

# JSON Web サービス・サポート

JSON (JavaScript Object Notation) は、データ構造の表示や変換を理想的に実現する、シンプルかつ軽量なデータ変換フォーマットです。JSON は言語非依存のプログラミングですが、JSON オブジェクトでデータ構造を簡単に再形成したり使用できる性能が評価され、JavaScript で最も多く使用されています。SQL Anywhere Panorama は、組み込み型 HTTP サーバを使用して Web サービスを要求し、JSON がフォーマットした結果セットを返します。この結果セットは、さまざまなキー値のペアをとって返されます。各キーは、結果セットのカラムに対応しています。

## JSON Web サービスのサンプル

このサンプルでは、SQL Anywhere デモ・データベースを使用して、JSON フォーマットで従業員リストを返し、インターネット・ブラウザでそのリストを表示します。JSON リストの評価を行い、従業員名が印刷されます。JSON の従業員リストは、AJAX 要求をサーバに返すことによって取得されます。ここでは、SQL Anywhere の組み込み型 HTTP サーバを使用するため、JavaScript が有効なインターネット・ブラウザが必要です。

この例では、サンプルの HTML ファイルを使用して JSON Web サービスを呼び出し、その結果セットを表示します。[このアーカイブをダウンロード](#)して、作業ディレクトリに抽出してください。

1. 稼働中の SQL Anywhere データベース・サーバをすべてシャット・ダウンします。

Windows の場合、コマンド・プロンプトを開き、以下のコマンドを実行します (ファイル *json.html* が同じディレクトリ内にあることを確認します)。

```
dbsrv11 "%SQLANYSAMP11%\demo.db" -xs http{port=8080}
```

Linux の場合、ターミナルを開き、以下の 2 つのコマンドを実行します (ファイル *json.html* が同じディレクトリ内にあることを確認します)。

```
source /opt/sqlanywhere/samples/sample_config.sh
```

```
dbsrv11 $SQLANYSAMP11/sqlanywhere/demo.db -ud -xs http{port=8080}
```

最後のコマンドで SQL Anywhere データベース・サーバを起動し、ポート 8080 で HTTP 要求を受信するように指示します。8080 がすでに使用中の場合は、別のポート番号を使用することもできます。

- Interactive SQL を開いて以下のコマンドを実行し、デモ・データベースに接続します。

```
dbisql -c "eng=demo;uid=dba;pwd=sql"
```

- ローカル (作業) ディレクトリから HTML ページを読み込むストアド・プロシージャを作成します。以下のコードをコピーして、Interactive SQL に貼り付けます。【F5】を押して実行します。

```
CREATE PROCEDURE sp_root()
BEGIN
    CALL dbo.sa_set_http_header( 'Content-Type', 'text/html' );
    SELECT xp_read_file( 'json.html' );
END;
```

