder options = new SATcpOptionsBuilder ("localonly=yes;port=6873");
```

## 1.29.2 ToString() メソッド

SATcpOptionsBuilder オブジェクトを文字列表現に変換します。

### 構文

#### Visual Basic

```
Public Overrides Function ToString () As String
```

#### C#

```
public override string ToString ()
```

### 戻り値

構築されるオプション文字列。

## 1.29.3 Broadcast プロパティ

Broadcast オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Broadcast As String
```

#### C#

```
public string Broadcast {get;set;}
```

## 1.29.4 BroadcastListener プロパティ

BroadcastListener オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property BroadcastListener As String
```

**C#**

```
public string BroadcastListener {get;set;}
```

## 1.29.5 ClientPort プロパティ

ClientPort オプションを取得または設定します。

 構文

**Visual Basic**

```
Public Property ClientPort As String
```

**C#**

```
public string ClientPort {get;set;}
```

## 1.29.6 DoBroadcast プロパティ

DoBroadcast オプションを取得または設定します。

 構文

**Visual Basic**

```
Public Property DoBroadcast As String
```

**C#**

```
public string DoBroadcast {get;set;}
```

## 1.29.7 Host プロパティ

Host オプションを取得または設定します。

 構文

**Visual Basic**

```
Public Property Host As String
```

**C#**

```
public string Host {get;set;}
```

## 1.29.8 IPV6 プロパティ

IPV6 オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property IPV6 As String
```

#### C#

```
public string IPV6 {get;set;}
```

## 1.29.9 LDAP プロパティ

LDAP オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property LDAP As String
```

#### C#

```
public string LDAP {get;set;}
```

## 1.29.10 LocalOnly プロパティ

LocalOnly オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property LocalOnly As String
```

#### C#

```
public string LocalOnly {get;set;}
```

## 1.29.11 MyIP プロパティ

MyIP オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property MyIP As String
```

#### C#

```
public string MyIP {get;set;}
```

## 1.29.12 ReceiveBufferSize プロパティ

ReceiveBufferSize オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ReceiveBufferSize As Integer
```

#### C#

```
public int ReceiveBufferSize {get;set;}
```

## 1.29.13 SendBufferSize プロパティ

Send BufferSize オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property SendBufferSize As Integer
```

#### C#

```
public int SendBufferSize {get;set;}
```

## 1.29.14 ServerPort プロパティ

ServerPort オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property ServerPort As String
```

#### C#

```
public string ServerPort {get;set;}
```

## 1.29.15 TDS プロパティ

TDS オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property TDS As String
```

#### C#

```
public string TDS {get;set;}
```

## 1.29.16 Timeout プロパティ

Timeout オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property Timeout As Integer
```

#### C#

```
public int Timeout {get;set;}
```



## 1.29.17 VerifyServerName プロパティ

VerifyServerName オプションを取得または設定します。

### 構文

#### Visual Basic

```
Public Property VerifyServerName As String
```

#### C#

```
public string VerifyServerName {get;set;}
```

## 1.30 SATransaction クラス

SQLトランザクションを表します。

### 構文

#### Visual Basic

```
Public NotInheritable Class SATransaction Inherits  
System.Data.Common.DbTransaction
```

#### C#

```
public sealed class SATransaction : System.Data.Common.DbTransaction
```

### メンバー

SATransaction のすべてのメンバー (継承されたメンバーも含みます) を以下に示します。

#### メソッド

変更子とタイプ	メソッド	説明
public override void	<a href="#">Commit() [339 ページ]</a>	データベーストランザクションをコミットします。
protected override void	<a href="#">Dispose(bool) [339 ページ]</a>	
public override void	<a href="#">Rollback [340 ページ]</a>	トランザクションを保留状態からロールバックします。

変更子とタイプ	メソッド	説明
public unsafe void	<a href="#">Save(string) [341 ページ]</a>	トランザクションの一部をロールバックするために使用できるトランザクションのセーブポイントを作成し、セーブポイント名を指定します。

