ADO.NET

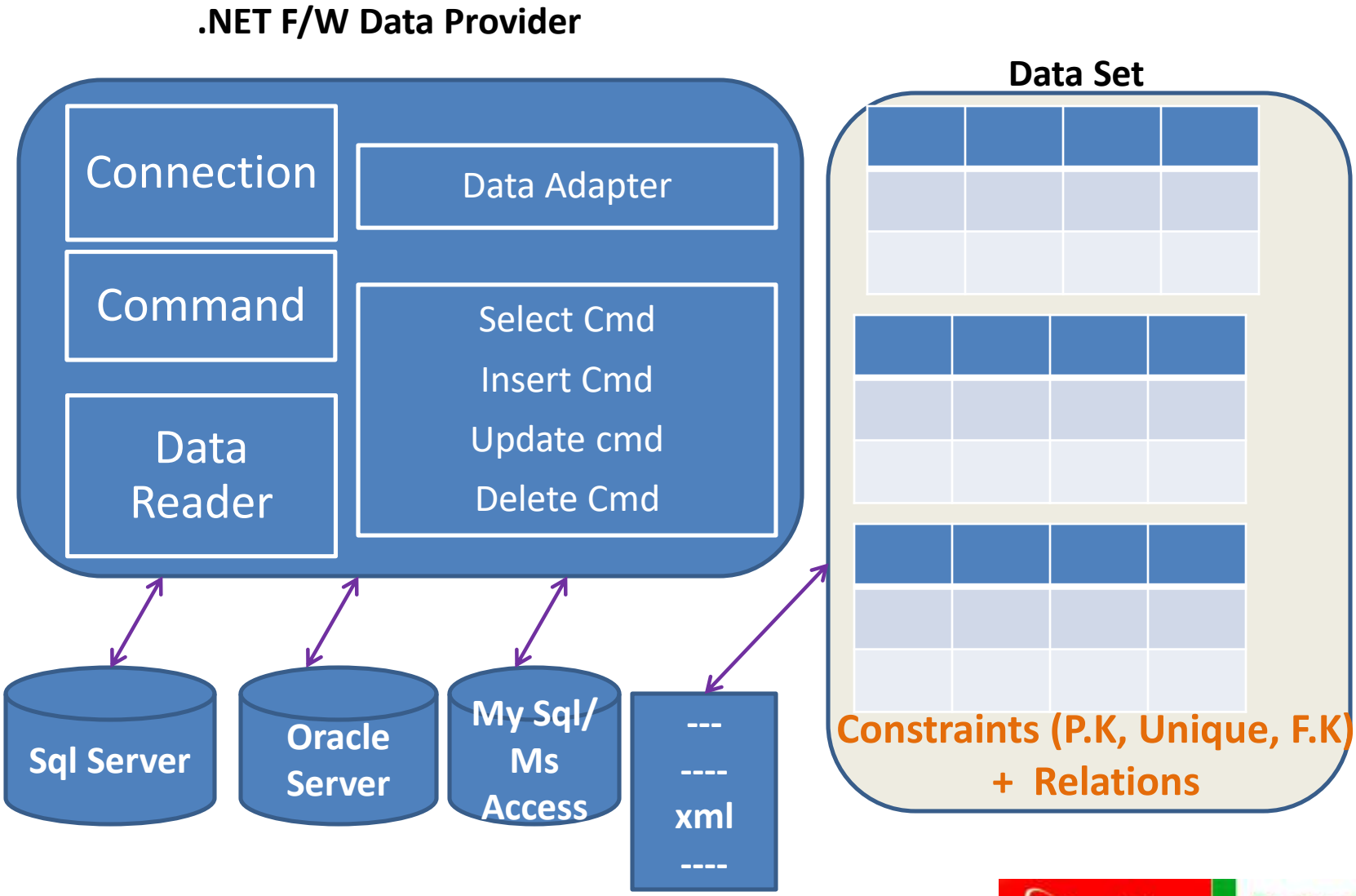
Lokanatha Palle

V 1.0

ADO.NET

- Is a technology present in Microsoft.NET
- Is used for communicating with different data sources like Sql server, Oracle, My Sql, DB2 etc..

ADO.NET Architecture



.NET F/w Data Providers

ADO.NET Supports following .net f/w data providers.

.NET F/W DP for Sql server

- used for connecting to sql server

.NET F/W DP for Oracle

- Used for connecting to Oracle Server

.NET F/W DP for OLEDB

- Used for connecting to oledb supported sources like MS Excel MS Access etc..

.NET F/W DP for ODBC

Entity Client

.NET F/W DP for Sql server Compact

.NET F/W Data Provider Namespaces Part1

{} system.data.sqlclient

- SqlConnection
- SqlCommand
- SqlDataReader
- SqlDataAdapter

{} system.data.oledb

- OledbConnection
- OledbCommand
- OledbDataReader
- OledbDataAdapter

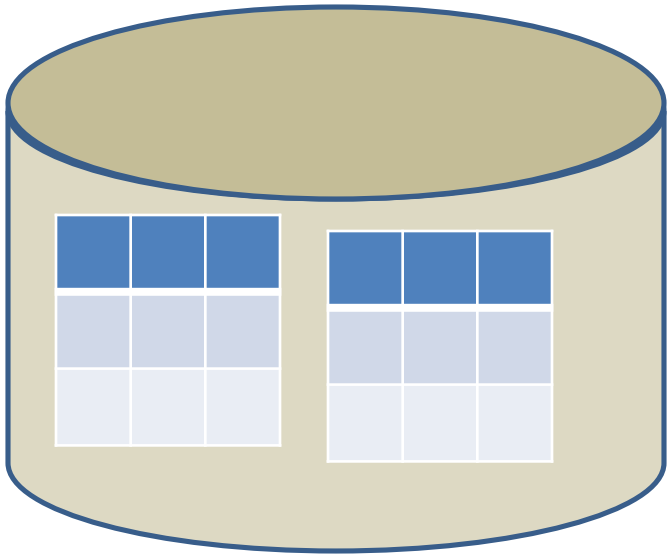
{} system.data.odbc

- OdbcConnection
- OdbcCommand
- OdbcDataReader
- OdbcDataAdapter

Connection Methods

- Connection class supports Open and Close Methods.

```
SqlConnection cn = new SqlConnection(" connection string");
"data sorce = name of sql server; initial catalog = db name; integrated security = sspi"
user id=....; password=.....'
cn.Open();
cn.Close();
```



Sql server

Command Methods

- For executing sql commands against data store we have 3 methods.
    1. **ExecuteNonQuery**
    2. **ExecuteScalar**
    3. **ExecuteReader**

ExecuteNonQuery: usually used for executing dml statements like insert/update/delete.

