Chapter-1

Introduction

B.Kannababu

(Sathyatechnologies)

Q)what is WebBrowser?

WebBrowser is a software which is used to access the content from webserver(server)

Ex:- internet Explorer,Googlechrome,Mozilla Firefox ,opera,Safari

Q)what is webserver?

webserver is a software which is used to provide responses or services to different devices likeBrowser,mobile etc...

Ex:- IIS(internet Information services),Apache

Q)what is the advantage of placing the files on webserver?

if we place any files on webserver then enduser can access those files via Browser and internet

Q)what file consists of ?

Data

Ex:- .txt,.doc, .html, .xml, .aspx, .pdf,.jpg, .cshtml ,.js ,.css

Data may be text,image,audio,video etc..

Q) What is Protocol?

A protocol is a set of rules and guidelines for communicating data.

There are various types of protocols that support a major and compassionate role in communicating with different devices across the network. These are:

1. Transmission Control Protocol (TCP)
2. Internet Protocol (IP)
3. User Datagram Protocol (UDP)
4. Post office Protocol (POP)
5. Simple mail transport Protocol (SMTP)
6. File Transfer Protocol (FTP)
7. Hyper Text Transfer Protocol (HTTP)
8. Hyper Text Transfer Protocol Secure (HTTPS)

Q)what is Http?

Http is Hyper Text Transfer Protocol

Http is used to create the communication between client and Server

Http is responsable to access the Data from server

Http is responsable to submit,update,delete the Data from server

Http internally consists of 4 predefined Methods

1. HttpGet it is used to get the data from server
2. HttpPost it is used to submit the data to server
3. HttpPut it is used to update the data to server
4. HttpDelete it is used to delete the data to server

Q)How to access the data from webserver?

by using url

url:- uniform resource locator(resource is file)

http://servername:portno/filename

http://localhost:1086/home.html

http://localhost:1086/ab.jpg

note:-

we can deploy multiple webapplications on a single webserver

Every webapplication i.e avaialable under webserver can be identified by using portno

Q) How to Deploy the webapplication on webserver?

In order to deploy the webapplication on webserver we need to follow 2 steps:-

1. we need to purchase space on webserver (Godaddy)
2. we need to purchase Domain name

Ex:- www.facebook.com

www.gmail.com

If the webapplication was developed in ASP.net or ASP.net MVC then we need to deploy the application under IIS

Q)what is WebApplication?

A **web application** is a computer program that utilizes **web** browsers and **web**technology to perform tasks over the Internet.

A **web application** is a software **application** that runs on a remote server.

Ex:- irctc,swiggy,icici.com,zomato.com etc….

Chapter-2

VisualStudioEditor, .net Framework

B.Kannababu
(Sathyatechnologies)

Microsoft VisualStudio Editor:- it is a Development Environment which is used to develop Different Types of Applications by using Different .net supportable Languages and Technologies like C#.net,vb.net,ASP.net,ASP.net MVC etc...

.net Framework:- .net Framework is a software framework which was developed by Microsoft for building,deploying and running different Types of applications

.net F/W provides some set of predefined classes and Methods under Base Class Libraries and these Libraries are available when we install .net F/W

.net F/W is responsible to run the appn

Every software will have versions

versions are of 4 Types

Major version
Minor Version
Build Version
Revised Version
0.0.0.0

.net 1.0	2000	MicrosoftVisualStudio-2000
.net 1.1	2003	MicrosoftVisualStudio -2003
.net 2.0	2005	MicrosoftVisualStudio -2005
.net 3.5	2008	MicrosoftVisualStudio -2008
.net 4.0	2010	MicrosoftVisualStudio -2010
.net 4.5	2012	MicrosoftVisualStudio -2012
.net 4.5.1	2013	MicrosoftVisualStudio -2013
.net 4.6	2015	MicrosoftVisualStudio -2015
.net 4.7	2017	MicrosoftVisualStudio -2017

steps to open Visualstudio:-

1. goto---->start--->Allprograms--->Visualstudio 2013
   -->Visualstudio 2013
2. goto-->start--->run--->devenv--->ok
   devenv:- development environment
   Different Types of application that we can develop by using VisualStudio Editor are

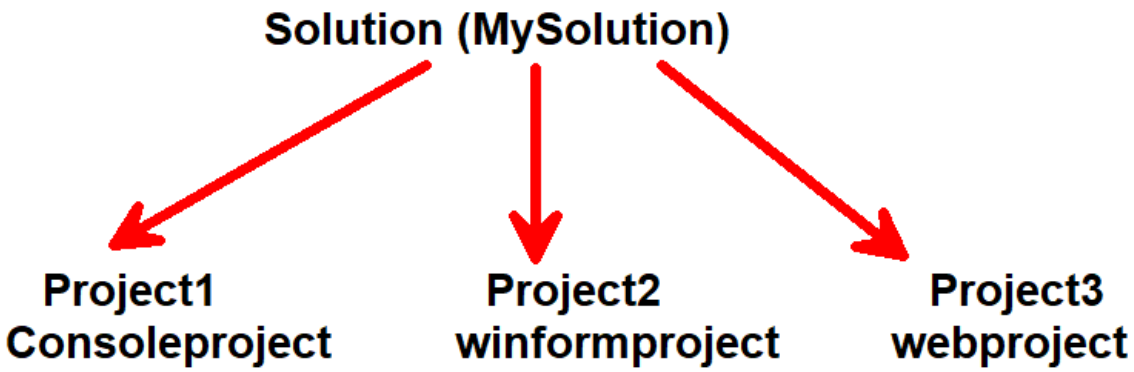
1. Console Applications **(C#.net)**
2. Windows forms Applications **(C#.net)**
3. WebApplications **(ASP.net,ASP.net MVC)**
4. WPF Applications **(C#.net)**
5. Service Oriented Applications **(WCF,WEBAPI)**

- whenever we compile Console appn,Windows Forms appn,WPF appn then the compiler will generate .exe file
- whenever we compile Webappn,WCF,WebApi,ClassLibraries then the compiler will generate .dll file

.dll---> Dynamic Link Library

note:-

- if we want to run Webappn,WCF,WebApi appn then webserver is manadatory
- whenever we develop a project in Visualstudio Editor then a solution is created
- solution is collection of projects
- project is collection of files



Creating a Solution in VisualStudio Editor:-

1. goto--> D Drive and create a folder with name MyDemo
2. goto--->start-->run--->devenv-->ok
3. select Newproject--->language=VisualC#
   .net FrameWork 4.5.1
   Template=ConsoleApplication

name=ConsoleProject
Location=D:/MyDemo
Solutionname=MySolution

4. goto-->view--->solutionExplorer
rc on consoleproject--->Add class-->name=First.cs

```
using System;
class First
{
   static void Main()
   {
      Console.WriteLine("First Prgram");
   }
}
```

goto-->solutionExplorer--->rc on consoleproject--->
Addclass--->name=Second.cs

```
using System;
class Second
{
   static void Main()
   {
      Console.WriteLine("Second Program");
   }
}
```

lly add Third program
note:- if multiple programs exist in single project
inorder to execute a specific program
goto-->project on menubar
Consoleproject properties--->
startupobject=First press ctrl+F5

5. creating a newproject within MySolution
goto-->solutionExplorer-->rc on MySolution---->Add new project-->select WindowsForms Application

Name=WindowsFormsProject --->ok

6. lly goto--->solutionExplorer ---->rc on MySolution ---->
   AddNewproject-->select ASP.net WebApplication
   ---->name=Webproject--->ok
   select Empty ----> WebForms--->ok
   rc on webproject--->Addnewitem-->select Webform

Chapter-3

Different Types of windows in Visual Studio Editor

B.Kannababu
(Sathyatechnologies)

goto-->start--->run--->devenv--->ok

File--->Newproject--->select Language=Visual C#

Template=ASp.net WebApplication

name=mywebapp--->Location=D:/Demo

solution=MySolution---->ok

goto--->solutionExplorer--->rc on project---->Addnewitem

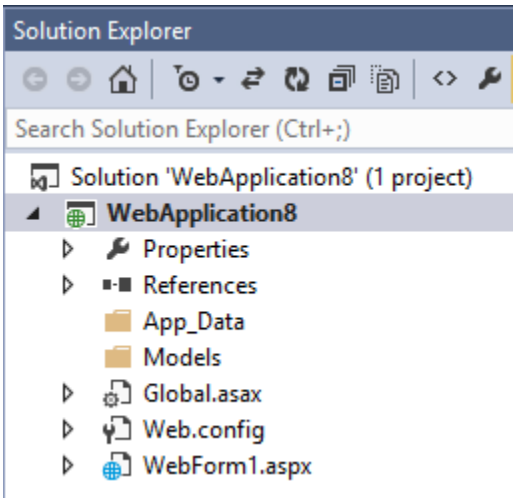
--->select webform-->name=page1.aspx--->Add

Different Windows:-

1. SolutionExplorer:-This window consists of the list of projects that are available within the solution.

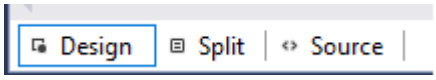
solution is collection of projects

view--→solutionExplorer



2. Design Window:-This window is used to Display the design of the webpage

Page1.aspx[Design]



3. source window:-This window consists of Designing code

Page1.aspx[Source]

Html,Javascript,CSS,ASP.net code

```
<html>
<head>
    javascript,css,jquery
</head>
```

```
<body>
    Html,ASp.net
</body>  </html>
```

4. split window:-This window is used to display both Design window and source window at a time

Page1.aspx[Split]

5. Business Logic Window or Code window:-This window is used to write Business logic(C#.net code)

goto-->page1.aspx[Design]

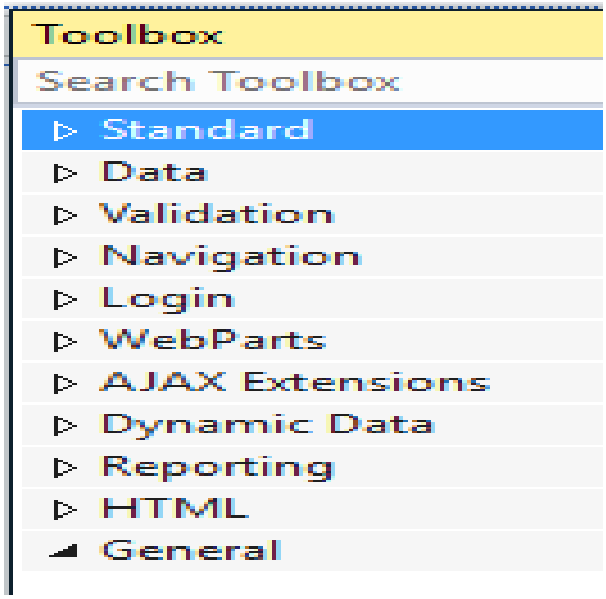
double click on Design window

Page1.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace WebApplication8
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
    }
}
```


6. ToolBox:-it will provide some predefined controls to Design the webpage

goto-->Design window--->View--->ToolBox

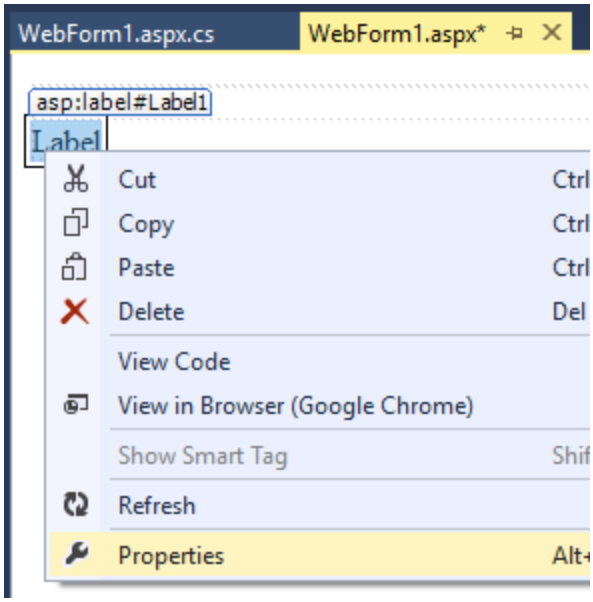


7. Properties Window:-This window will provide the List of Properties

Properties are used to set the value and get the value for controls

goto--> page1.aspx[Design] ---> rc on any control

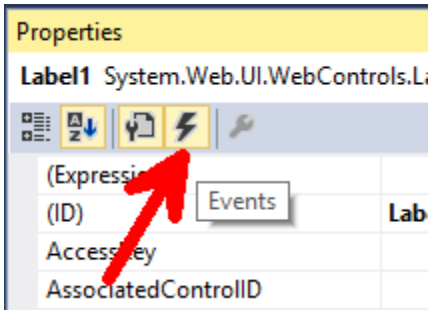
properties



8. Events Window:- Events are used to invoke Methods

goto--> page1.aspx[Design] ---> rc on any control

properties ---->Events



9. ErrorList:- This window is used to display List of Errors

10. ServerExplorer:-This window is used to interact with SqlServer

Q)what is Partial class?

Partial class means defyning multiple classes with same name

```
partial class A                    partial class A
{                                  {
   public void Show()                 public void Display()
   {                                  {

   }                                  }
}                                  }
```

csc A.cs

At compile time both the partial classes will consider as single class