## プロパティ

変更子とタイプ	プロパティ	説明
public new SACConnection	<a href="#">Connection [342 ページ]</a>	トランザクションに関連付けられている SACConnection オブジェクトです。トランザクションが無効な場合は NULL 参照 (Visual Basic の場合は Nothing) です。
protected override DbConnection	<a href="#">DbConnection [342 ページ]</a>	トランザクションに関連付けられている System.Data.Common.DbConnection を指定します。
public override System.Data.IsolationLevel	<a href="#">IsolationLevel [343 ページ]</a>	このトランザクションの独立性レベルを指定します。
public SAIsolationLevel	<a href="#">SAIsolationLevel [343 ページ]</a>	このトランザクションの拡張された独立性レベルを指定します。

## 備考

SATransaction にはコンストラクタがありません。SATransaction オブジェクトを取得するには、いずれかの BeginTransaction メソッドを使用します。コマンドをトランザクションに関連付けるには、SACCommand.Transaction プロパティを使用します。

詳細については、「トランザクション処理」および「データのアクセスとデータの操作」を参照してください。

このセクションの内容:

### [Commit\(\) メソッド \[339 ページ\]](#)

データベーストランザクションをコミットします。

### [Dispose\(bool\) メソッド \[339 ページ\]](#)

### [Rollback メソッド \[340 ページ\]](#)

トランザクションを保留状態からロールバックします。

### [Save\(string\) メソッド \[341 ページ\]](#)

トランザクションの一部をロールバックするために使用できるトランザクションのセーブポイントを作成し、セーブポイント名を指定します。

### [Connection プロパティ \[342 ページ\]](#)

トランザクションに関連付けられている SACConnection オブジェクトです。トランザクションが無効な場合は NULL 参照 (Visual Basic の場合は Nothing) です。

### [DbConnection プロパティ \[342 ページ\]](#)

トランザクションに関連付けられている System.Data.Common.DbConnection を指定します。

[IsolationLevel プロパティ \[343 ページ\]](#)

このトランザクションの独立性レベルを指定します。

[SAIsolationLevel プロパティ \[343 ページ\]](#)

このトランザクションの拡張された独立性レベルを指定します。

## 関連情報

[BeginTransaction\(\) メソッド \[126 ページ\]](#)

[BeginTransaction\(SAIsolationLevel\) メソッド \[128 ページ\]](#)

[Transaction プロパティ \[90 ページ\]](#)

### 1.30.1 Commit() メソッド

データベーストランザクションをコミットします。

#### 構文

##### Visual Basic

```
Public Overrides Sub Commit ()
```

##### C#

```
public override void Commit ()
```

### 1.30.2 Dispose(bool) メソッド

#### 構文

##### Visual Basic

```
Protected Overrides Sub Dispose (ByVal disposing As Boolean)
```

##### C#

```
protected override void Dispose (bool disposing)
```

## 1.30.3 Rollback メソッド

トランザクションを保留状態からロールバックします。

### オーバーロードリスト

変更子とタイプ	オーバーロード名	説明
public override void	<a href="#">Rollback() [340 ページ]</a>	トランザクションを保留状態からロールバックします。
public unsafe void	<a href="#">Rollback(string) [341 ページ]</a>	トランザクションを保留状態からロールバックします。

このセクションの内容:

[Rollback\(\) メソッド \[340 ページ\]](#)

トランザクションを保留状態からロールバックします。

[Rollback\(string\) メソッド \[341 ページ\]](#)

トランザクションを保留状態からロールバックします。

### 1.30.3.1 Rollback() メソッド

トランザクションを保留状態からロールバックします。

#### 構文

##### Visual Basic

```
Public Overrides Sub Rollback ()
```

##### C#

```
public override void Rollback ()
```

### 備考

このトランザクションがロールバックできるのは、保留状態 (BeginTransaction が呼び出された後だが、Commit が呼び出される前) からのみです。