ExecuteScalar: used for executing **select** statement which **returns** a single value as result.

ExecuteReader: used for executing **select** statement which **returns** more than one value as result.

Identify right command method

```
select * from patient;
select * from patient where pid=4;
select fname+lname as 'fullname' from patient where pid=5;
select fname+lname as 'fullname' from patient ;
insert into patient values (11,'hucha','venkat',55, 'o+ve')
delete patient where pid=7;
update patient set fname='karia' where pid=3;
```

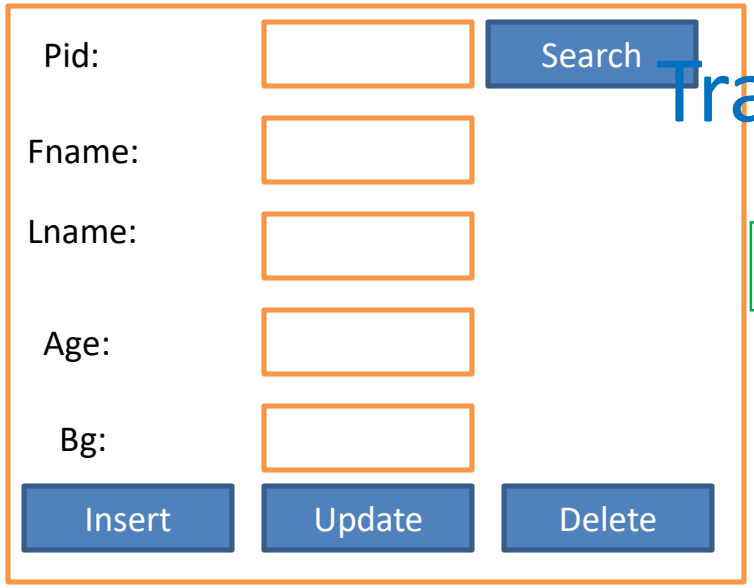
ExecuteScalar

ExecuteNonQuery

ExecuteReader

pid	fname	lname	age	bg
1	madhava	reddy	45	o+ve
2	abhinav	bandra	45	o-ve
4	hari	kiran	60	b-ve
3	madhava	kiran	52	o+ve
5	veena	kumari	42	NULL
6	k_iran	kumar	39	b-ve
2	abhinav	bandra	45	o-ve
7	mahes%h	nambootri	36	b+ve
8	rahul	kumar	46	b-ve
9	bharat	kumar	56	b-ve

PatientDetails.aspx

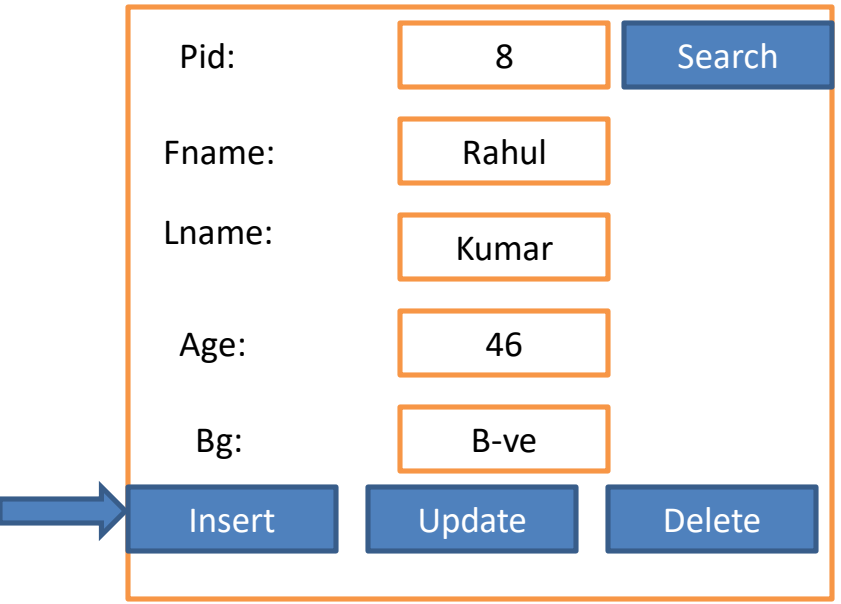


Trainer

Req: insert data into patient table

pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve

Actual patient table



pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve
8	Rahul	Kumar	46	B-ve

patient table after insertion

ado standard practices – part1

create objects which are implementing **IDisposable** interface in side the using block.

```
using(ClassName obj=new ClassName())
{


}
```

ado standard practices – part2

- use try catch & finally blocks appropriately.
- place suspected code like connection opening and command execution code in try block.
- handle SqlException in catch block
- close connection in finally block.

```
try
{
    -------
   cn.Open();
   ……..
   cmd.ExecuteNonquery(); // or any other method
    …..
}
catch( SqlException ex)
{
}
finally
{
    if(cn.State==ConnectionState.Open)
      cn.Close();
}
```

ado standard practices – part 3

- always store connection string inside the configuration file.

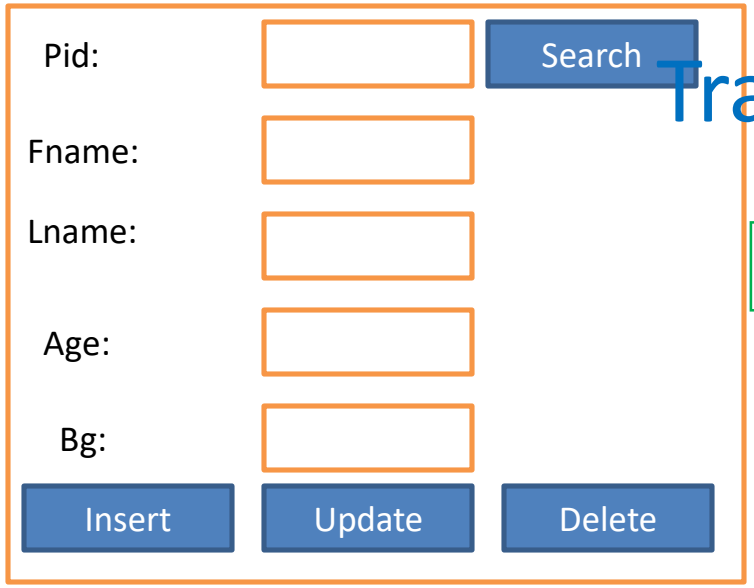
Web.config*

```
<connectionStrings>
  <add name="connection_name" connectionString="connectio string goes here"/>
</connectionStrings>
```

