Seminar01

Seminar 1 ADO .NET

The Data Cycle
1. Connect to Data	(Table Adapter, Sql Connection)
2. Prepare app to receive data	(DataSet, LINQ to SQL obj)
3. Fetch data	(Table Adapter)
4. Display data	
5. Edit data	
6. Validation	
7. Saving	(Table Adapter Manager)

        Data Models
                - typed / untyped dataset
                - LINQ to SQL
                - conceptual model to EntityDataModel

Data provider   <=>
-Connection   -Data Set
                -> using SystemData
-Command   -
DataRelationCollection
-Data Reader   -Data Table
-Data Adapter   -
DataTable
(Select cmd, Insert, Update, Delete)
        -DataRowCol
            -using System Data Sql Client   -
        DataCollumnCol
                -Oracle   -
            ConstraintCollection
                                                        C# app

Sql Connections

SqlConnection con = new SqlConnection("Data Source = DESKTOP ...; Inial Catalog = name_db;
Integrated Security=true"

```
try{
   con.Open();
}catch(Exception ex){...}
finally{
   con.Close();
}
```

Sql Command

```
SqlCommand cmd = new SqlCommand("select count(*) from TableABC", con);

int nr_rows = (int) cmd.ExecuteScalar();          // First row of the first collumn
cmd.CommandText = "insert into ABC (note) values ("abc")";
cmd.Connection = con
      .CommandTimeout =
      .CommandType = CommandType.Text.StoredProcedures

cmd.ExecuteNonQuery();


SqlDataReader dr = /*new SqlDataReader();*/
cmd.CommandText = "select * from ABC";
dr = cmd.ExecuteReader();

while(dr.Read()){
   Console.WriteLine(dr[0].ToString());   //or dr[name_col]
}
if(dr != null)
   dr.Close();


SqlDataAdapter da = new SqlDataAdapter("select * from ABC", con);
DataSet ds = new DataSet();
da.Fill(ds, "ABC");

DataTable dt = ds.Tables[0]
DataRow drow = dt.NewRow();
drow[0] = "xyz";
dt.Rows.AddRow(drow);

daUpdate(dt);
da.Update(ds, "ABC");




da.InsertCommand = new SqlCommand("insert into ABC (note) valies (@n)", con);
da. ---,,---.Parameters.Add("@n", SqlTypeVarchar) = "xyz";
try{
  con.Open();
  da.InsertCommand.ExecuteNonQuery();
}
catch...
finally{
  con.Close();
}
```

```
ds.Clear();
da.Fill(ds, "ABC");
ds.HasChanges();


SqlCommandBuilder cb = new Sql CommandBuilder(da);
cb.GetUpdateCommand();
da.Update(ds, "ABC");
```

DateSet
-used for all providers
-relational structure
-display/manages data if conDB is closed
-local-in-memory cache