## 1.30.3.2 Rollback(string) メソッド

トランザクションを保留状態からロールバックします。

### 構文

#### Visual Basic

```
Public Sub Rollback (ByVal savePoint As String)
```

#### C#

```
public unsafe void Rollback (string savePoint)
```

### パラメータ

**savePoint** ロールバック先のセーブポイントの名前。

### 備考

このトランザクションがロールバックできるのは、保留状態 (BeginTransaction が呼び出された後だが、Commit が呼び出される前) からのみです。

## 1.30.4 Save(string) メソッド

トランザクションの一部をロールバックするために使用できるトランザクションのセーブポイントを作成し、セーブポイント名を指定します。

### 構文

#### Visual Basic

```
Public Sub Save (ByVal savePoint As String)
```

#### C#

```
public unsafe void Save (string savePoint)
```

### パラメータ

**savePoint** ロールバック先のセーブポイントの名前。

## 1.30.5 Connection プロパティ

トランザクションに関連付けられている `SACConnection` オブジェクトです。トランザクションが無効な場合は `NULL` 参照 (Visual Basic の場合は `Nothing`) です。

### 構文

#### Visual Basic

```
Public ReadOnly Shadows Property Connection As SACConnection
```

#### C#

```
public new SACConnection Connection {get;}
```

### 備考

1つのアプリケーションに複数のデータベース接続があり、各接続に0以上のトランザクションがある場合があります。このプロパティを使用して、`BeginTransaction` によって作成された特定のトランザクションに関連付けられた接続オブジェクトを確認できます。

## 1.30.6 DbConnection プロパティ

トランザクションに関連付けられている `System.Data.Common.DbConnection` を指定します。

### 構文

#### Visual Basic

```
Protected ReadOnly Overrides Property DbConnection As DbConnection
```

#### C#

```
protected override DbConnection DbConnection {get;}
```

### 戻り値

トランザクションに関連付けられている `System.Data.Common.DbConnection`。

## 1.30.7 IsolationLevel プロパティ

このトランザクションの独立性レベルを指定します。

### 構文

#### Visual Basic

```
Public ReadOnly Overrides Property IsolationLevel As System.Data.IsolationLevel
```

#### C#

```
public override System.Data.IsolationLevel IsolationLevel {get;}
```

### 備考

このトランザクションの IsolationLevel。次のいずれかを指定してください。

- Unspecified
- Chaos
- ReadUncommitted
- ReadCommitted
- RepeatableRead
- Serializable
- Snapshot

デフォルトは ReadCommitted です。

## 1.30.8 SAIsolationLevel プロパティ

このトランザクションの拡張された独立性レベルを指定します。

### 構文

#### Visual Basic

```
Public ReadOnly Property SAIsolationLevel As SAIsolationLevel
```

#### C#

```
public SAIsolationLevel SAIsolationLevel {get;}
```

## 備考

このトランザクションの `SAIsolationLevel`。次のいずれかを指定してください。

- Unspecified
- Chaos
- ReadUncommitted
- ReadCommitted
- RepeatableRead
- Serializable
- Snapshot
- StatementSnapshot
- ReadOnlySnapshot

デフォルトは `ReadCommitted` です。

並列トランザクションはサポートされていません。このため、`SAIsolationLevel` はトランザクション全体に適用されます。

## 1.31 SAInfoMessageEventHandler(object, SAInfoMessageEventArgs) デリゲート

`SACConnection` オブジェクトの `SACConnection.InfoMessage` イベントを処理するメソッドを表します。

### 構文

#### Visual Basic

```
Public Delegate Sub SAInfoMessageEventHandler (
    ByVal obj As Object,
    ByVal args As SAInfoMessageEventArgs
) As delegate void
```

#### C#

```
public delegate void SAInfoMessageEventHandler (
    object obj,
    SAInfoMessageEventArgs args
);
```

## 関連情報

[SACConnection クラス \[118 ページ\]](#)