Code for reading the connection string data

```
string scn=ConfigurationManager.ConnectionStrings["connection_name"].ConnectionString;
```

PatientDetails.aspx

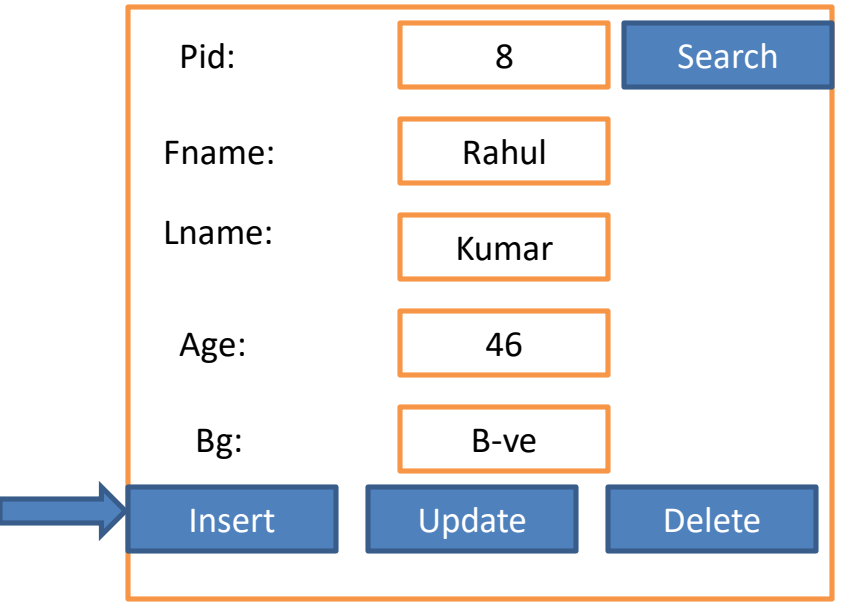


Trainer

Req: insert data into patient table

pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve

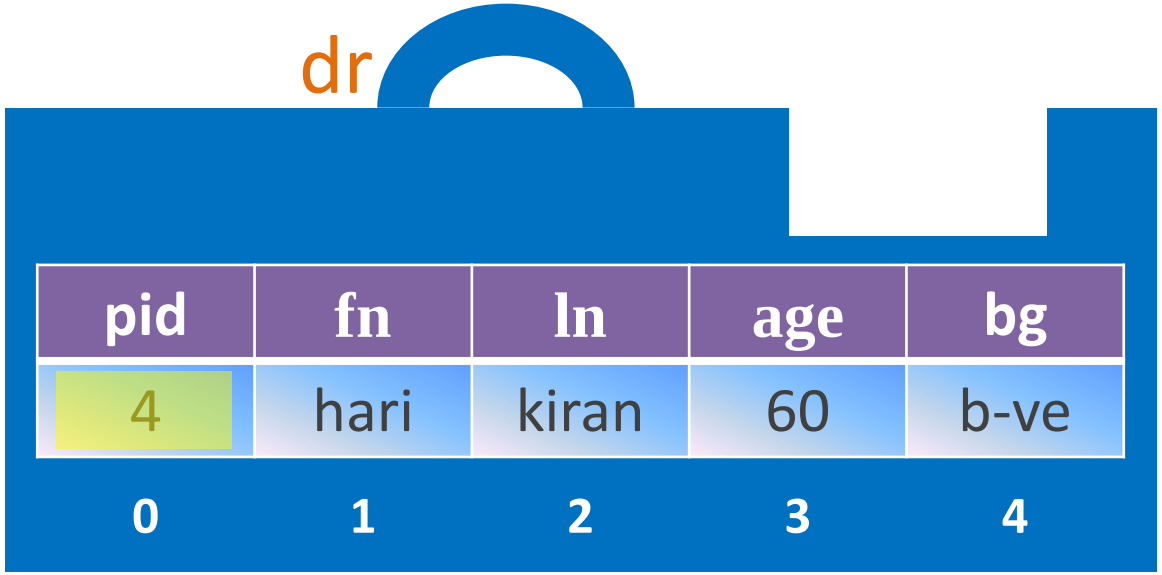
Actual patient table



pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve
8	Rahul	Kumar	46	B-ve

patient table after insertion

Reading Data from Data Reader




```
while (dr.Read())
{
    object o1 = dr[0];
    object o2 = dr[1];
    object o3 = dr[2];
    object o4 = dr[3];
    object o5 = dr[4];

}
dr.Close();
```

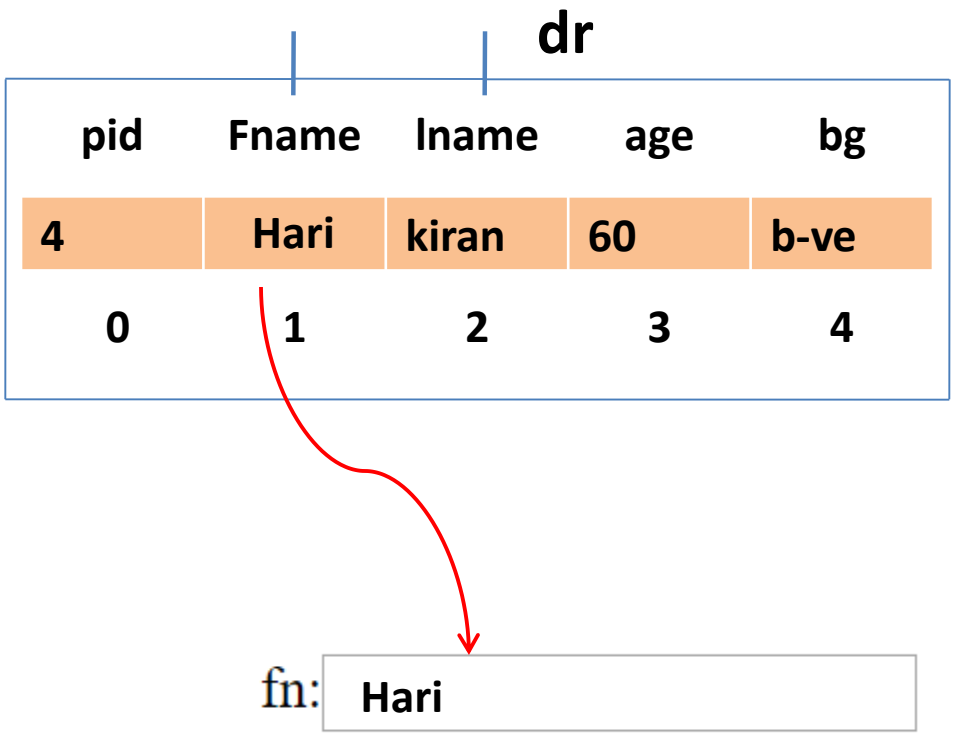
```
while (dr.Read())
{
    object o1 = dr["pid"];
    object o2 = dr["fn"];
    object o3 = dr["ln"];
    object o4 = dr["age"];
    object o5 = dr["bg"];

}
dr.Close();
```