```
partial class Page1:Page           partial class Page1
{                                  {
  UI code                                BL code
}                                  }
```

K A N N A B A B U

Chapter-4

Controls in ASP.net

B.Kannababu
(Sathyatechnologies)

Controls in ASp.net:-

Controls are small building blocks of the graphical user interface, which include text boxes, buttons, check boxes, list boxes, labels, and numerous other tools. Using these tools, the users can enter data, make selections and indicate their preferences.

Controls are also used for structural jobs, like validation, data access, security, creating master pages, and data manipulation.

All Controls in ASP.net are available under ToolBox

Control:-Every Control is a predefined class

controls are used to Design the webpage

Every control will have 3 things in common

1. Properties
2. Events
3. Methods

- Properties are used to set the value and get the value from controls
- Events are used to invoke Methods
- Method is used to perform a specific operation
- All the Controls in ASp.net are available under System.Web.UI.WebControls namespace
- The super class for all the controls is Control class

General syntax for any asp.net control:-

```
<asp:controlname  id="someid" runat="server">
</asp:controlname>
```

note:-

- All asp.net controls are serverside controls
- These controls will execute on webserver
- All controls are available under ToolBox

Label:- it is used to Display the Text

it is similar like Html span tag

```
<asp:Label id="label1" runat="server">
</asp:Label>
```

```
class Label
{
  pubic string Text{set;get;}
}
Label label1=new Label();
label1.Text="sathya";
```

Every control is a predefined class

whenever we drag and drop the control from Toolbox then object is created for control class

Common Properties for all the controls:-

1. id :- id is the name given for the control

The purpose of id is to identify the control

2.Text:-This property is used to display the Text within the control

TextBox:-TextBox is Dataentry control which is used to accept the data from the user

TextBoxes are of 3 Types:-

1. SingleLine TextBox

2. Password TextBox

3. MultiLine TextBox

SingleLine TextBox:- it is used to accept singleline of Text from the user

Password TextBox:- it is used to accept singleline of Text from the user and display the text in hidden format

MultiLine TextBox:- it is used to accept multiple lines of Text from the user

Properties:-

1. id
2. Text
3. TextMode=SingleLine
   Password
   MultiLine
4. ReadOnly=True/False:-if true we cannot edit the textbox

Examples:-

By Defualt TextBox will accept the data in the form of string

1. goto-->View--->SolutionExplorer--->rc on project--->

   Add newitem---> select Webform --> name=page1.aspx --->Add

2. goto-->Page1.aspx[Design]

   goto--->ToolBox and Design the webpage

Q)what is the purpose of id?

The purpose of controlid is to identify the control

syn to store the value in control:-

controlid.Text=value;

syn to read the value from control:-

string s=controlid.Text;

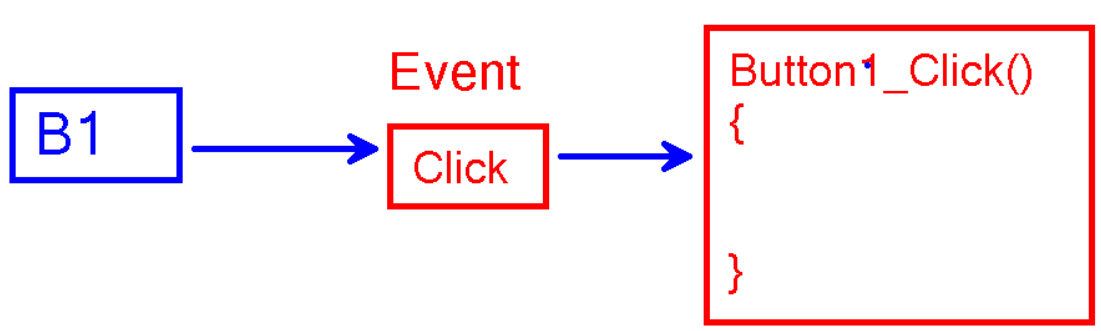
double click on Button and write the code:-

```
protected void Button1_Click()
{
        string fname = TextBox1.Text;
        string lname = TextBox2.Text;
        string fullname = fname + lname;
```

```
        Label1.Text = fullname;
}
```

SATHYATECHNOLOGIES

observation:- user will enter firstname in textbox1 and lastname in textbox2 and click on Button then an Event will fire and the event will invoke Button1_Click() method and the code that was written inside the Method will gets executed

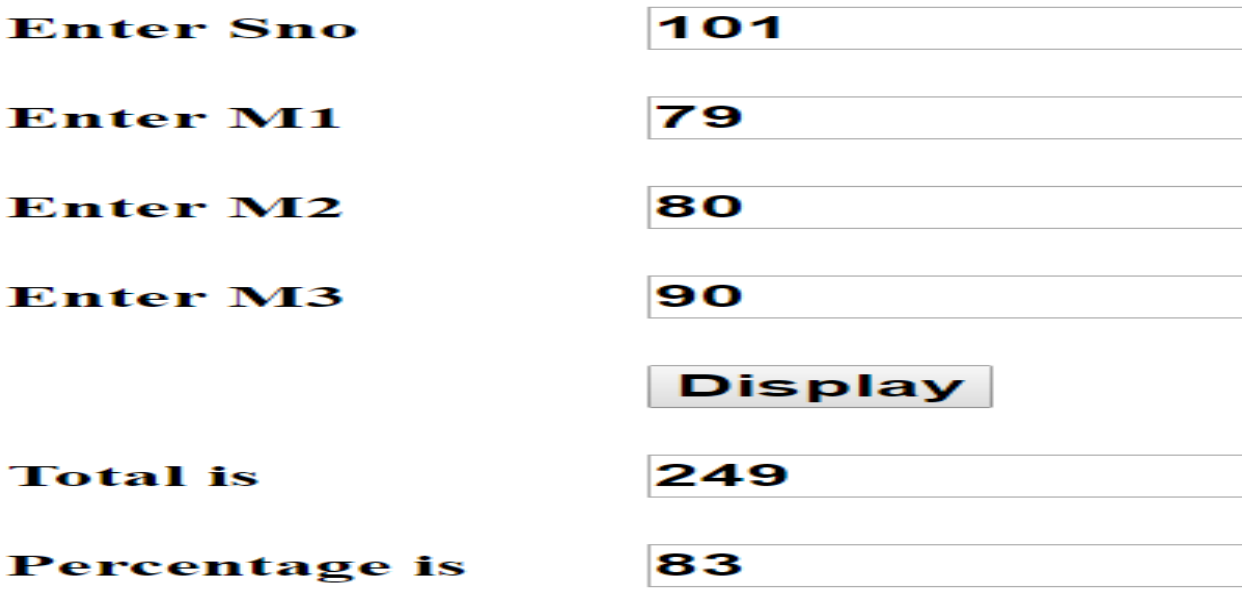


KANNABABU


```
protected void Button1_Click(object sender, EventArgs e)
    {
        int a = int.Parse(TextBox1.Text);
        int b = int.Parse(TextBox2.Text);
        int c = a + b;
        Label1.Text = c.ToString();      }
    protected void Button2_Click(object sender, EventArgs e)
    {
        int a = int.Parse(TextBox1.Text);
```

```
int b = int.Parse(TextBox2.Text);
int c = a - b;
Label1.Text = c.ToString();
```

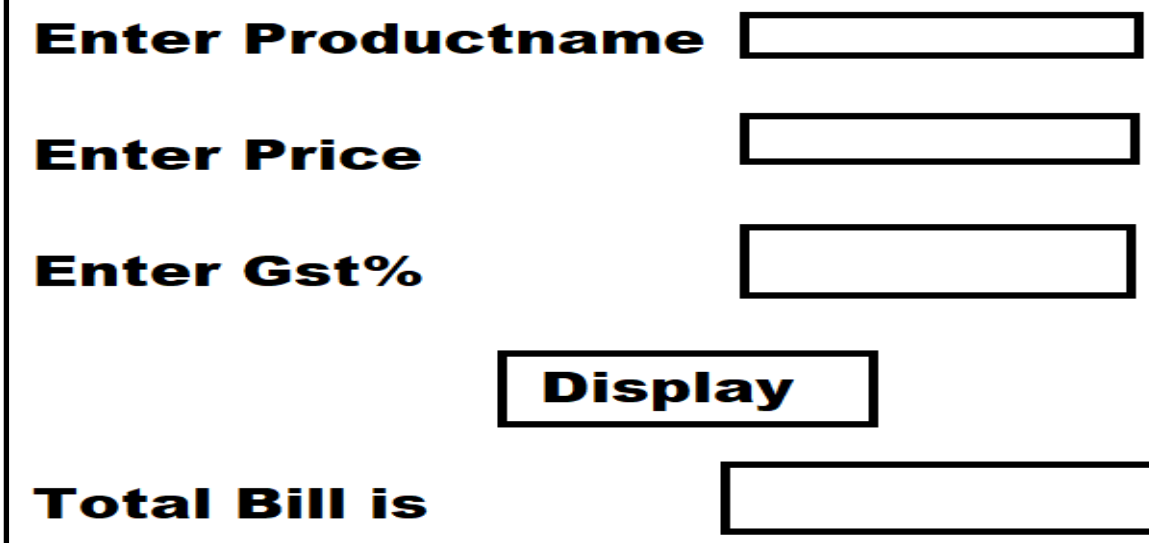

Example:-