[InfoMessage イベント \[146 ページ\]](#)



## 1.32 SARowsCopiedEventHandler(object, SARowsCopiedEventArgs) デリゲート

SABulkCopy の SABulkCopy.SARowsCopied イベントを処理するメソッドを表します。

### 構文

#### Visual Basic

```
Public Delegate Sub SARowsCopiedEventHandler (  
    ByVal sender As Object,  
    ByVal rowsCopiedEventArgs As SARowsCopiedEventArgs  
) As delegate void
```

#### C#

```
public delegate void SARowsCopiedEventHandler (  
    object sender,  
    SARowsCopiedEventArgs rowsCopiedEventArgs  
);
```

### 関連情報

[SABulkCopy クラス \[14 ページ\]](#)

## 1.33 SARowUpdatedEventHandler(object, SARowUpdatedEventArgs) デリゲート

SADDataAdapter の RowUpdated イベントを処理するメソッドを表します。

### 構文

#### Visual Basic

```
Public Delegate Sub SARowUpdatedEventHandler (  
    ByVal sender As Object,  
    ByVal e As SARowUpdatedEventArgs  
) As delegate void
```

#### C#

```
public delegate void SARowUpdatedEventHandler (  
    object sender,  
    SARowUpdatedEventArgs e  
);
```

## 1.34 SARowUpdatingEventHandler(object, SARowUpdatingEventArgs) デリゲート

SDataAdapter の RowUpdating イベントを処理するメソッドを表します。

### 構文

#### Visual Basic

```
Public Delegate Sub SARowUpdatingEventHandler (
    ByVal sender As Object,
    ByVal e As SARowUpdatingEventArgs
) As delegate void
```

#### C#

```
public delegate void SARowUpdatingEventHandler (
    object sender,
    SARowUpdatingEventArgs e
);
```

## 1.35 SABulkCopyOptions 列挙

SABulkCopy のインスタンスで使用する 1 つ以上のオプションを指定するビット単位フラグです。

### 構文

#### Visual Basic

```
Public Enum SABulkCopyOptions
```

#### C#

```
enum SABulkCopyOptions
```

### メンバー

メンバー名	説明	値
Default	この値だけを指定すると、デフォルトの動作が使用されます。 デフォルトでは、トリガは有効です。	0x0

メンバー名	説明	値
DoNotFireTriggers	これを指定すると、トリガが起動しません。  トリガを無効にするには、DBA パーミッションが必要です。トリガは、WriteToServer の開始時に接続に対して無効となり、メソッドの終了時に値がリストアされます。	0x1
KeepIdentity	これを指定すると、IDENTITY カラムにコピーされる元の値が保持されます。  デフォルトでは、送信先テーブルでは新しい ID 番号が生成されます。	0x2
TableLock	これを指定すると、コマンド LOCK TABLE table_name WITH HOLD IN SHARE MODE を使用してテーブルがロックされます。  このロックは、接続が閉じるまで続きます。	0x4
UseInternalTransaction	これを指定すると、バルクコピーオペレーションの各バッチがトランザクションの中で実行されます。  これが指定されない場合、トランザクションは使用されません。このオプションを指定し、コンストラクタに SATransaction オブジェクトも指定すると、System.ArgumentException が発生します。	0x8

## 備考

SABulkCopyOptions 列挙は、SABulkCopy オブジェクトを構築し、WriteToServer メソッドの動作を指定する場合に使用します。

CheckConstraints オプションと KeepNulls オプションはサポートされません。

## 関連情報

[SABulkCopy クラス \[14 ページ\]](#)