How to deal with DataReader

Patient

P_Id	fname	LName	Age	Bg
1	Madhava	Reddy	45	O+ve
2	Abhinav	bandra	45	O-ve
3	Madhava	kiran	52	o+ve
4	Hari	kiran	60	b-ve
5	veena	kumari	42	NULL
6	k_iran	kumar	39	b-ve
2	Abhinav	bandra	45	O-ve
7	Mahes%h	Nambotri	36	b+ve
8	Rahul	kumar	46	b-ve
9	Bharat	kumar	56	b-ve

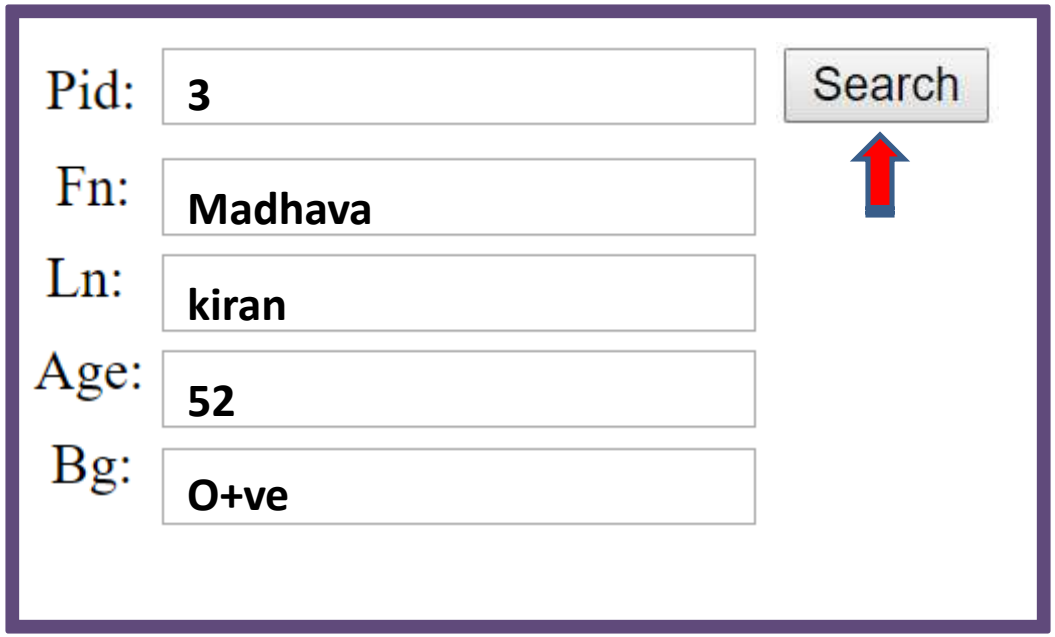



```
while (dr.Read())
{
  object o1 = dr[0];
  object o2 = dr["pid"];
  tbFname.Text = Convert.ToString(dr["fname"]);
 }
```

It is not recommended

data reader sample 1

Patient.aspx

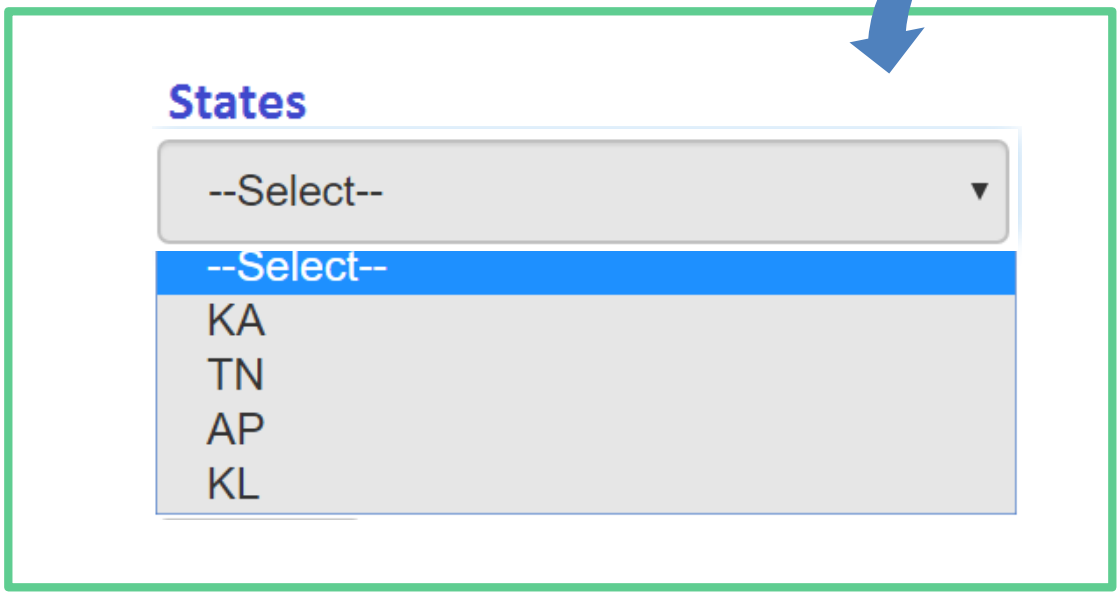


P_Id	fname	LName	Age	Bg
1	Madhava	Reddy	45	O+ve
2	Abhinav	bandra	45	O-ve
3	Madhava	kiran	52	o+ve
4	Hari	kiran	60	b-ve
5	veena	kumari	42	NULL
6	k_iran	kumar	39	b-ve
2	Abhinav	bandra	45	O-ve
7	Mahes%h	Nambotri	36	b+ve
8	Rahul	kumar	46	b-ve
9	Bharat	kumar	56	b-ve