```
protected void Button1_Click()
{
   int m1 = int.Parse(TextBox2.Text);
   int m2 = int.Parse(TextBox3.Text);
   int m3 = int.Parse(TextBox4.Text);
   int total = m1 + m2 + m3;
   int per = total / 3;
   TextBox5.Text = total.ToString();
   TextBox6.Text = per.ToString();
}
```

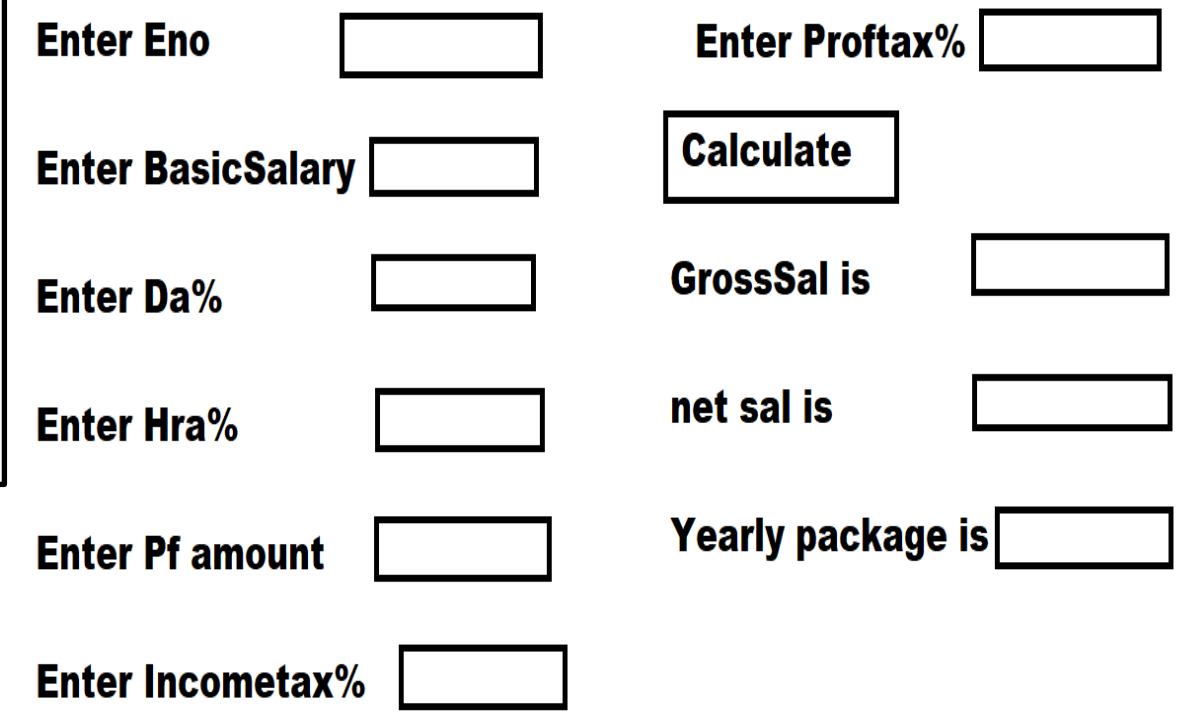
Example:-

SATHYATECHNOLOGIES



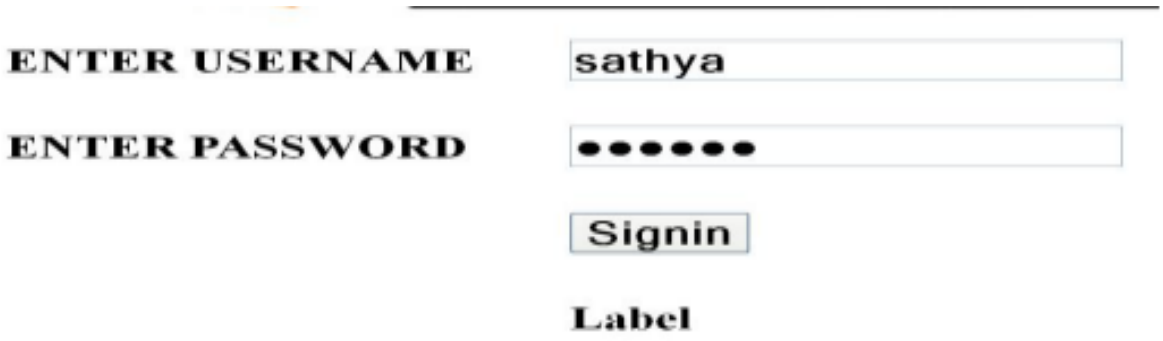
Example:-

Ex:-



KANNABABU

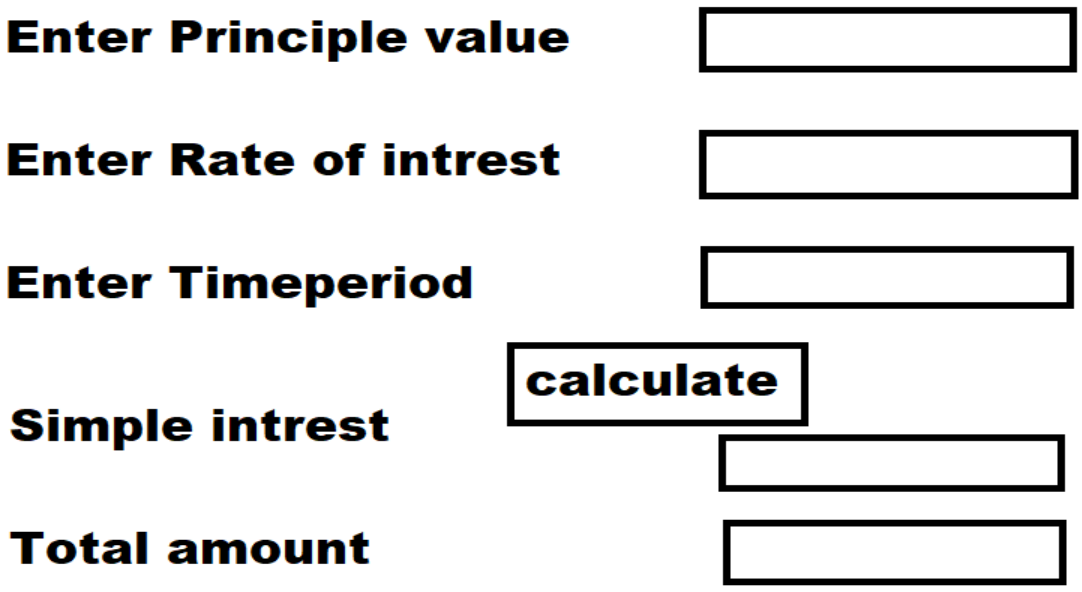
Ex:-



Requirement:-

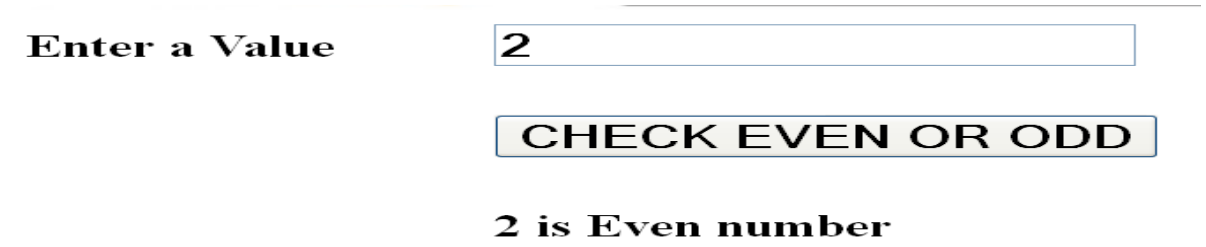
- Username must not be empty
- Password must not be empty
- if the username is sathya and password is sathya Display welcome sathya in label otherwise display invaliduser in label

Ex:-

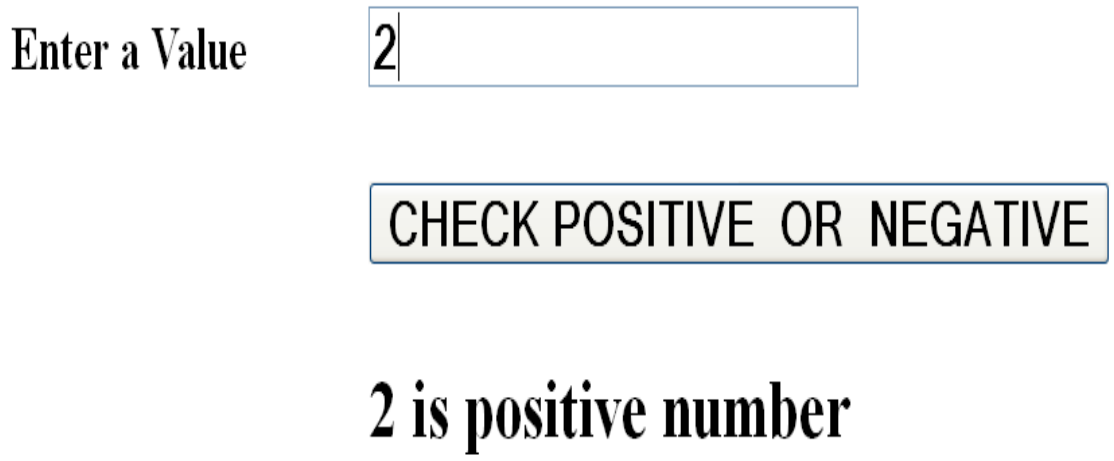


Ex:-

Design the below page



Ex:-



Ex:-

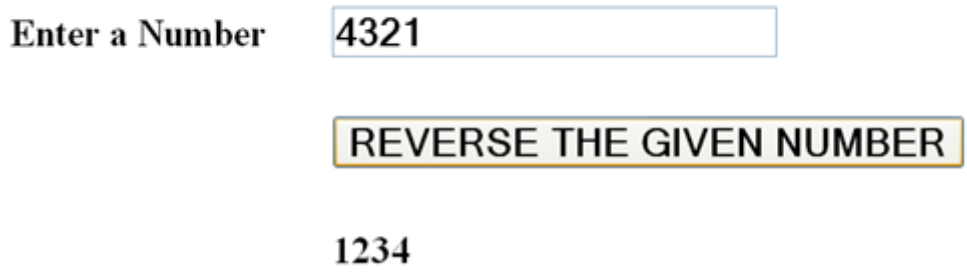
```
protected void Button1_Click(object sender, EventArgs e)
 {       string s = "";
    if (TextBox1.Text=="")
    {   s = "uname must not be empty" + "<br>";          }
    if (TextBox2.Text=="")
    {   s = s + "pwd must not be empty" + "<br>";         }
    if (TextBox3.Text == "")
    {    s = s + "cpwd must not be empty" + "<br>";         }
    if (TextBox1.Text ==TextBox2.Text)
    {
       s = s + "uname and pwd must not be same" + "<br>";
    }
    if (TextBox2.Text != TextBox3.Text)
    {
       s = s + "pwd mismatch" + "<br>";
    }
    Label1.Text=s;
 }
```

Ex:-

Design the Below page

Requirement:-when user enters a value and b value and click on Button swap both the values and display in Textbox3 and textbox4

Ex:-Design the below page

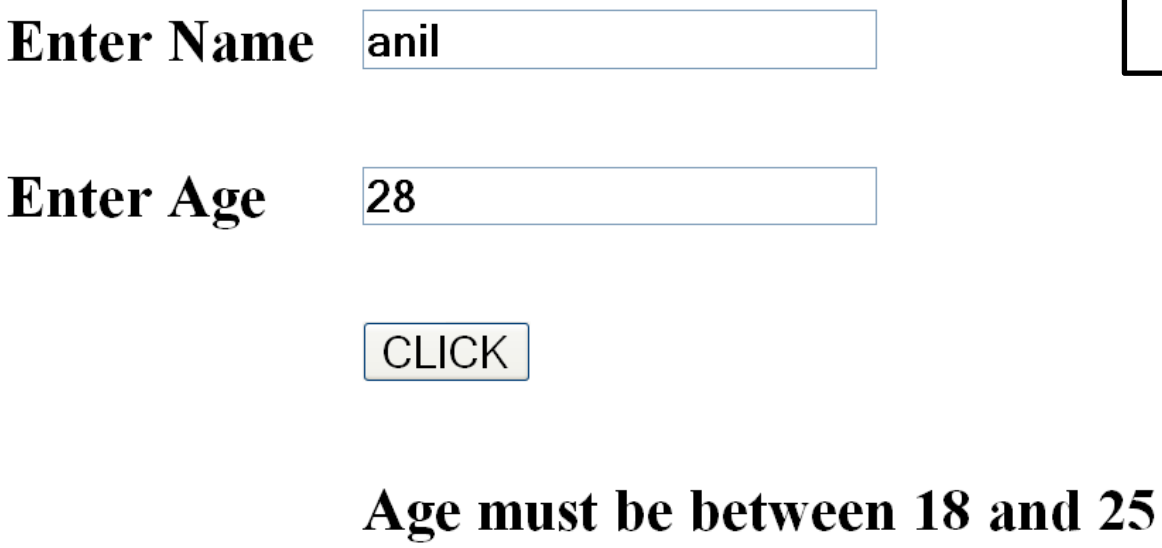


Requirement: - when user enters a number in Textbox1 and click on Button display the reverse number in Label

Ex:-Design the below page

Requirement: - username and password must not be same
Password and confirm password must be same

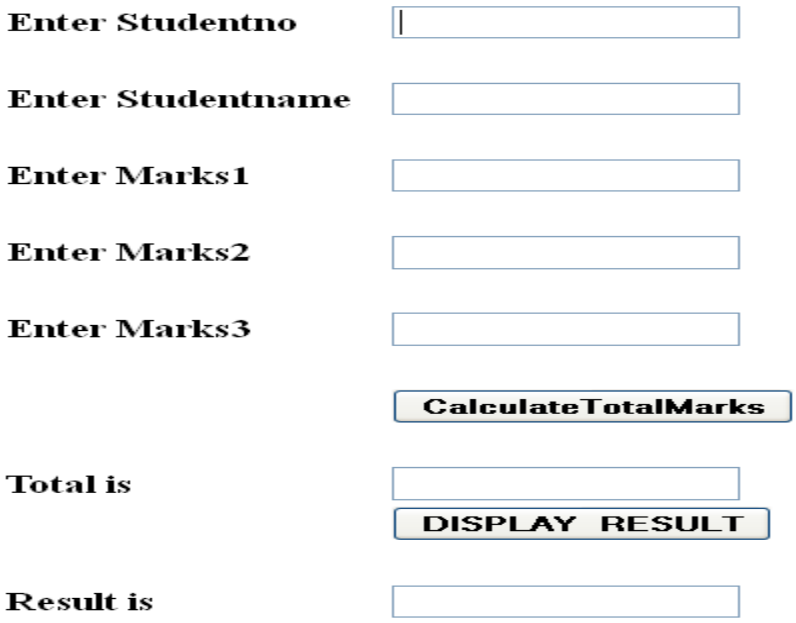
Ex:-Design the below page(Age must be between 18 and 25)



Ex:-Design the below page

Requirement:-

- when user enters sno,sname,m1,m2,m3 calculate total marks and display total in textbox
- if total marks>=250 and <=300 display First class
- if total marks>=200 and <250 display Second class
- if total marks>=150 and <200 display Third class
- if total marks<150 display Falied



CheckBox:-it is used to select morethan one item among group of items.

It is used to get multiple inputs from the user. It allows user to select choices from the set of choices.

Property	Description
id	it is used to set the name for the control
Text	It is used to set text to be shown for the control.
Checked	It is used to set check state of the control either true or false.
AutoPostBack=True/False	if this property is set to true then request will go to server

	when user selects checkbox
Events:-	

Enter Yourname

Select Course C Java .net

Display

Label

2. Text
3. Checked=true/false
   it is a boolean property which will return either true or false
   if the checkbox is checked then it will return true
   otherwise false

Algorithm:-
1. start
2. read the value from t1 and store in name
3. declare a variable with name s
4. if checkbox1 is checked then add c1.Text to s
   if checkbox2 is checked then add c2.Text to s
   if checkbox3 is checked then add c3.Text to s
5. add name to s and display in label1
6. stop

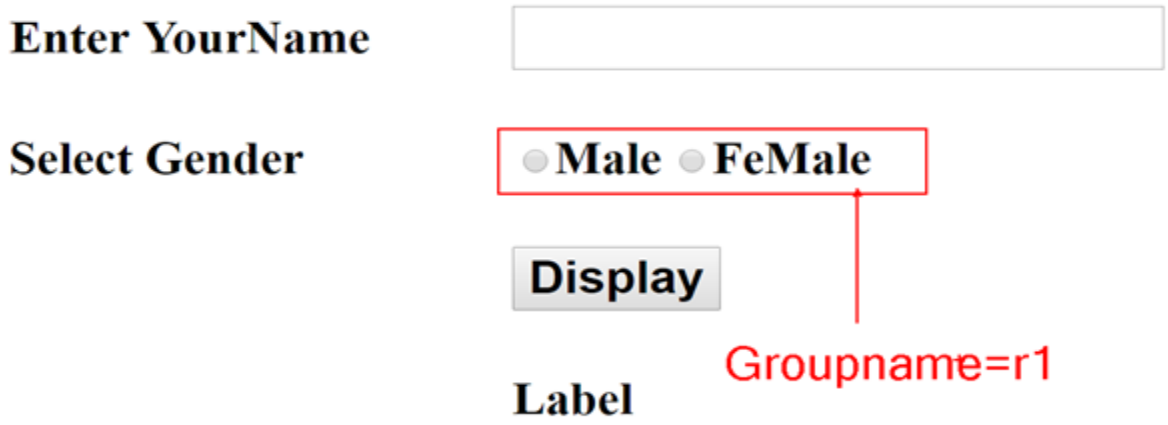
```
protected void Button1_Click()
{
    string name = TextBox1.Text;
    string s="";
    if(CheckBox1.Checked==true)
    {
        s = s + CheckBox1.Text+",";
    }
    if(CheckBox2.Checked==true)
    {
        s = s + CheckBox2.Text+",";
    }
    if (CheckBox3.Checked==true)
    {
        s = s + CheckBox3.Text+",";
    }
    Label1.Text = name + " courses are " + s;
}
```

RadioButton:- it is used to select only one item among group of items

Properties:-

1. id
2. Text
3. Checked=True/False
4. GroupName:- This property is used to group the radiobuttons

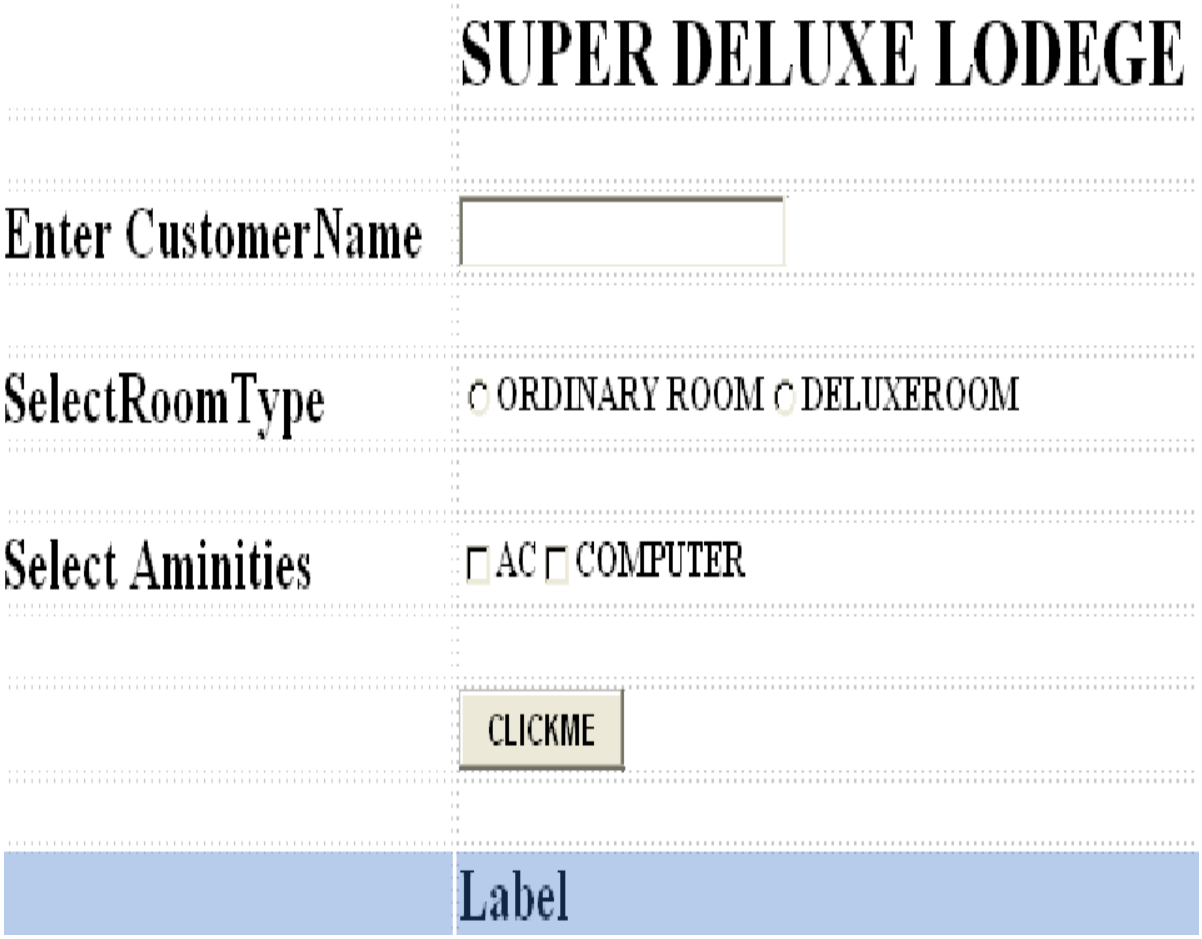
Ex:-




