

Advanced Programming Language (630501)

Fall 2011/2012 – Lecture Notes # 24

Retrieving Real-Time Data II

- The Command Object
- Programming Examples

The Command Object

Creating and Configuring a Command Object

- To create a Command object. The **CreateCommand()** method of the Connection object does that.

The following line shows how to create the object:

```
SqlCommand addressCommand = addressConnection.CreateCommand();
```

- The Command object needs to have some properties set to be able to retrieve the data from the data source:
- The **CommandText** property defines the text of the **SELECT** statement or the **stored procedure**.

The following code lines declare and configure the Command object to run the **SP_SelectUsers** stored procedure on the database.

```
addressCommand.CommandText = "SP_SelectUsers";
```

```
addressCommand.CommandType = CommandType.StoredProcedure;
```

- Create a command object that encapsulates an SQL statement. The command object then uses an existing connection to perform the requested operation. For example, the following code segment creates a command based on a connection:

```
SqlCommand cmCategories = new SqlCommand("SELECT * FROM Categories", sqlCn);
```

- The command object has a number of properties and methods that are used to manipulate the object.

The Members of the Command Object

Member	Description
CommandText	The text that defines the command object
CommandType	The type of the command, either command text (generic), SQL statements, stored procedure, or DirectTable
Connection	The connection to use for this command object
ExecuteNonQuery	Executes the command that performs some data processing;

ADO.NET Programming

	returns the number of rows affected
ExecuteReader	Executes the command, returning a rowset
ExecuteScalar	Executes the command, returning a single value

Programming Examples

Example 24-1

```

string constr = @"Provider=Microsoft.jet.OleDb.4.0; Data Source =
d:\Northwind.mdb ";
string sqlstr = "select * from Customers";
OleDbConnection conn = new OleDbConnection(constr);
conn.Open();
OleDbCommand comm = new OleDbCommand();
OleDbDataReader dr;
comm.CommandText = sqlstr;
comm.Connection = conn;
comm.CommandTimeout = 20;
dr = comm.ExecuteReader(CommandBehavior.CloseConnection);
//output into mobile page
Response.Write("<table border='3' align='center'>");
Response.Write("<tr><th>Customer ID</th><th>Company Name</th></tr>");
while (dr.Read())
{
Response.Write("<tr><td>" + dr.GetString(0) +
"</td><td>" + dr.GetString(1) +
"</th></tr>");
}
Response.Write("</table>");
//output into mobile controls First row
dr.Read();
this.TextBox1.Text = dr["CustomerID"].ToString();
this.TextBox2.Text = dr[2].ToString();
this.TextBox3.Text = dr[3].ToString();
//output into mobile controls all the rows
string str = " ";
while (dr.Read())
{

```

ADO.NET Programming

```
str += dr.GetString(0) + "---" + dr.GetString(1) + "---";
}
this.TextBox3.Text = str;
this.Label1.Text = str;
dr.Close();
conn.Close();
```

Example 24-2

```
private void Page_Load (object sender, System.EventArgs e)
{
txtName.Text=lstAddress.Selection.Text;
int lstIndex = Int32.Parse(lstAddress.Selection.Value);
SqlConnection addrCN;
SqlCommand addrCom;
SqlCommand phoneCom;
SqlDataReader addrDR;
SqlDataReader phoneDR;
string strCon = "user id=sa;initial catalog=Address;" +
"data source=KENSABBEN\\KEN";
string strCom = "SELECT Address1, Address2, City, Province,
PostalCode, Country " + "FROM Addresses WHERE NameID = " + intIndex;
string strCom1 = "SELECT PhoneNumber FROM PhoneNumbers WHERE NameID =
" + intIndex + "AND Type = 'Home'";
addrCN = new SqlConnection(strCon);
try
{
addrCN.Open();
addrCom = new SqlCommand(strCom, addrCN);
phoneCom = new SqlCommand(strCom1, addrCN);
addrDR = addrCom.ExecuteReader();
addrDR.Read();
// The data population commands will go here.
txtAddress1.Text = addrDR.GetString(0);
txtAddress2.Text = addrDR.GetString(1);
txtCity.Text = addrDR.GetString(2);
txtProvince.Text = addrDR.GetString(3);
txtPostalCode.Text = addrDR.GetString(4);
```

ADO.NET Programming

```
txtCountry.Text = addrDR.GetString(5);
addrDR.Close();
phoneDR = phoneCom.ExecuteReader();
phoneDR.Read();
txtPhoneNumber.Text = phoneDR.GetString(0);
phoneDR.Close();
}
catch (DataException ex)
{
Response.Write(ex.ToString());
}
finally
{
addrCN.Close();
phoneDR = null;
addrDR = null;
addrCom = null;
addrCN = null;
}
}
```