Patient

data reader sample - 2

States.aspx

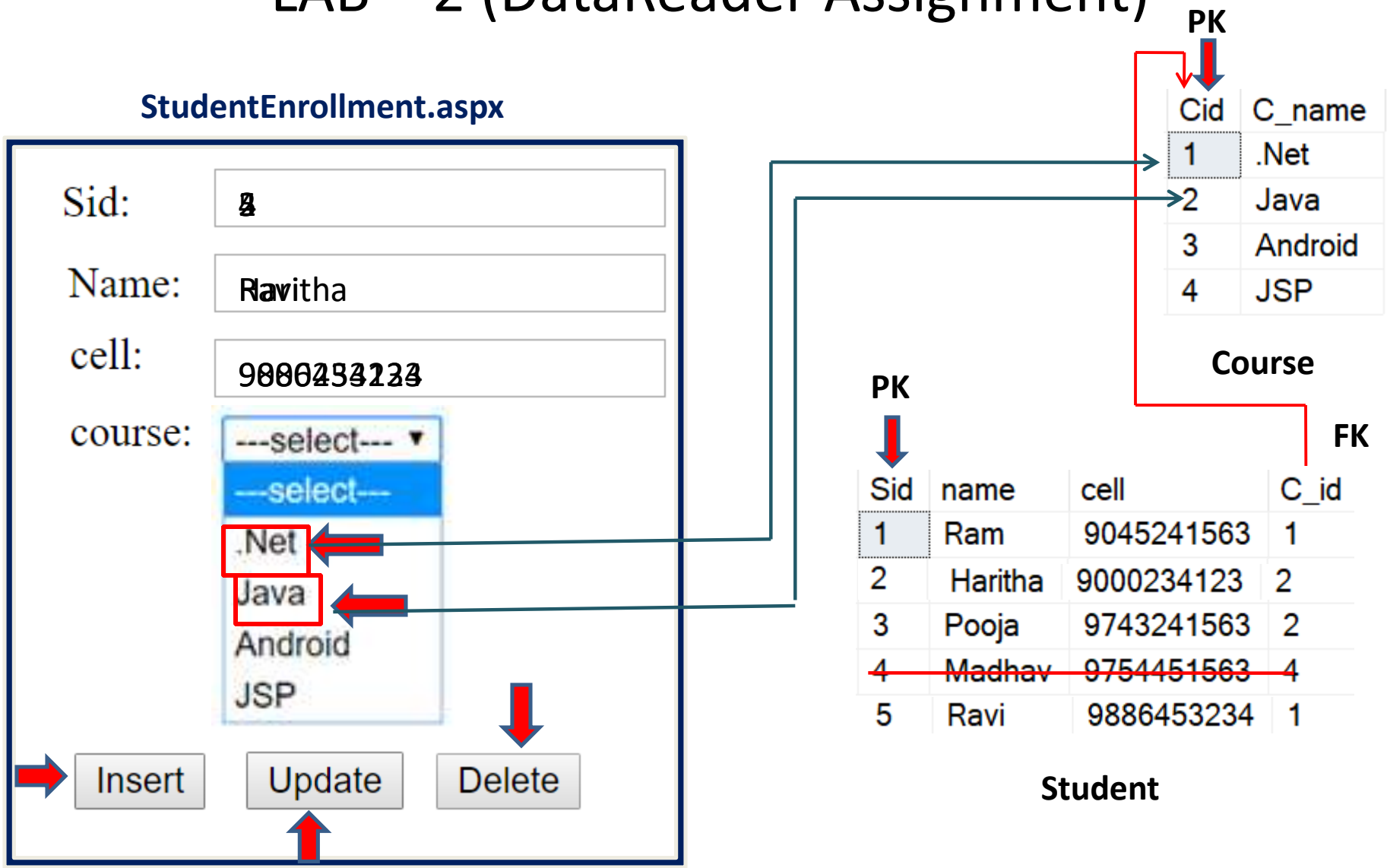


PK

Sid	Sname
1	KA
2	TN
3	AP
4	KL

States

LAB – 2 (DataReader Assignment)



LAB – 3 (DataReader Assignment)

1.Requirement:

Create two tables in Database

Int	Vc(40)	Int
sid	name	courseid
1	Ravi	2
2	Hari	1
3	Veena	2
4	Kiran	1

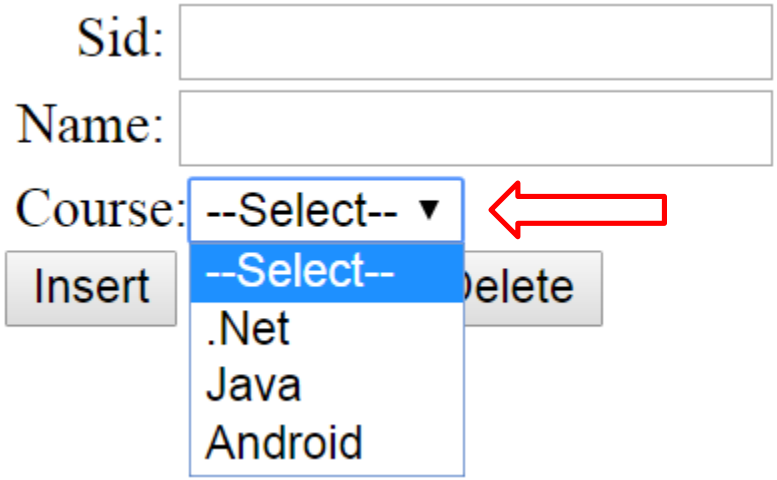
Student

Int	Vc(40)
id	coursename
1	.Net
2	Java
3	Android

Course

2.Requirement:

Create StudentDetails.aspx



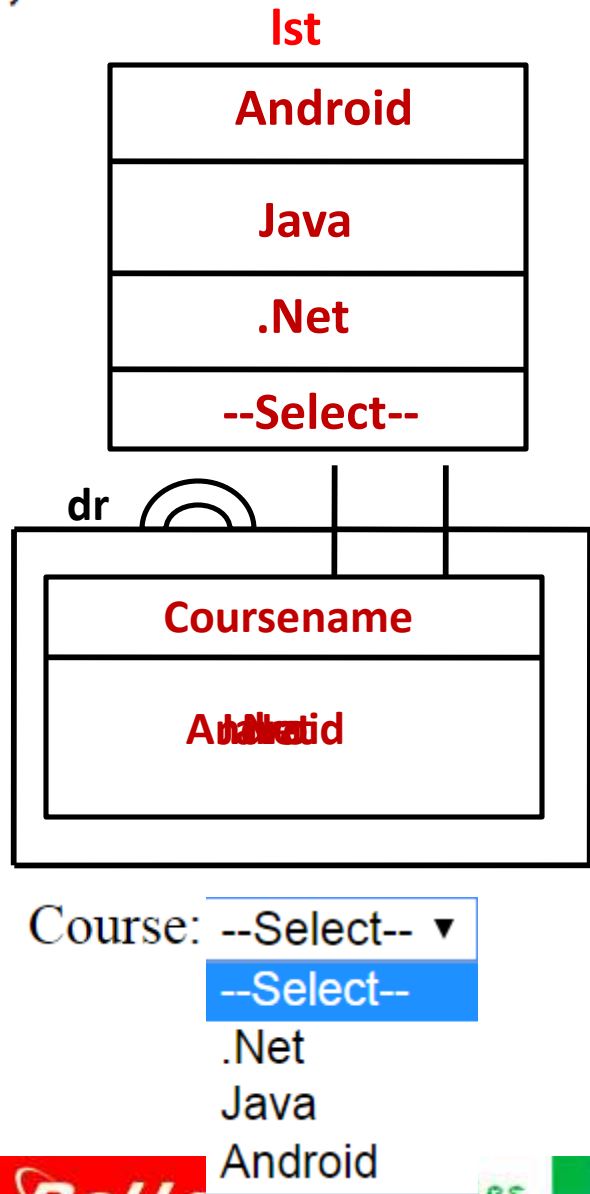
3.Requirement:

Select coursename from course table and fetch data into course dropdownlist.

Solution for binding data into DropDownList.

F11

```
using (SqlConnection cn = new SqlConnection(scn))
{
   try
   {
       cn.Open();
      string qry = "select coursename from course";
     using (SqlCommand cmd = new SqlCommand(qry, cn))
     {
        List<string> lst = new List<string>();
        lst.Add("--Select--");
       SqlDataReader dr = cmd.ExecuteReader();
       while (dr.Read())
       {
         lst.Add(Convert.ToString(dr["coursename"]));
       }
      ddlcourse.DataSource = lst;
      ddlcourse.DataBind();
      dr.Close();
     }
   }
}
```



Requirement 4:

How to Insert Courseid into student table..

By using sub queries..

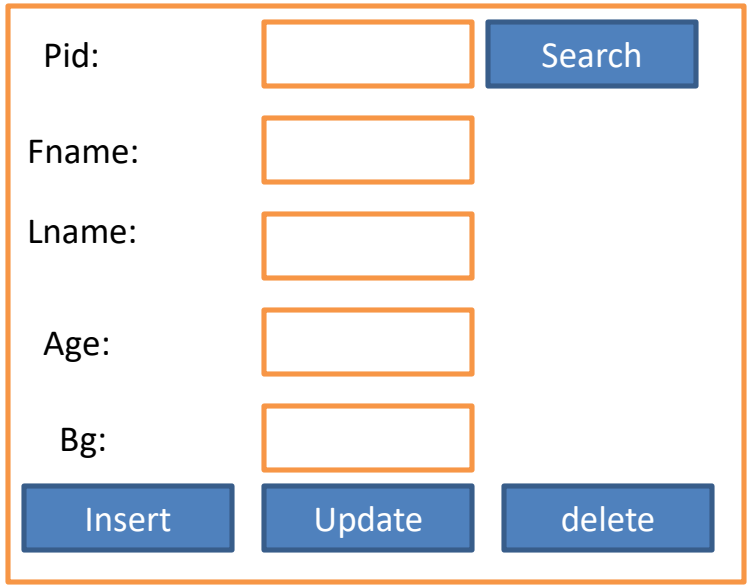
```
insert into student values(5,'Hema',(-----------------------))
```

Write a Subquery to get the courseid based on coursename.