```
protected void Button1_Click()
{
    string s = TextBox1.Text;
    string gender = "";
    if (RadioButton1.Checked==true)
    {
        gender = RadioButton1.Text;
    }
    else
    {
        gender = RadioButton2.Text;
    }
    Label1.Text = s + " is " + gender;
}
```

Ex:-

Design the Below page:-



Requirement:

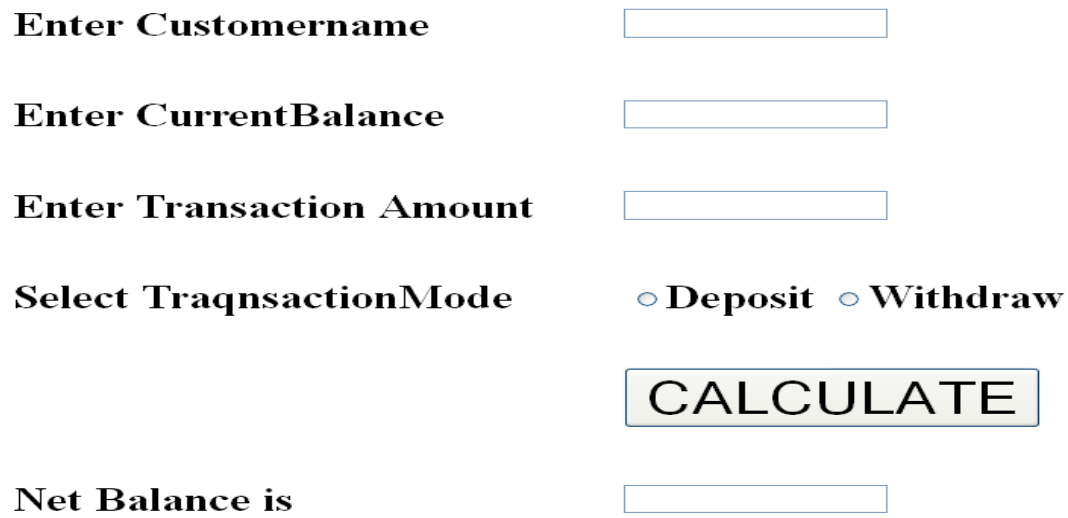
- when user Enters John as Customer name and Select ordinary Room then display John you selected ordinary Room which cost 1500/-
- When user selects Ordinary Room then Display Select Amenities, AC and Computer in invisible mode
- When user selects Deluxe Room then Display Select Amenities, AC and Computer in visible mode the cost of deluxe room is 1500/- The cost of AC is 500/- and The cost of Computer is 300/-

- When user selects Deluxe Room with AC then Display John you selected Deluxe Room with AC which cost 2000/-
- When user selects Deluxe Room with Computer then Display John you selected Deluxe Room with Computer which cost 1800/-
- When user selects Deluxe Room with AC and Computer then Display John you selected Deluxe Room with AC and Computer which cost 2300/-

Ex:- Design the below page

Requirement:-

- Whenever user enters customername,currentbalance, transaction amount and when user selects Deposit add current balance to transaction amount and display total in netbalance
- when user selects Deposit deduct transaction amount from current balance and display the total in netbalance

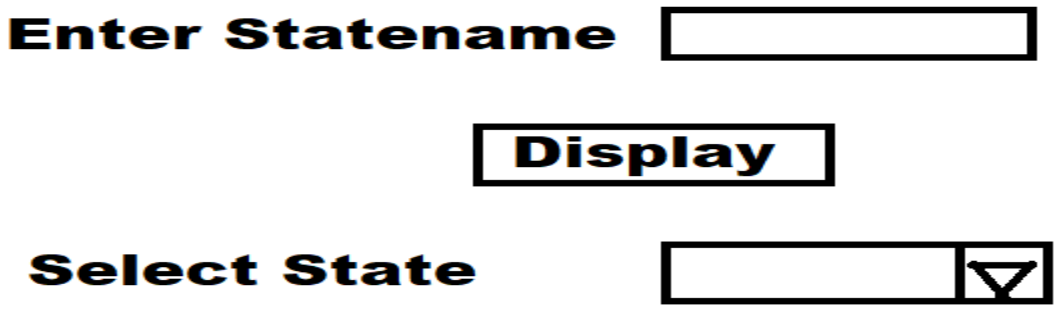


Dropdownlist: - **Dropdownlist** control allows user to select only single item at a time.

The Dropdown List is a web server control which is used to create an HTML Select component. It allows us to select an option from the dropdown list. It can contain any number of items.

id	it is used to set the name for the control
Text	It is used to set text to be shown for the control.
SelectedValue	Get the value of the Selected item from the dropdown box.
SelectedIndex	Gets or Sets the index of the selected item in the dropdown box.
SelectedItem	Gets the selected item from the list.
Items	Gets the collection of items from the dropdown box.
DataTextField	Name of the data source field to supply the text of the items. (No need to set when you are adding items directly into .aspx page.)
DataValueField	Name of the data source field to supply the value of the items. (No need to set when you are adding items directly into .aspx page.)
DataSource	The datasource that populates the items in the dropdown box. (Generally used when you are dynamically generating the items from Database.)
AutoPostBack	true or false. If true, the form is automatically posted back to the server when user changes the dropdown list selection. It will also fire OnSelectedIndexChanged method.
OnSelectedIndexChanged	Method name that fires when user changes the selection of the dropdown box. (Fires only when AutoPostBack=true.)

Ex:-



Ex:-

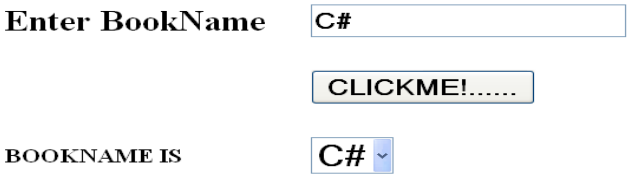
Requirement: -

Whenever customer enters name and select state display the customer name with state name in Label

Ex:-

Design the below page

Requirement: -whenever user enter bookname in textbox1 and click on Button Add the bookname in Dropdownlist control



Ex:- Design the below page

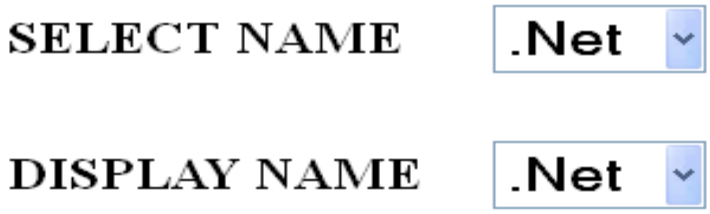
Requirement:-

- When the user selects and clicks on submit button it must calculate total and displayed in total fee text box.

- If it's fast track, +50 will be added for each subject and total will be calculated.
- If it's super fast track, +100 will be added for each subject and total will be calculated.

Ex:-

Design the below page

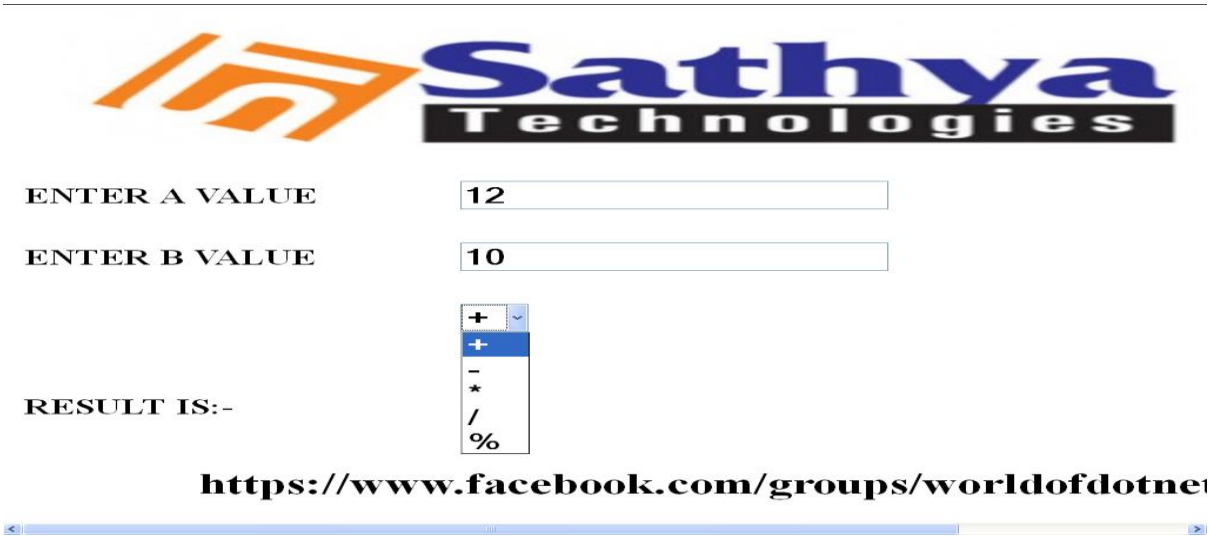


Requirement:- whenever user selects course name in Dropdownlist1 then Display the Selected coursename in Dropdownlist2

Ex:-

Design The Below Page and when user enter name and select birthday display name and birthday in label

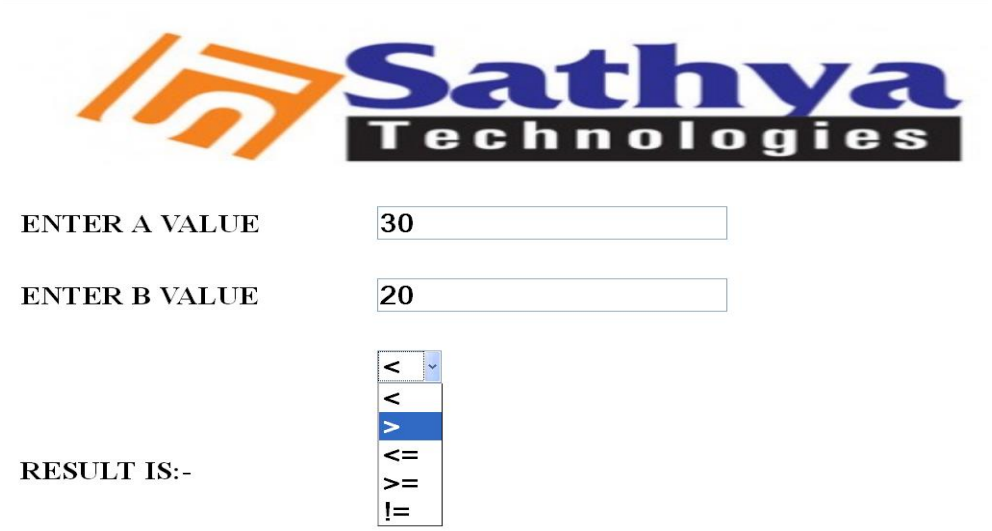
Ex:-Design the Below page



Requirement:-when ever user enters a value in textbox1 and b value in textbox2 and select operator in Dropdownlist then display the result in Label

Ex:-

Design the below page



Requirement:- whenever user Enters a value and b value and Selects the operator in Dropdown list display the result in label

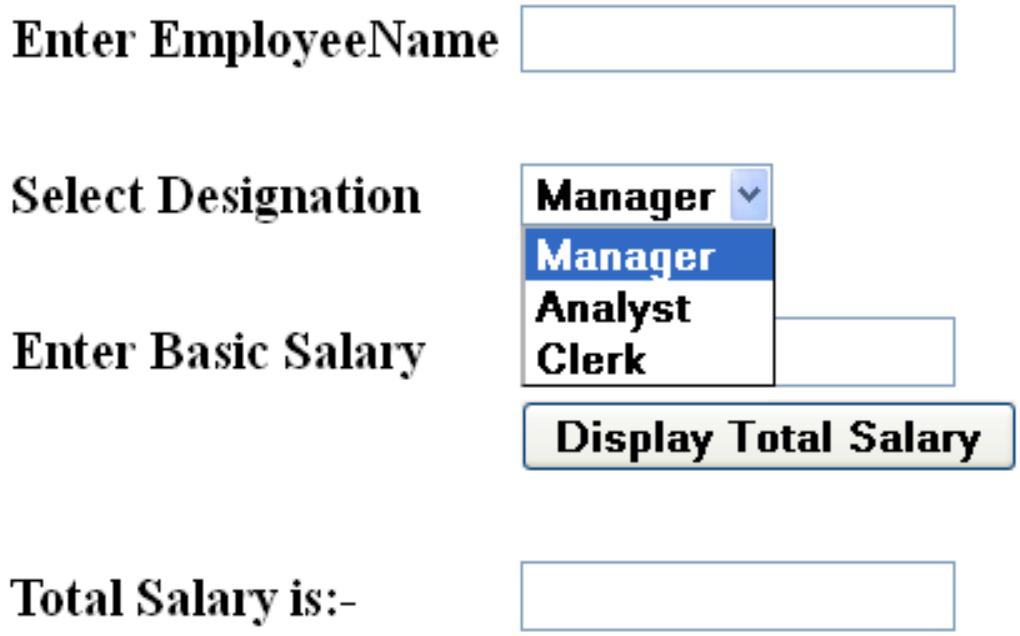
Ex:-Design The below page

Requirement:-

- When user enters srno and erno we have to calculate number of units and display in textbox3 (Hint:- noofunits=srno-erno)Consider srno=300 and erno 200
- When user selects slab type as industry then unit rate is 5/-

- When user selects slab type as commercial complex then unit rate is 4/-
- When user selects slab type as Residence then unit rate is 3/-

Ex:-Deign the below page



Requirement:-

- Whenever user enters Employee name and selects designation and enter basicsalary and when user clicks on Button add bonus to basicsalary and display total salary
- If designation is Manager then bonus is 20% on his Basic salary
- If designation is Analyst then bonus is 10% on his Basic salary
- If designation is Clerk then bonus is 5% on his Basicsalary

Chapter-5

HyperLink,LinkButton, Image Button

B.Kannababu
(Sathyatechnologies)

Navigating the user Request between multiple webpages :-

we can navigate the user request between multiple webpages in 2 ways:-

1. By using Designing Controls
2. By Writing Coding

Designing Controls:-Different types of controls to navigate the user request are:-

1. HyperLink Button

2. Link Button

3. Image Button

HyperLink Button:-it is used to navigate the user request either for plain text or form image

properties:-

1. id
2. Text
3. NavigateUrl:- it is used to set the Destination url address
4. ImageUrl

EX:-

Drag and HyperLink from Toolbox in Page1.aspx

goto-->project--->Addnewitem--->select webform-->

name=page2.aspx

select HyperLink---->properties--->

Navigateurl=Page2.aspx

Ex:- to navigate the user request for Image

goto--->solutionExplorer--->rc on project---->Addnewfolder