## 1.36 SADBType 列挙

データベースサーバの .NET データベースデータ型を列挙します。

### 構文

#### Visual Basic

```
Public Enum SADBType
```

#### C#

```
enum SADBType
```

### メンバー

メンバー名	説明
BigInt	符号付き 64 ビット整数値
Binary	指定された最大長のバイナリデータ。 列挙値 Binary と VarBinary は同等のエイリアスです。
Bit	1 ビットフラグ
Char	指定された長さの文字データ。 このタイプは常に Unicode 文字をサポートします。Char 型と VarChar 型は、完全に互換性があります。
Date	日付情報
DateTime	タイムスタンプ情報 (日付、時刻)。 列挙値 DateTime と TimeStamp は同等のエイリアスです。
DateTimeOffset	タイムスタンプ情報 (日付、時刻) オフセット。
Decimal	精度と桁数が指定された正確な数値データ。 列挙値 Decimal と Numeric は同等のエイリアスです。
Double	倍精度浮動小数点数 (8 バイト)
Float	単精度浮動小数点数 (4 バイト)。 列挙値 Float と Real は同等のエイリアスです。
Image	任意の長さのバイナリデータを格納します。
Integer	符号なし 32 ビット整数値。
LongBinary	可変長のバイナリデータ

メンバー名	説明
LongNVarchar	NCHAR 文字セットの可変長文字データ。 このタイプは常に Unicode 文字をサポートします。
LongVarbit	可変長のビット配列
LongVarchar	可変長の文字データ。 このタイプは常に Unicode 文字をサポートします。
Money	通貨データ
NChar	32767 文字までの Unicode 文字データを格納します。
NText	任意の長さの Unicode 文字データを格納します。
Numeric	精度と桁数が指定された正確な数値データ。 列挙値 Decimal と Numeric は同等のエイリアスです。
NVarChar	32767 文字までの Unicode 文字データを格納します。
Real	単精度浮動小数点数 (4 バイト)。 列挙値 Float と Real は同等のエイリアスです。
SmallDateTime	TIMESTAMP として実装されたドメイン
SmallInt	符号付き 16 ビット整数値
SmallMoney	100 万通貨単位未満の通貨データを格納します。
SysName	任意の長さの文字データを格納します。
Text	任意の長さの文字データを格納します。
Time	時刻情報
TimeStamp	タイムスタンプ情報 (日付、時刻)。 列挙値 DateTime と TimeStamp は同等のエイリアスです。
TimeStampWithTimeZone	タイムスタンプ情報 (日付、時刻、タイムゾーン)。 列挙値 DateTime と TimeStamp は同等のエイリアスです。
TinyInt	符号なし 8 ビット整数値
UniqueIdentifier	ユニバーサルユニーク識別子 (UUID/GUID)
UniqueIdentifierStr	CHAR(36) として実装されたドメイン。 UniqueIdentifierStr は、Microsoft SQL Server の uniqueidentifier カラムをマッピングするとき、リモートデータアクセスに使用されます。
UnsignedBigInt	符号なし 64 ビット整数値
UnsignedInt	符号なし 32 ビット整数値。
UnsignedSmallInt	符号なし 16 ビット整数値

メンバー名	説明
VarBinary	指定された最大長のバイナリデータ。 列挙値 <i>Binary</i> と <i>VarBinary</i> は同等のエイリアスです。
VarBit	長さが 1 ~ 32767 ビットのビット配列
VarChar	指定された最大長の文字データ。 このタイプは常に Unicode 文字をサポートします。Char 型と VarChar 型は、完全に互換性があります。
Xml	XML データ。 この型は、任意の長さの文字データを格納し、XML 文書を格納するために使用されます。

## 備考

下の表には、各 SADBType との互換性がある .NET 型がリストされています。整数型の場合、テーブルのカラムは、常により小さい整数型を使用して設定できるほか、実際の値がその型の範囲内にあるかぎり、より大きい型を使用して設定することも可能です。