Calling SP from ADO.NET

students must watch videos before starting the stored procedure calling concepts

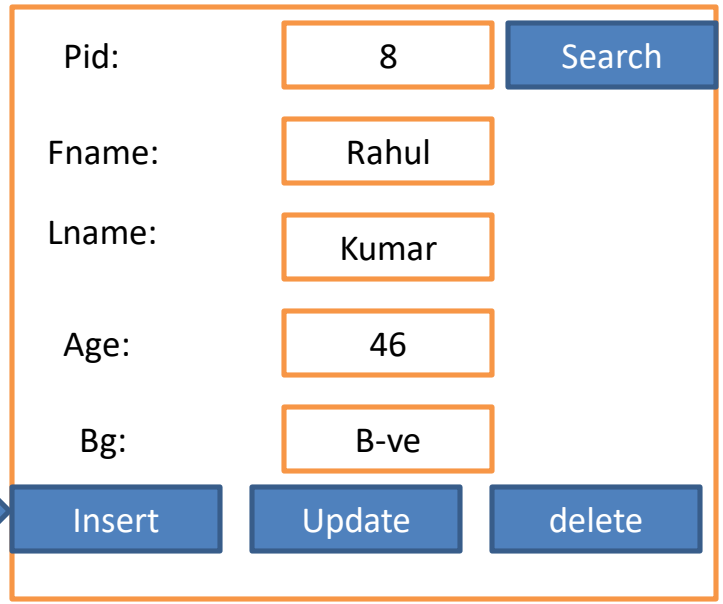
PatientDetails.aspx



Req: insert data into patient table

pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve

patient



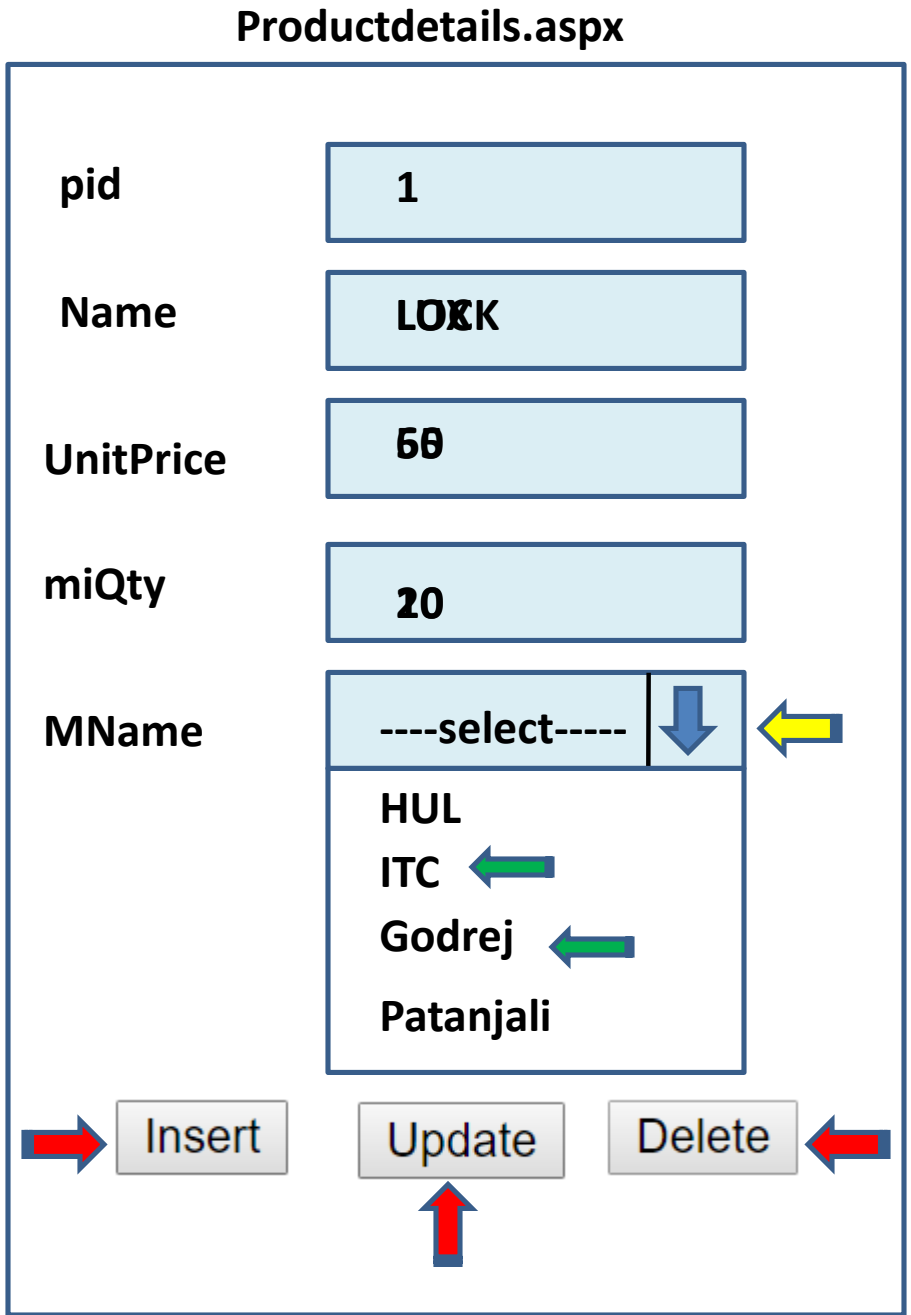
pid	fn	ln	age	bg
1	Madhava	Reddy	45	O+ve
4	Hari	Kiran	60	B-ve
3	Madhava	Kiran	52	O+ve
5	Veena	Kumari	42	null
6	K iran	Kumar	39	B-ve
2	Abhinav	Bandra	45	O-ve
7	Mahes%h	Nambotri	36	B+ve
8	Rahul	Kumar	46	B-ve

patient table after insertion

inserting data into Patienttable

```
create proc Inspat
(
@Pid int,
@Fname varchar(40),
@Lname varchar(40),
@Age int,
@Bg varchar(40),
)
as
begin
insert into Patienttable values(@Pid ,@Fname,@Lname,@Age,@Bg)
end
```

Lab -5



Manufacturer

PK

Mid	MName
1	HUL
2	ITC
3	Godrej
4	Patanjali

Product

PK FK

pid	Name	Unitprice	miqty	m_id
1	LOCK	65	10	3

authentication & authorization

- authentication is the process of finding users identity (finding user is a genuine user or not).
- authorization is about finding whether an authenticated user is allowed to access a specific page or not.
- in internet based web applications we normally use forms based authentication & role based authorization.

steps for authentication & authorization

1. enable authentication in root web.config file
2. implement authentication code in login page
3. add the user role to generic principal object
4. add the generic principal object to current user prop
5. enable authorization in respective web.config file

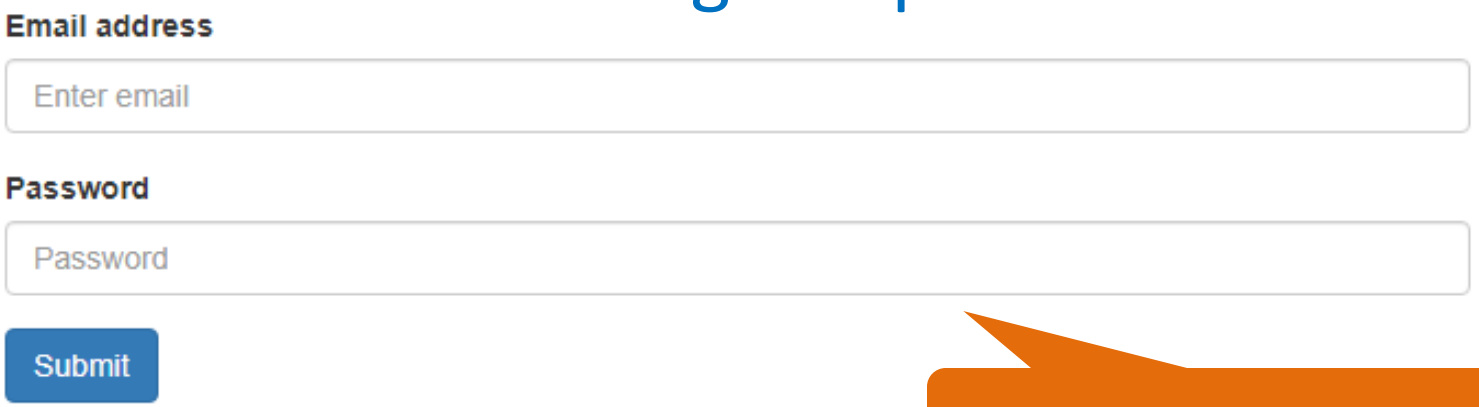
step 1

sino	name	email	cell	role	password
1	lokanatha	lokanatha.palle@techpalle.com	9852488228	management	lokanatha123
2	adira	adira@g.com	9809809807	teacher	adira123
3	swayam	swayam@g.com	9182737645	student	sw123