name=images

rc on images--->Add Existingitem----> select an image from desktop

Drag and Drop HyperLink from Toolbox in Page1.aspx

rc on HyperLink -->properties---> imageurl=select image from images folder

Navigateurl=Page2.aspx

LinkButton:-it is used to postback the request to server

properties:-

1. id
2. Text
3. PostBackUrl

Link button is used to postback the request only for plain text

ImageButton:-it is used to postback the request to server when user clicks on image

properties:-

1. id
2. Text
3. PostBackUrl
4. Imageurl

HyperLink	LinkButton	ImageButton
1. it is used to navigate the request from one page to another page either for Text or image	1. it is used to postback the request to server when user clicks on Text	1. it is used to postback the request to server when user clicks on image
2. will have navigateurl	2. will have postbackurl	2. postbackurl
3.double click on hyperlink and we cannot write code	3. we can write code	3. we can wrirte code

Q) what is the Difference between Response.Redirect() and Server.Transfer()?

Response.Redirect()	Server.Transfer()
1. it is used to redirect the user request between multiple webservers	1. it is used to redirect the user request within the same webserver
2. it will not hide the Destination url address	2. it will hide the destination url address
3. Response.Redirect("url");	3. Server.Transfer("url");

KANNABABU

Chapter-6

ADO.net

B.Kannababu
(Sathyatechnologies)

Q) what is Data

Data is collection of rawfacts

Q)what is information?

information is the processed data

Q) what is Database?

Database collection of information

in datbase the data will be stored in the form Tables

Q)what is Table?

Table is Database object which will store the data in the

form of Rows and columns

note:- whenever we want to perform any operations on Data,we need to store the data we need to store the data

inorder to store the data we require some memory

Memory is of 2 Types:-

1. Temporary Memory (Ram)

2. Permanent Memory(Harddisc)

in any programming languages like C#.net or Java we will

store the data in variable or Array or object or collection

i.e the data was saved in Ram

if we want to save the data permanently in Harddisc we have to use Files or Database

Q)what is SQLServer?

SQLServer is Database Server

Database Server is a software which is used to manage Databases

Q)what is SQL?

SQL is a Language which is used to write queries to interact with databases

Q)why to interact with Database?

To perform CRUD Operations

C ----> Create insert

R ----> Retrieve select

U ----> Update update

D ----> Delete delete

Q)what is SqlServer Managementstudio?

SSMS is a software which is used to write Queries to interact with Database server(SqlServer)

Q)in How Many ways we can interact with SqlServer?

we can interact with Sqlserver in 2 Ways:-

1. By using SqlServer Management Studio

2. By using FronteEnd Application

if we want to interact with Sqlserver using SSMS

then we have to write SQL Queries

steps to open SqlServer Management studio:-

1. goto--->start-->Allprograms---->Sqlserver 2008R2--->

SqlServer Management studio

2. Servername

ServerType=DatabaseEngine

Authetication=WindowsAuthetication

SqlServerAuthetication

user id=sa

password=abc

connect

3. goto--->new query --->

syn to create database:-

create database databasename

syn to use database:- use databasename

C#datatype SqlDatataype

byte	tinyint
short	smallint
int	int
long	bigint
float	float
double	small money,money
decimal	decimal
string	varchar

4. syn to create table:-

create table tablename(colname datatype,

colname datatype)

Ex:-

create table emp(eno int,ename varchar(50),

salary money)

5. syntax for insert query:-

insert into tablename values(val1,val2,val3)

6. syn for delete command:-

delete from tablename where condition

delete from emp where eno=101

7. syn for update command:-

update tablename set colname=value where condition

8. syn for select command

select * from tablename

Ex:- select * from emp

9. How to check Tables

goto--->+ObjectExplorer

+DataBases

+demo4pm

+Tables

+emp

+eno

+ename

+salary

rc on emp--->EditTop 200Rows

Q)How to start Sqlserver manually?

goto--->start--->run--->services.msc--->

rc SqlServer(MS SqlServer) ---->start

Q)what is Frontend application?

The application where enduser will interact is called as Frontend application

Ex:- facebook,irctc,icici

Frontend application can be developed by using Frontend Tools like java,ASp.net,ASP.net MVC etc...

Q)what is Backend Application?

The application where the user data is saved permanently is called as Backend application

Backend Application

Backend Application can be developed by using Backend Tools like Sqlserver,Oracle,Mysql etc..

Q) what is ADO.net ?

Active x Database Object Network Enable Technology

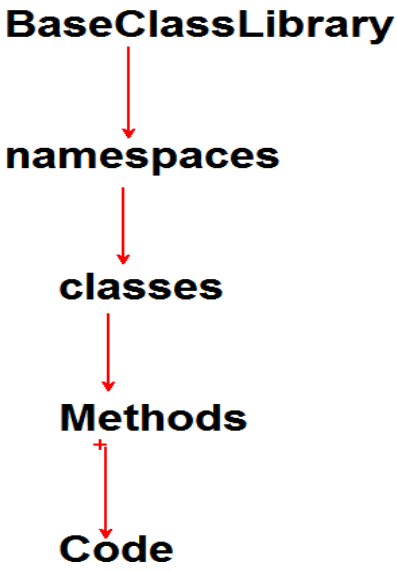
ADO.net is a Technology which is used to create the Communication between Frontend application and Backend Application

Different Types of Architectures in ADO.net:-

1. Connection Oriented Architecture
2. DisConnection Oriented Architecture

Connection Oriented Architecture:- in this Architecture we need to create the connection to the database server ,open the connection,pass the query,execute the query and close the connection

in order to work with ADO.net MS has given some predefined classes and Methods under predefined namespaces



Q)what is a Driver?

a Driver is a software which is used to create the communication between 2 softwares or a software and Hardware

ADO.net supports Different Types of Drivers to interact with Databases

1. ODBC :- Open Database Connectivity
2. OLEDB:- Object Linking Embeded Database
3. SqlClient

SATHYATECHNOLOGIES

KANNABABU

4. OracleClient

Q)what is .dll?

Dyanmic Link Library

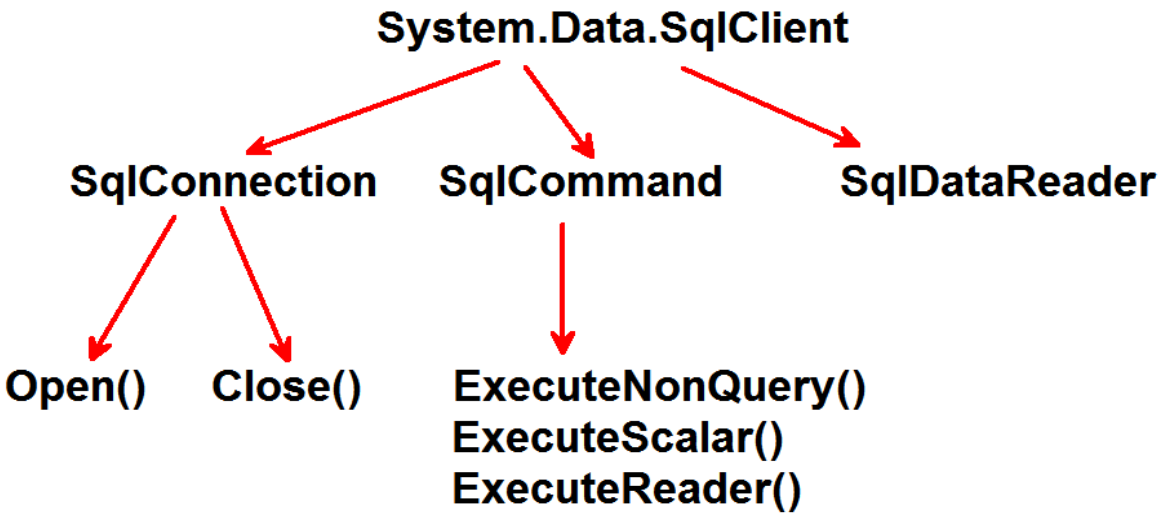
.dll is Reusable but not Executable

.dll is collection of namespaces

namespace is collection of classes

class consits of Methods

Method consists of code




