Using SQL queries programmatically

In the previous recipe, we discussed how to use a UFT DB checkpoin
Here, we will show you how to execute a SQL statement using VBScri
code.

Getting ready

We will use the function library `DB_Func.vbs` as in the previous recipe
Establishing and closing a database connection.

How to do it...

In our custom class `DB_Handler`, we will add a new private `m_oRecordset` field to hold the results of our query and a new method `executeSQLQuery(SQLQuery)`, which, of course, accepts a string with a valid SQL query as the argument:

```
Private m_oRecordset

function executeSQLQuery(SQLQuery)
    Set m_oRecordset = m_oDBConnection.Execute(SQLQuery)
End Function
```

Additionally, in our `Action1` datasheet, we would call the `executeSQLQuery(SQLQuery)` method by passing our SQL string as the argument:

```
call oDBHandler.executeSQLQuery(SQLQuery)
```

As mentioned earlier, the method would store the returned `Recordset` in the `m_oRecordset` field. Then, we will be able to perform operations with these data, such as making comparison between the expected and actual results.

How it works...

With our `oDBHandler` object already initialized and having an open DB connection, as shown in the previous recipe, we call the `executeSQLQuery` member function passing our SQL query string as the parameter. Inside the `m_oRecordset` method, member field is assigned the result of the `Execute` method of the `ADODB.Connection` object.