

# illustrates how to perform a SELECT statement using ADO.NET

```
/*
```

```
Mastering Visual C# .NET
```

```
by Jason Price, Mike Gunderloy
```

```
Publisher: Sybex;
```

```
ISBN: 0782129110
```

```
*/
```

```
/*
```

```
Example23_1.cs illustrates how to perform a SELECT statement  
using ADO.NET
```

```
*/
```

```
using System;
```

```
using System.Data;
```

```
using System.Data.SqlClient;
```

```
public class Example23_1
```

```
{
```

```
public static void Main()
```

```
{
```

```
// step 1: formulate a string containing the details of the
```

```
// database connection
```

```
string connectionString =
```

```
“server=localhost;database=Northwind;uid=sa;pwd=sa”;
```

```
// step 2: create a SqlConnection object to connect to the
```

```
// database, passing the connection string to the constructor
```

```
SqlConnection mySqlConnection =
```

```
new SqlConnection(connectionString);
```

```

// step 3: formulate a SELECT statement to retrieve the
// CustomerID, CompanyName, ContactName, and Address
// columns for the first ten rows from the Customers table
string selectString =
"SELECT CustomerID, CompanyName, ContactName, Address " +
"FROM Customers " +
"WHERE CustomerID < 'BSBEV'"; // step 4: create a SqlCommand
object to hold the SELECT statement SqlCommand mySqlCommand =
mySqlConnection.CreateCommand(); // step 5: set the
CommandText property of the SqlCommand object to // the SELECT
string mySqlCommand.CommandText = selectString; // step 6:
create a SqlDataAdapter object SqlDataAdapter mySqlDataAdapter
= new SqlDataAdapter(); // step 7: set the SelectCommand
property of the SqlDataAdapter object // to the SqlCommand object
mySqlDataAdapter.SelectCommand = mySqlCommand; // step 8:
create a DataSet object to store the results of // the SELECT
statement DataSet myDataSet = new DataSet(); // step 9: open
the database connection using the // Open() method of the
SqlConnection object mySqlConnection.Open(); // step 10: use
the Fill() method of the SqlDataAdapter object to // retrieve
the rows from the table, storing the rows locally // in a
DataTable of the DataSet object Console.WriteLine("Retrieving
rows from the Customers table"); string dataTable_name =
"Customers"; mySqlDataAdapter.Fill(myDataSet, dataTable_name);
// step 11: get the DataTable object from the DataSet object
DataTable myDataTable = myDataSet.Tables[dataTable_name]; //
step 12: display the columns for each row in the DataTable, //
using a DataRow object to access each row in the DataTable
foreach (DataRow myDataRow in myDataTable.Rows) {
Console.WriteLine("CustomerID = " + myDataRow["CustomerID"]);
Console.WriteLine("CompanyName          =          " +
myDataRow["CompanyName"]); Console.WriteLine("ContactName = "
+ myDataRow["ContactName"]); Console.WriteLine("Address = "
+ myDataRow["Address"]); } // step 13: close the database
connection using the Close() method // of the SqlConnection
object created in Step 2 mySqlConnection.Close(); } }
[/csharp]

```