```
namespace System.Data.SqlClient
{
  class SqlConnection
   {
```

```
    public void Open() { }
    public void Close()  {  }
 }
class SqlCommand
{
   public int ExecuteNonQuery(){   }
   public object ExecuteScalar(){  }
   public SqlDataReader ExecuteReader(){  }
} }
```

Q)How to call Open() and Close()?

```
SqlConnection con=new SqlConnection();
con.Open();
con.Close();
```

Q)How to invoke ExecuteNonquery(),Scalar()?

```
SqlCommand cmd=new SqlCommand();
cmd.ExecuteNonQuery();
cmd.ExecuteScalar();
cmd.ExecuteReader();
```

Steps to work with Connection Oriented Architecture:-

1. Declare the namespace

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

2. create the connection by using SqlConnection class

we can connect to SqlServer in 2 ways

a. Windows Authetication

b. SqlServer Authetication

syn to create the connection by using Windows Authetication:-

```
SqlConnection con=new SqlConnection("
data source=.;database=sathya;
integrated security=yes");
```

syn to create the connection by using SqlServer Authetication:-

```
SqlConnection con=new SqlConnection("
data source=.;database=sathya;
user id=sa;password=abc");
```

3. open the connection

```
con.Open();
```

4. pass the query by using SqlCommand class:-

syn:-

SqlCommand cmd=new SqlCommand("pass the query",con);

5. Execute the query

int i=cmd.ExecuteNonQuery();

6. Close the connection

con.Close();

1. open Sqlserver managemnetstudio
   goto-->start--->Allprograms--->Sqlserver 2008R2
   Sqlservermanagementstudio
2. ServerType=Databaseengine
   Servername=.
   Authenticatio=SqlServer Authetication
   user id=sa
   password=abc

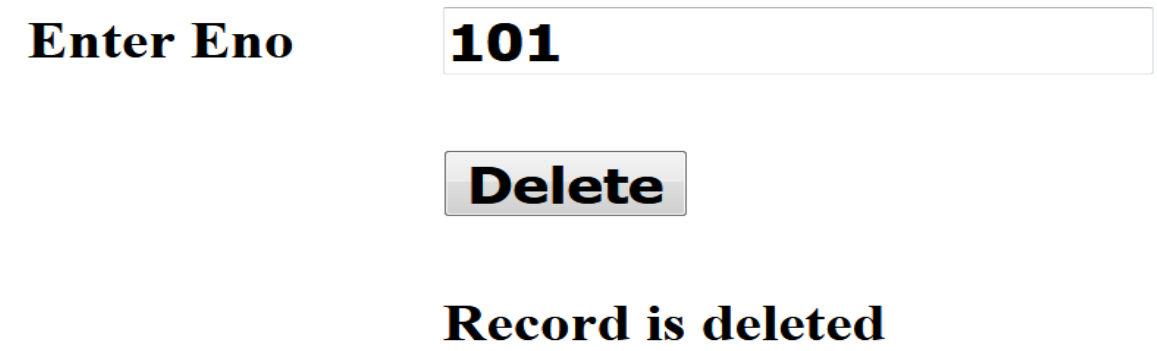
connect

newquery------>

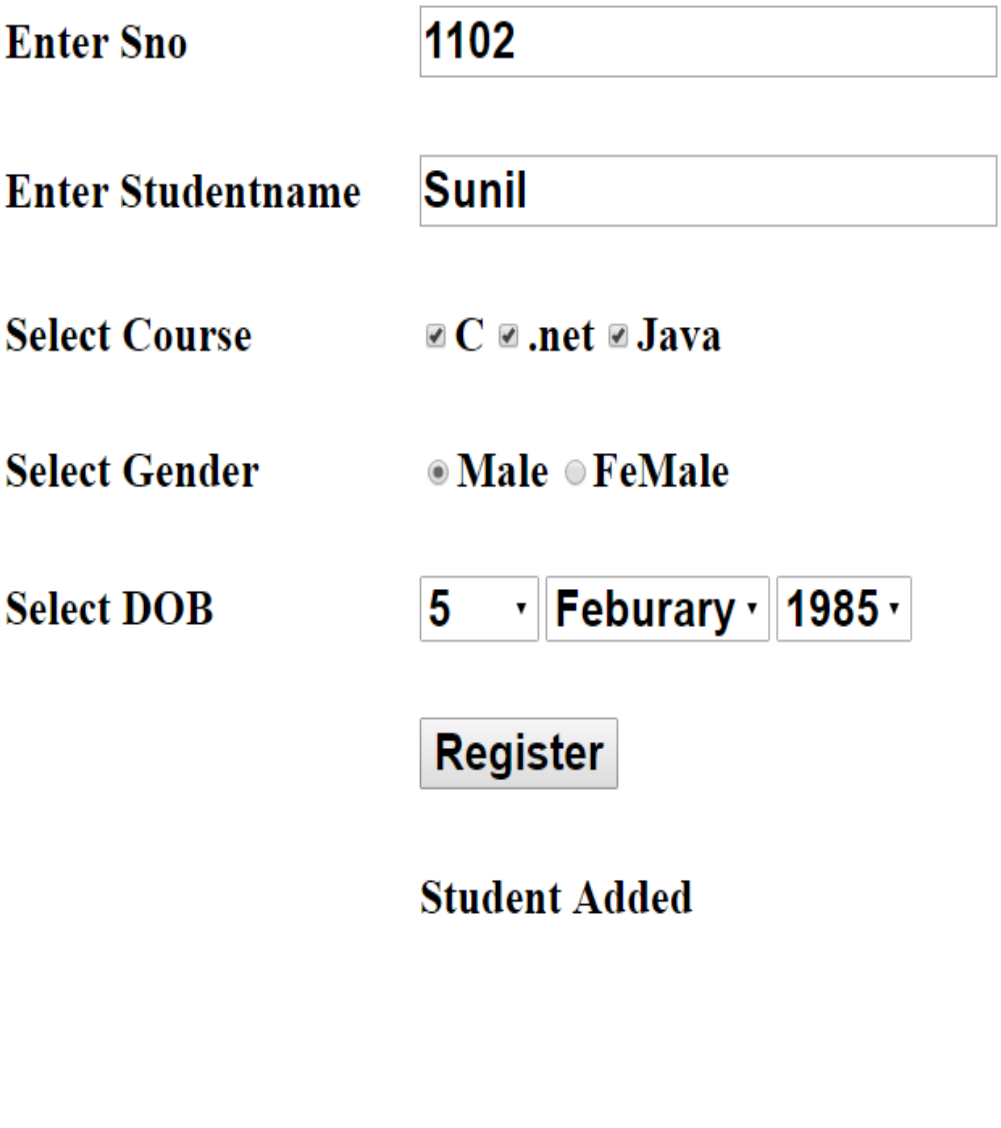
```
create database sathya2019
use sathya2019
create table emp(eno int,ename varchar(50),
salary money)
```

SATHYATECHNOLOGIES

```
using System.Data.SqlClient;

 protected void Button1_Click(object sender, EventArgs e)
 {
       SqlConnection con = new SqlConnection("
         user id=sa;password=abc;database=sathya2019;data source=.");
       con.Open();
       string query="insert into emp
       values('"+TextBox1.Text+"','"+TextBox2.Text+"','"+TextBox3.Text+"')";
       SqlCommand cmd = new SqlCommand(query,con);
       int i=cmd.ExecuteNonQuery();
       con.Close();
       if (i==1)
       {
          Label1.Text = "Record inserted successfully";
       }
       else
       {
          Label1.Text = "Failed";
       }
    }
```




```
using System.Data.SqlClient;

    protected void Button1_Click(object sender, EventArgs e)
     {
        SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=sathya2019;data source=.");
         con.Open();
         string query="delete from emp where eno='"+TextBox1.Text+"'";
         SqlCommand cmd = new SqlCommand(query,con);
        int i=cmd.ExecuteNonQuery();
        con.Close();
        if (i==1)
        {
           Label1.Text="Record is deleted";
        }
         else
        {
           Label1.Text="Failed";
               }
     }
```


Enter Sno
1102
Enter Studentname
Sunil
Select Course
C .net Java
Select Gender
Male FeMale
Select DOB
5 Feburary 1985
Register
Student Added

THYA-PC.mydemo - dbo.student | SQLQuery1.sql - (....mydemo (sa (56))* | SATHYA-PC.m

sno	sname	course	gender	dob
101	Anil	C.net	Male	5-January-198
102	Sunil	C.netJava	Male	5-Feburary-19
NULL	NULL	NULL	NULL	NULL


```
using System.Data.SqlClient;

    protected void Button1_Click(object sender, EventArgs e)
     {
        int sno = int.Parse(TextBox1.Text);
        string sname = TextBox2.Text;
        string course = "";
        if (CheckBox1.Checked==true)
        {
           course = course + CheckBox1.Text;
        }
        if (CheckBox2.Checked==true)
        {
           course = course + CheckBox2.Text;
        }
        if (CheckBox3.Checked == true)
        {
           course = course + CheckBox3.Text;
        }
        string gender = "";
        if (RadioButton1.Checked==true)
        {
           gender = RadioButton1.Text;
        }
```

```
        else
        {
           gender = RadioButton2.Text;
        }
        string dob = DropDownList1.SelectedItem.ToString() + "-" +
DropDownList2.SelectedItem.ToString() + "-" +
DropDownList3.SelectedItem.ToString(); ;
        SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=mydemo;data source=.");
        con.Open();
        string query="insert into student
values('"+sno+"','"+sname+"','"+course+"','"+gender+"','"+dob+"')";
        SqlCommand cmd = new SqlCommand(query,con);
        int i=cmd.ExecuteNonQuery();
        con.Close();
        if(i==1)
        {
           Label1.Text = "Student Added";
        }
        else
        {
           Label1.Text = "Failed";
        }
     }
```

SqlCommand:-SqlCommand class provides 3 predefined Methods to execute the query

1. ExecuteNonQuery()

2. ExecuteScalar()

3. ExecuteReader()

ExecuteNonQuery():- whenever we want to perform DML operations like insert,delete,update then we have to use

ExecuteNonQuery()

```
int i=cmd.ExecuteNonQuery();
```

ExecuteScalar():- This method will return single cell value The returntype of ExecuteScalar() is object

object :- object is special datatype

we can store any type of value in object

```
object o=10;  valid
object a="abc";  valid
object o=cmd.ExecuteScalar();
```

Q)waq to display ename whose eno is 101?

```
select ename from emp where eno=101
```

Q)waq to display salary of anil?

```
select salary from emp where ename='anil'
```

Q)waq to display max salary of an emp?

```
select Max(salary) from emp
```

Q)waq to display min salary of an emp?

```
select Min(salary) from emp
```

Q)waq to display avg salary of an emp?

```
select Avg(salary) from emp
```

Q)waq to display total salary of all emps?

```
select Sum(salary) from emp
```

Q)waq to count the no of emps working in the company?

```
select count(*) from emp
```

ExecuteReader():- whenever we want to return single row,single column,multiple rows,multiple columns or entire table then use ExecuteReader()

The return type of ExecuteReader is SqlDataReader

```
SqlDataReader dr=cmd.ExecuteReader();
```

Q)waq to Display emp details?