SADBType	互換性のある .NET 型	C# 組み込みタイプ	Visual Basic 組み込みタイプ
<i>BigInt</i>	System.Int64	long	Long
<i>Binary</i> 、 <i>VarBinary</i>	System.Byte[], または System.Guid (サイズが 16 の場合)	byte[]	Byte()
<i>Bit</i>	System.Boolean	bool	Boolean
<i>Char</i> 、 <i>VarChar</i>	System.String	String	String
<i>Date</i>	System.DateTime	DateTime (組み込みタイプなし)	Date
<i>DateTime</i> 、 <i>TimeStamp</i>	System.DateTime	DateTime (組み込みタイプなし)	DateTime
<i>DateTimeOffset</i>	System.DateTimeOffset	DateTimeOffset (組み込みタイプなし)	DateTimeOffset
<i>Decimal</i> 、 <i>Numeric</i>	System.String	decimal	Decimal
<i>Double</i>	System.Double	double	Double
<i>Float</i> 、 <i>Real</i>	System.Single	float	Single
<i>Image</i>	System.Byte[]	byte[]	Byte()
<i>Integer</i>	System.Int32	int	Integer
<i>LongBinary</i>	System.Byte[]	byte[]	Byte()
<i>LongNVarChar</i>	System.String	String	String
<i>LongVarChar</i>	System.String	String	String

SADbType	互換性のある .NET 型	C# 組み込みタイプ	Visual Basic 組み込みタイプ
<i>Money</i>	System.String	decimal	Decimal
<i>NChar</i>	System.String	String	String
<i>NText</i>	System.String	String	String
<i>Numeric</i>	System.String	decimal	Decimal
<i>NVarChar</i>	System.String	String	String
<i>SmallDateTime</i>	System.DateTime	DateTime (組み込みタイプなし)	Date
<i>SmallInt</i>	System.Int16	short	Short
<i>SmallMoney</i>	System.String	decimal	Decimal
<i>SysName</i>	System.String	String	String
<i>Text</i>	System.String	String	String
<i>Time</i>	System.TimeSpan	TimeSpan (組み込みタイプなし)	TimeSpan (組み込みタイプなし)
<i>TimeStamp</i>	System.DateTime	DateTime (組み込みタイプなし)	Date
<i>TimeStampWithTimeZone</i>	System.DateTimeOffset	DateTimeOffset (組み込みタイプなし)	DateTimeOffset
<i>TinyInt</i>	System.Byte	byte	Byte
<i>UniquelIdentifier</i>	System.Guid	Guid (組み込みタイプなし)	Guid (組み込みタイプなし)
<i>UniquelIdentifierStr</i>	System.String	String	String
<i>UnsignedBigInt</i>	System.UInt64	ulong	UInt64 (組み込みタイプなし)
<i>UnsignedInt</i>	System.UInt32	uint	UInt64 (組み込みタイプなし)
<i>UnsignedSmallInt</i>	System.UInt16	ushort	UInt64 (組み込みタイプなし)
<i>Xml</i>	System.Xml	String	String

長さが 16 のバイナリカラムには、UniquelIdentifier 型との完全な互換性があります。

## 関連情報

[GetFieldType\(int\) メソッド \[222 ページ\]](#)

[GetDataTypeName\(int\) メソッド \[217 ページ\]](#)

## 1.37 SAIsolationLevel 列挙

データベースサーバの独立性レベルを指定します。

### 構文

#### Visual Basic

```
Public Enum SAIsolationLevel
```

#### C#

```
enum SAIsolationLevel
```

### メンバー

メンバー名	説明
Chaos	<p>この独立性レベルはサポートされていません。</p> <p>この独立性レベルで <code>SACConnection.BeginTransaction</code> を呼び出すと、例外がスローされます。</p> <p>詳細については、「<code>BeginTransaction</code> メソッド」を確認してください。</p>
ReadCommitted	<p>独立性レベル 1 と同等な動作を設定します。</p> <p>この独立性レベルはダーティリードを防ぎますが、繰り返し不可能読み出しと幻ローを許可します。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>
ReadUncommitted	<p>独立性レベル 0 と同等な動作を設定します。</p> <p>この独立性レベルはダーティリード、繰り返し不可能読み出し、幻ローを許可します。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>
RepeatableRead	<p>独立性レベル 2 と同等な動作を設定します。</p> <p>この独立性レベルはダーティリードを防ぎ、繰り返し可能読み出しを保証します。ただし、幻ローを許可します。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>
Serializable	<p>独立性レベル 3 と同等な動作を設定します。</p> <p>この独立性レベルはダーティリードを防ぎ、繰り返し可能読み出しを保証し、幻ローを防ぎます。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>



