ADO.NET – set of classes that allows .NET Framework programs to interact with different data sources.

* Main processes in a data app:

1.connecting to data

2.preaparing the app to receive the data

3.bringing data into the app

4.displaying data to the user

5. changing data

6.validating data

7.saving changes

1.bidirectional communication – app, data source; connection object of a data adapter

2.disconected data model – object that temporarily store data in the app while working with it(e.g. datasets).

3.executing queries/stored procedures

4.data-bound controls

5I/U/D.

6.

7.data adapter (commands)

.NET Framework data providers

* Connect to data source
* Retrieve data from data source
* Execute diff. commands on data source
* Connection, Command, DataReader, DataAdapter

.NET Framework data provider for Oracle

* System.Data.OracleClient namespace

\*SQLConnection

SQLConnection dbConn = new SQLConnection(

“Data Source = myServer\\SQLExpress;

Initial Catalog = myPrivateDb;

User ID = sa;

Password = weakpass;”);

//Integrated Security = true; SSPI

// vs (UID, PASSWORD).

dbConn.Open();

//code that works with the DB

DbConn.Close();

-PROPERTIES:

1.ConnectionString

2.ConnectionTimeout

\*SqlCommand

SqlCommand cmd = new SqlCommand(“SELECT COUNT(\*) FROM Pirates”, dbConn);

Int noPirates = (int)cmd.ExecuteScalar();

Cmd.CommandText = “INSERT Pirates (PID, Name) VALUES (1,’JS’)”;

Cmd.CommandType.Text;

Cmd.Connection = dbConn;

Cmd.ExecuteNonQuery();

//SqlParameter

ExecuteReader

While(dr.Read())

Console.WriteLine(“{0} \t {1}”, dr.GetInt32(0),dr.GetString(1)};

Dr.close().

\*SQLDataAdapter

\*DataSet

- properties: Tables(DataTableCollection), Relations (DataRelationCollection)

\* DataTable

-properties: Rows(DataRowCollection),Columns(DataColumnCollection), Constraints

-K=V;(key value pairs)

-Data Source / Server

//SQLConnection dbConn – instantiated, not open

DataSet ds = new DataSet();

SqlDataAdapter daPirates = new SqlDataAdapter(“SELECT \* FROM PIRATES”, dbConn);

SqlCommandBuilder cbp = new SqlCommandBuilder(daPirates);

daPirates.Fill(ds, “Pirates”);

//FILL automatically opens and closes the connection

DataRow dr = ds.Tables[“Pirates”].NewRow();

dr.[“PID”] = 1;

dr[“PName”] = “JS”;

ds.Tables[“Pirates”].Rows.Add(dr);

daPirates.Update(ds, “Pirates”);

//DataRow – property RowState with values(+,-,modif,unchanged)