```
select * from emp
```

Q)waq to display enames?

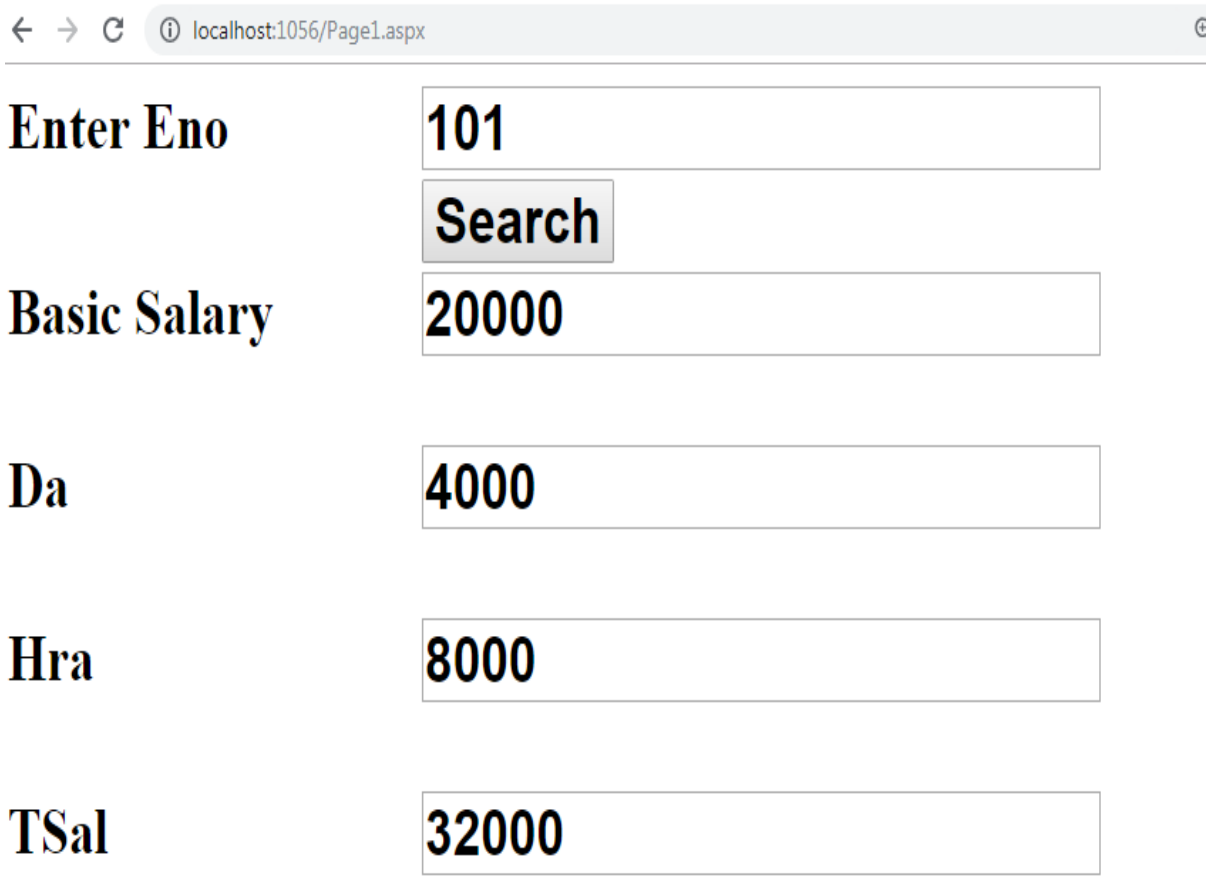
```
select ename from emp
```

Q)waq to display emp details whose eno is 101?

```
select * from emp where eno=101
```

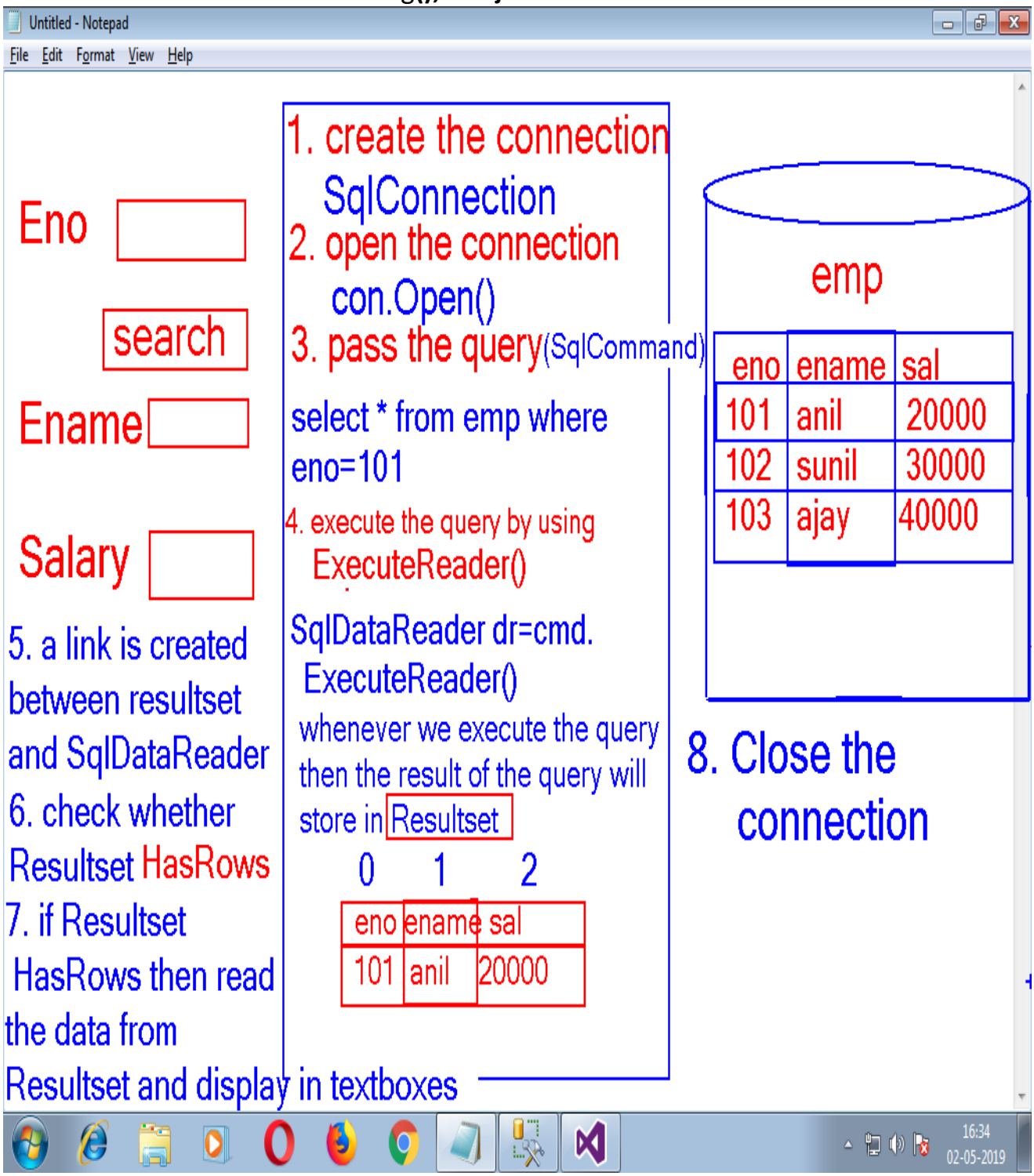
Q)waq to display emp details whose sal>20000?

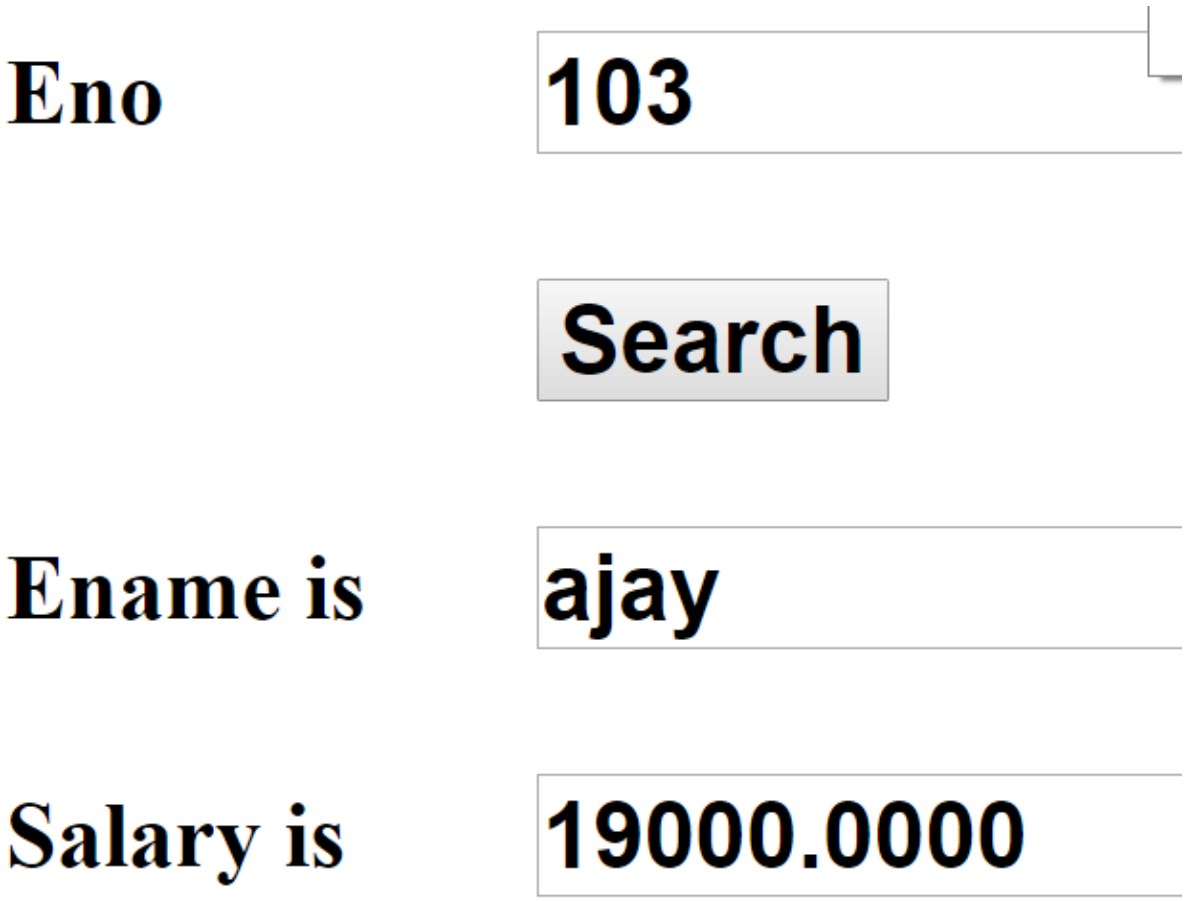
```
select * from emp where sal>20000
```




```
using System.Data.SqlClient;

   protected void Button1_Click(object sender, EventArgs e)
      {
         SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=demo;data source=.");
          con.Open();
          string query="select salary from emp where eno='"+TextBox1.Text+"'";
          SqlCommand cmd = new SqlCommand(query,con);
         object o=cmd.ExecuteScalar();
         double sal = Convert.ToDouble(o);
         con.Close();
         double da = 0.2 * sal;
         double hra = 0.4 * sal;
         double tsal = sal + da + hra;
         TextBox2.Text = sal.ToString();
         TextBox3.Text = da.ToString();
```

```
        TextBox4.Text = hra.ToString();
        TextBox5.Text = tsal.ToString();      }
```






```
using System.Data.SqlClient;

    protected void Button1_Click(object sender, EventArgs e)
     {
        SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=demo;data source=.");
        con.Open();
        string query="select * from emp where eno='"+TextBox1.Text+"'";
        SqlCommand cmd=new SqlCommand(query,con);
        SqlDataReader dr=cmd.ExecuteReader();
        if(dr.HasRows)
        {
           if(dr.Read())
           {
              TextBox2.Text = dr[1].ToString();
              TextBox3.Text = dr[2].ToString();
```

```
        }
    }       con.Close();    }
```

IsPostBack:-

Http is a communication protocol which is used to create the communication between client and server

Http is responsible to send request to server and is responsible to send response to Browser

in ASP.net we can submit the request to server to access webpages in 2 ways

1. First Request

2. PostBack Request

First Request means the request that was generated to a webpage for first time

PostBackRequest:-whenever user will interact with webpage through controls then postBack request will go to server i.e user clicks on Button,select item inDropdownlist,check the item in CheckBox etc...

The no of postBack requests will depend on the no of controls that the user will interact

IsPostBack:- it is a boolean property which will return either true or false

IsPostBack property is used to check whether the request is First request or PostBack request

if IsPostBack is false then it will consider the request as FirstRequest

if IsPostBack is true then it will consider the request as PostBackRequest

by default IsPostBack is false

if user enter url or user clicks on Hyperlink then the Request is FirstRequest

when user clicks on submit button or any control then the request is PostBackRequest

The code that was return inside Page_Load will gets executed whether the request is First request or PostBack request

Ex:-

```
protected void Page_Load()
{
   Response.Write("i am page load"+"<br>");
}
protected void Button1_Click()
{        Response.Write("i am Button");      }
```

observation:-Each and every time when the request was made to server then Page_Load event will gets executed

and the code that was written inside page_Load will gets executed

if we want to execute the code that was written inside Page_Load only for FirstRequest then we have to use IsPostBack property

i.e if IsPostBack is false then the code that was written

inside Page_Load will gets executed

whenthe Request is PostBack Request then IsPostBack will automatically set to true