メンバー名	説明
Snapshot	<p>トランザクションが最初のローの読み込み、挿入、更新、または削除を行った時点から、コミットされたデータのスナップショットを使用します。</p> <p>この独立性レベルでは、トランザクションが最初のローの読み込みまたは更新を行った時点から、コミットされたデータのスナップショットを使用します。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>
Unspecified	<p>この独立性レベルはサポートされていません。</p> <p>この独立性レベルで <code>SqlConnection.BeginTransaction</code> を呼び出すと、例外がスローされます。</p> <p>詳細については、「BeginTransaction メソッド」を確認してください。</p>
ReadOnlySnapshot	<p>読み込み専用の文についてのみ、データベースから最初のローが読み込まれた時点から、コミットされたデータのスナップショットを使用します。</p> <p>繰り返し不可能読み出しと幻ローは、1つのトランザクション内で発生することはありますが、1つの文の中で発生することはありません。更新可能な文については、<code>updatable_statement_isolation</code> オプションに指定された独立性レベル (0 (デフォルト)、1、2、3 のいずれか) を使用します。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>
StatementSnapshot	<p>文で最初のローが読み込まれた時点から、コミットされたデータのスナップショットを使用します。</p> <p>トランザクション内の各文で参照されるデータのスナップショットはそれぞれ異なる時点のものになります。</p> <p>個々の文について、データベースから最初のローが読み込まれた時点から、コミットされたデータのスナップショットを使用します。繰り返し不可能読み出しと幻ローは、1つのトランザクション内で発生することはありますが、1つの文の中で発生することはありません。</p> <p>詳細については、「独立性レベルと一貫性」を参照してください。</p>

## 備考

このクラス引数は `System.Data.IsolationLevel` クラスです。

.NET データプロバイダは、スナップショットアイソレーションのレベルなど、すべてのデータベースサーバ独立性レベルをサポートします。スナップショットアイソレーションを使用するには、`SAIsolationLevel.Snapshot`、`SAIsolationLevel.ReadOnlySnapshot`、`SAIsolationLevel.StatementSnapshot` のいずれかを、`BeginTransaction` へのパラメータとして指定します。`BeginTransaction` はオーバーロードされているため、`IsolationLevel` または `SAIsolationLevel` を指定できます。2つの列挙内の値は同じですが、`ReadOnlySnapshot` と `StatementSnapshot` は例外で、`SAIsolationLevel` にのみ存在します。`SATransaction` には、`SAIsolationLevel` を取得する `SAIsolationLevel` という名前の新しいプロパティがあります。

詳細については、「スナップショットアイソレーション」を参照してください。

## 1.38 SAMessageType 列挙

メッセージのタイプを示します。

### 構文

#### Visual Basic

```
Public Enum SAMessageType
```

#### C#

```
enum SAMessageType
```

### メンバー

メンバー名	説明	値
Action	タイプ ACTION のメッセージ	2
Info	タイプ INFO のメッセージ	0
Status	タイプ STATUS のメッセージ	3
Warning	タイプ WARNING のメッセージ	1

### 備考

次のいずれかを指定してください。Action、Info、Status、または Warning。

---

## 2 このマニュアルの印刷、再生、および再配布

次の条件に従うかぎり、このマニュアルの全部または一部を使用、印刷、再生、配布することができます。

1. ここに示したものとそれ以外のすべての著作権と商標の表示をすべてのコピーに含めること。
2. マニュアルに変更を加えないこと。
3. SAP 以外の人間がマニュアルの著者または情報源であるかのように示す一切の行為をしないこと。