sa\_set\_http\_header() システム・プロシージャを呼び出し、Web サービスを許可して HTTP ヘッダを設定できるようにします。

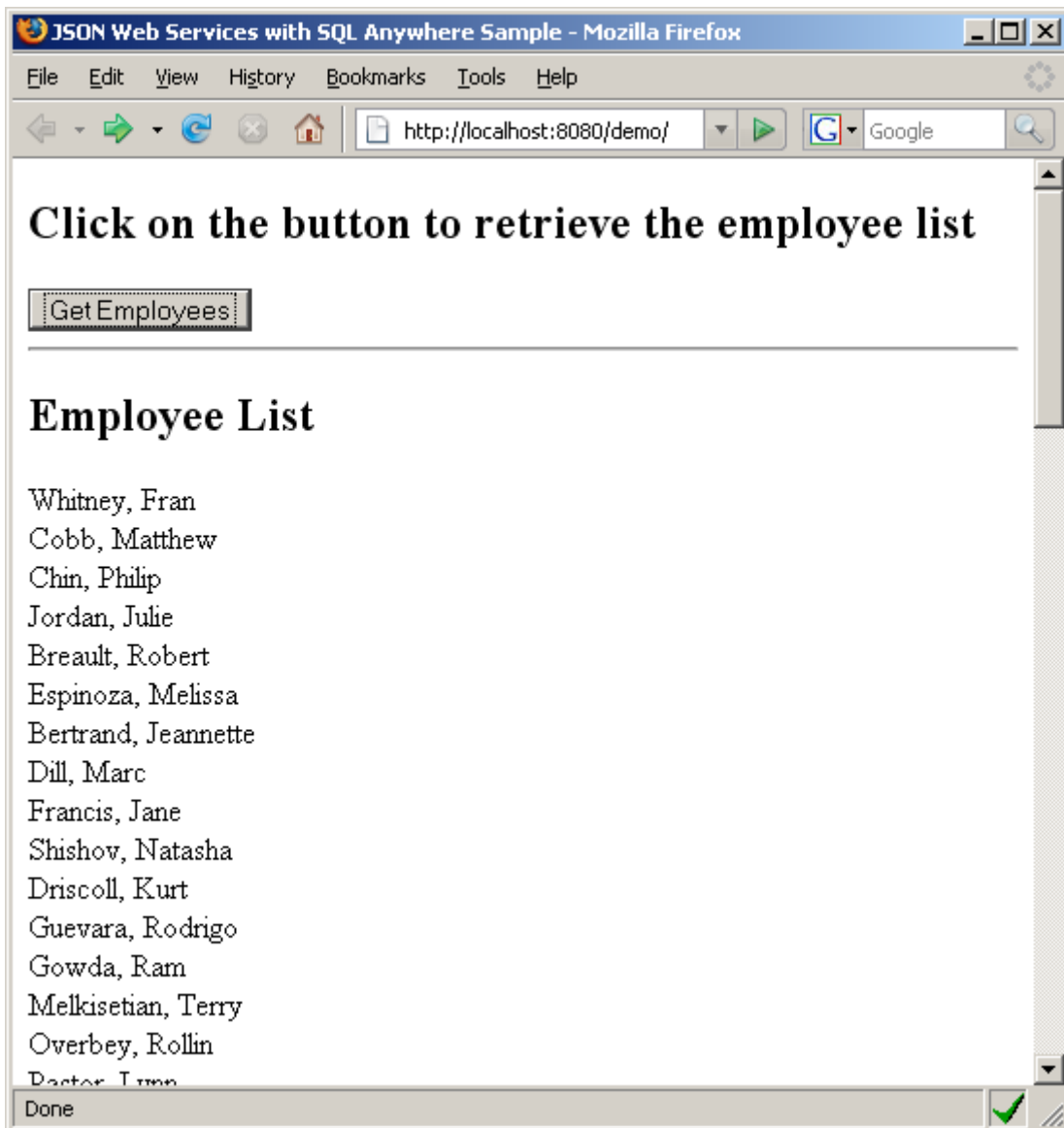
- HTML ページを返すルート Web サービスを作成します。以下のコードをコピーして、Interactive SQL に貼り付けます。【F5】を押して実行します。

```
CREATE SERVICE "root"
TYPE 'RAW'
AUTHORIZATION OFF
USER DBA
AS CALL sp_root();
```

- ファイル *json.html* で呼び出す、従業員の Web サービスを作成します。以下のコードをコピーして、Interactive SQL に貼り付けます。【F5】を押して実行します。

```
CREATE SERVICE "employees"
TYPE 'JSON'
AUTHORIZATION OFF
USER DBA
AS SELECT * FROM "Employees";
```

- インターネット・ブラウザを開いて、<http://localhost:8080/demo/> にナビゲートします。
- 【Get Employees】 ボタンをクリックすると、従業員の Web サービスに対する AJAX 要求が表示されます。デモ・データベースの従業員リストが表示されます。



8. HTML ページで、ユーザ関数 "getEmployees" を呼び出します。この関数は、XMLHttpRequest オブジェクトを使用して要求を実行します。その後、JSON フォーマットの応答として、応答テキストを評価します。以下に、関連コードの抜粋を示します。

```
// Create an XMLHttpRequest
try {
    // Firefox, Opera 8.0+, Safari, IE7+
    jsonHttp = new XMLHttpRequest();
} catch (e) {
    // Internet Explorer
    try {
        jsonHttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
```

```

    jsonHttp = new ActiveXObject("Microsoft.XMLHTTP");
  }
}

// Evaluate JSON response text and assign to an object.
// employeeList becomes an array of the result set rows.
// Individual fields can be accessed by the format employeeList[<Row
Number>].<Column Name>
employeeList = eval("(" + jsonHttp.responseText + ")");

// Iterate through list and print names in "Lastname, Firstname"
format
document.getElementById("employeeListDiv").innerHTML =
"<h2>Employee List</h2>";
for (x in employeeList)
{
  document.getElementById("employeeListDiv").innerHTML +=
employeeList[x].Surname + ", " + employeeList[x].GivenName +
"<br>";
}

```

9. ファイル *json.html* のコンテンツをさらに調査して、JSON 応答の使用方法に関する詳細を確認することもできます。
10. インターネット・ブラウザを終了します。
11. このサンプルで使用したデータベース・オブジェクトを削除する場合は、Interactive SQL で以下の文を実行します。

```

DROP SERVICE "employees";
DROP SERVICE "root";
DROP PROCEDURE sp_root;

```

12. Interactive SQL および SQL Anywhere データベース・サーバをシャット・ダウンします。

## 結び

SQL Anywhere の組み込み型 HTTP サーバは、Web サービス要求を作成できるだけでなく、HTML や XML、JSON などのさまざまなタイプの結果セットをフォーマットすることが可能です。これにより、クライアント・アプリケーションに対し、データにアクセスしてそのデータを最適なフォーマットで表示するための代替方法を提供しています。