userroles

create table

login.aspx

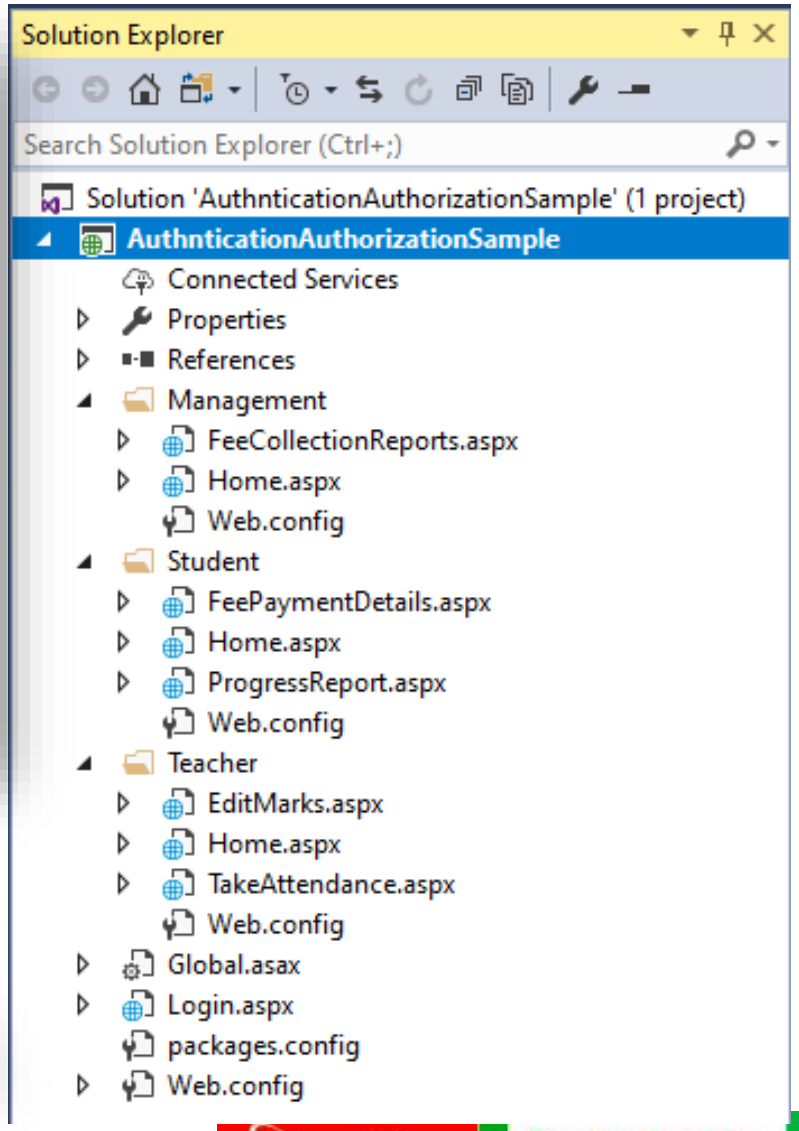


create page

step 2

~/web.config

```
<?xml version="1.0"?>
<configuration>
  <system.web>
    <authentication mode="Forms">
  <forms defaultUrl="login.aspx"
loginUrl="login.aspx" protection="All"
path="/"></forms>
 </authentication> </system.web>
</configuration>
```



step 3

in login button click

```
protected void btnLogin_Click(object sender, EventArgs e)
{
    SqlConnection cn = new SqlConnection(@"data source=DESKTOP-QAVVTBQ;initial catalog
                                          =mvcloksamples;integrated security=sspi");
    string qry = string.Format("select role from userroles where email='{0}'" +
        " and password='{1}'", email.Value, password.Value);
    SqlCommand cmd = new SqlCommand(qry, cn);
    cn.Open();
    string role=Convert.ToString(cmd.ExecuteScalar());
    cn.Close();
    if(role!=string.Empty||role!=null)
    {
```

```
        FormsAuthenticationTicket ticket = new FormsAuthenticationTicket(1, "palle", DateTime.Now,
                    DateTime.Now.AddMinutes(50), true, role, FormsAuthentication.FormsCookiePath);
        string hashCookies = FormsAuthentication.Encrypt(ticket);
        HttpCookie cookie = new HttpCookie(FormsAuthentication.FormsCookieName, hashCookies) {
            Expires = ticket.Expiration };
        Response.Cookies.Add(cookie);
        if(role.ToLower()=="management")
            Response.Redirect("~/Management/Home.aspx");
        else if(role.ToLower() == "teacher")
            Response.Redirect("Teacher/Home.aspx");
        else if(role.ToLower() == "student")
            Response.Redirect("~/Student/Home.aspx");
    }
    else {/*display error*/}
}
```

in login button click

step 4

in global.asax

```
protected void Application_AuthenticateRequest(object sender, EventArgs e)
{
    if (HttpContext.Current.User != null)
    {
        if (HttpContext.Current.User.Identity.IsAuthenticated)
        {
            if (HttpContext.Current.User.Identity is FormsIdentity)
            {
                FormsIdentity id =
                    (FormsIdentity)HttpContext.Current.User.Identity;
                FormsAuthenticationTicket ticket = id.Ticket;
                string userData = ticket.UserData;
                string[] roles = userData.Split(',');
                HttpContext.Current.User =
                    new System.Security.Principal.GenericPrincipal(id, roles);
            }
        }
    }
}
```

step 5

~/Management/web.config

```
<authorization>
        <allow roles="admin"/>
        <deny users="*"/>
      </authorization>
```

~/Students/web.config

```
<authorization>
        <allow roles="parent"/>
        <deny users="*"/>
      </authorization>
```