ここに記載された情報は事前の通知なしに変更されることがあります。

# 重要免責事項および法的情報

## コードサンプル

この文書に含まれるソフトウェアコード及び / 又はコードライン / 文字列 (「コード」) はすべてサンプルとしてのみ提供されるものであり、本稼動システム環境で使用することが目的ではありません。「コード」は、特定のコードの構文及び表現規則を分かりやすく説明及び視覚化することのみを目的としています。SAP は、この文書に記載される「コード」の正確性及び完全性の保証を行いません。更に、SAP は、「コード」の使用により発生したエラー又は損害が SAP の故意又は重大な過失が原因で発生させたものでない限り、そのエラー又は損害に対して一切責任を負いません。

## アクセシビリティ

この SAP 文書に含まれる情報は、公開日現在のアクセシビリティ基準に関する SAP の最新の見解を表明するものであり、ソフトウェア製品のアクセシビリティ機能の確実な提供方法に関する拘束力のあるガイドラインとして意図されるものではありません。SAP は、この文書に関する一切の責任を明確に放棄するものです。ただし、この免責事項は、SAP の意図的な違法行為または重大な過失による場合は、適用されません。さらに、この文書により SAP の直接的または間接的な契約上の義務が発生することは一切ありません。

## ジェンダーニュートラルな表現

SAP 文書では、可能な限りジェンダーニュートラルな表現を使用しています。文脈により、文書の読者は「あなた」と直接的な呼ばれ方をされたり、ジェンダーニュートラルな名詞 (例:「販売員」又は「勤務日数」) で表現されます。ただし、男女両方を指すとき、三人称単数形の使用が避けられない又はジェンダーニュートラルな名詞が存在しない場合、SAP はその名詞又は代名詞の男性形を使用する権利を有します。これは、文書を分かりやすくするためです。

## インターネットハイパーリンク

SAP 文書にはインターネットへのハイパーリンクが含まれる場合があります。これらのハイパーリンクは、関連情報を見いだすヒントを提供することが目的です。SAP は、この関連情報の可用性や正確性又はこの情報が特定の目的に役立つことの保証は行いません。SAP は、関連情報の使用により発生した損害が、SAP の重大な過失又は意図的な違法行為が原因で発生したものでない限り、その損害に対して一切責任を負いません。すべてのリンクは、透明性を目的に分類されています (<http://help.sap.com/disclaimer> を参照)。



[go.sap.com/registration/  
contact.html](http://go.sap.com/registration/contact.html)

© 2016 SAP SE or an SAP affiliate company. All rights reserved.

本書のいかなる部分も、SAP SE 又は SAP の関連会社の明示的な許可なくして、いかなる形式でも、いかなる目的にも複製又は伝送することはできません。本書に記載された情報は、予告なしに変更されることがあります。SAP SE 及びその頒布業者によって販売される一部のソフトウェア製品には、他のソフトウェアベンダーの専有ソフトウェアコンポーネントが含まれています。製品仕様は、国ごとに変わる場合があります。

これらの文書は、いかなる種類の表明又は保証もなしで、情報提供のみを目的として、SAP SE 又はその関連会社によって提供され、SAP 又はその関連会社は、これら文書に関する誤記脱落等の過失に対する責任を負うものではありません。SAP 又はその関連会社の製品及びサービスに対する唯一の保証は、当該製品及びサービスに伴う明示的な保証がある場合に、これに規定されたものに限られます。本書のいかなる記述も、追加の保証となるものではありません。

本書に記載される SAP 及びその他の SAP の製品やサービス、並びにそれらの個々のロゴは、ドイツ及びその他の国における SAP SE (又は SAP の関連会社) の商標若しくは登録商標です。本書に記載されたその他すべての製品およびサービス名は、それぞれの企業の商標です。

商標に関する詳細の情報や通知については、<http://www.sap.com/corporate-en/legal/copyright/index.epx> をご覧ください。