```
protected void Page_Load()
{
    if (IsPostBack == false)
    {
        Response.Write("i am page load" + "<br>");
    }
}
protected void Button1_Click()
{
    Response.Write("i am Button");
}
```

Limitations in Connection Oriented Architecture:-

1. in Connection Oriented Architecture we need to manually Open the connection and Close the connection,if we are not closing the connection or any Exception will occur then the connection to the Database will be maintained and the network traffic will be increased and the burden on the server will be increased and the performance will be decreased

2. in Connection Oriented Architecture SqlDataReader class will read the data in Forward only Direction

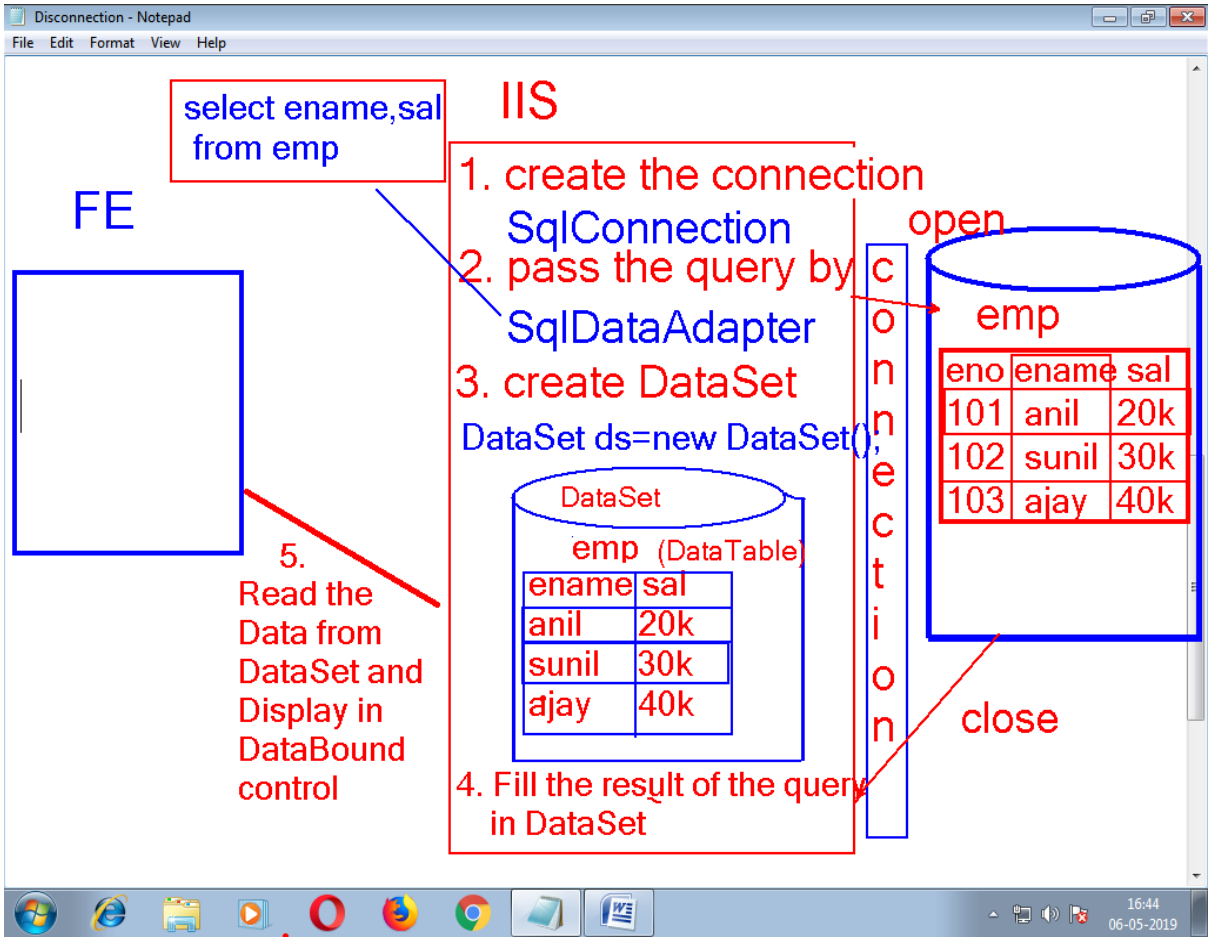
Disconnection Oriented Architecture:-

in DisConnection Oriented Architecture it is not required to open the connection and close the connection

Here The Opening and Closing of connection will done automatically by using SqlDataAdapter class

Steps to work with DisConnection Oriented Architecture:-

1. Design the DataBound control
2. Prepare the Datasource
3. Attach the Datasource to DataBound control



Data Bound Control:-it is used to Display the data from DataSet to Frontend

Different Types of DataBound Controls are :-

1. Repeater Control
2. DataList Control
3. GridView Control
4. ListView Control
5. DetailsView Control
6. DataPager Control

Gridview:-it is used to display the data from DataSet to Frontend

we can perform Deleting,Editing,Paging,Sorting Operations within the Gridview control

Properties:-

1. **id**:-id is the name given for the control
2. **AutoGenerateColumns=True/False:-** if true it is not required to generate the code for Templates
3. **AutoGenerateDeleteButton=True/False:-** if this property is set to true then delete button will display within the Gridview
4. **AutoGenerateEditButton=True/False:-** if True then edit button is displayed within the Gridview control
5. **AutoGenerateSelectButton=True/False:-** if True then select button is displayed within the Gridview control
6. **AllowPaging=True/False:-** if True then paging is enabled within the Gridview control
7. **AllowSorting=True/False:-** if True then sorting is enabled within the Gridview control
8. **ShowFooter=True/False:-** if this property is set to true then FooterRow is enabled within the GridView
9. **ShowHeader=True/False:-** if this property is set to true then HeaderRow is enabled within the GridView
10. **PageSize:**- This property is used to set the no of Pages that we want to Display within the Gridview.

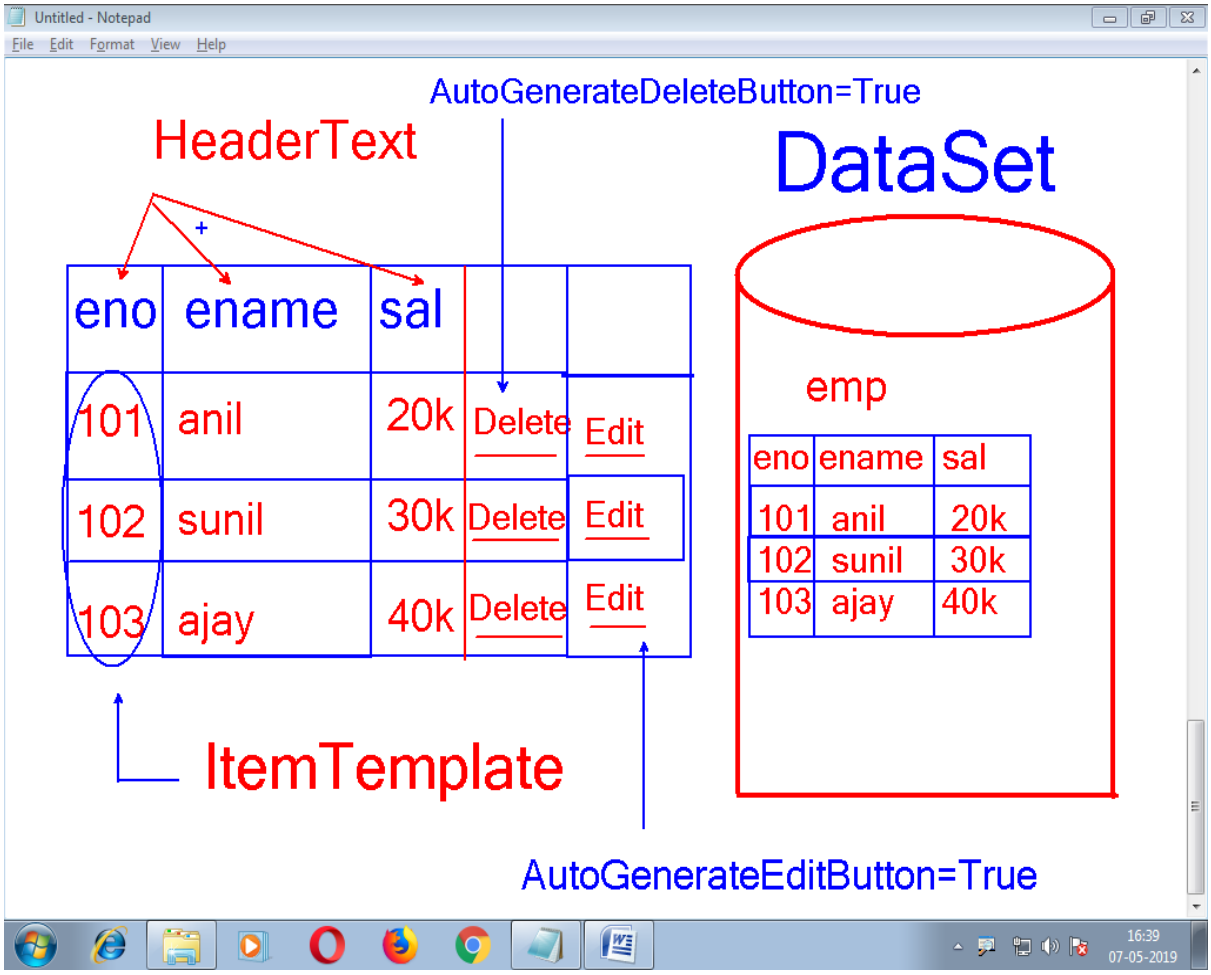
Events :-

1. **RowDeleting**:- This Event will fire when user clicks on delete button within the Gridview
2. **RowEditing**:- This Event will fire when user clicks on Edit button within the Gridview
3. **RowUpdating**:-This Event will fire when user clicks on Update button within the Gridview
4. **RowCancellingEdit**:-This Event will fire when user clicks on Cancel button within the Gridview
5. **SelectedIndexChanged**:- This Event will fire when user clicks on Select button within the Gridview
6. **PageIndexChanging**:- This Event will fire when user clicks on page number within the Gridview
7. **Sorting**:- This Event will fire when user clicks on column headings within the Gridview

Templates:-Templates are used to Design the Databound Controls

Different Types of Templates are:-

1. **HeaderTemplate**:-it is used to Display the Column headings within the Databoundcontrol
2. **ItemTemplate**:- it is used to bind Dataset data column by column within the Gridview control
3. **EditItemTemplate**:- it is used to perform Editing operations within the Gridview control
4. **FooterTemplate**:- it is used to add controls within the Footerrow



Ex:-

1. Design DataBoundControl
2. Drag and Drop Gridview control from Toolbox

eno	ename	salary
101	anil	20000.0000
102	sunil	23000.0000
103	ajay	19000.0000
104	james	20000.0000

2. Prepare DataSource

Code for PageLoad:-

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

    protected void Page_Load(object sender, EventArgs e)
    {
      //create the connection
       SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=demo;data source=.");
       //pass the query
       string query="select * from emp";
       SqlDataAdapter da = new SqlDataAdapter(query,con);
       //create object for DataSet
       DataSet ds = new DataSet();
       //Fill the result of the query in DataSet
       da.Fill(ds, "emp");
       //Atatch dataset to Databound control
       GridView1.DataSource = ds;
       GridView1.DataBind();
     }
```

Example to Display the Data in Gridview control by setting AutoGenerateColumns=False:-

By Default AutoGenerateColumns=True

if AutoGenerateColumns=True it is not required to generate the code for Templates

if we want to customize the Gridview control then we have to set AutoGenerateColumns=False

i.e if we want to perform any other operations apart from Display the data like Editing,Deleting,Paging,Sorting etc... we have to set AutoGenerateColumns=False

SATHYA-PC.Demo - dbo.emp* | SATHYA-

eno	ename	salary
101	anil	20000.0000
102	sunil	23000.0000
103	ajay	19000.0000
104	james	20000.0000

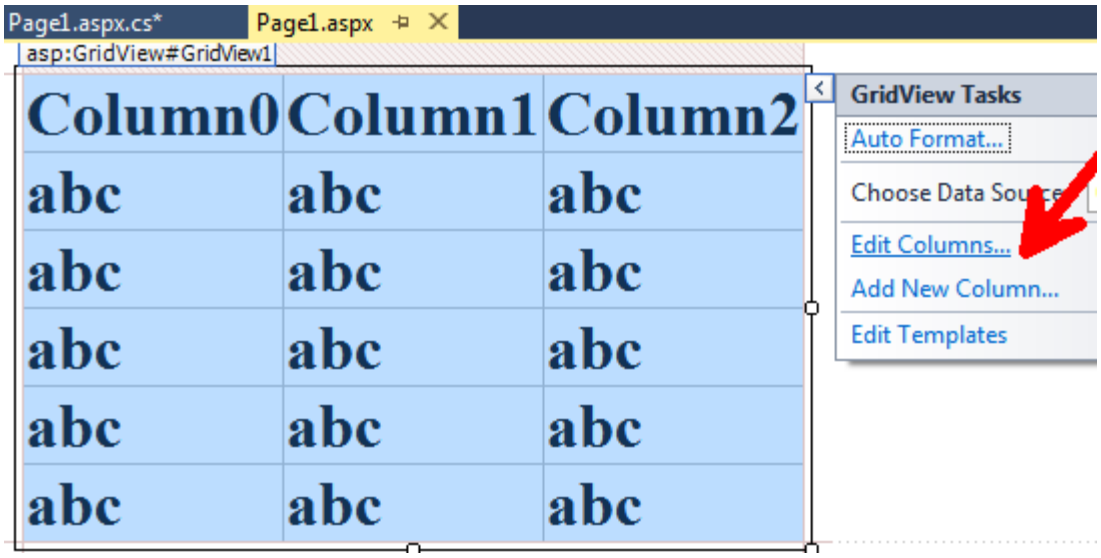
Ex:-

1. Drag and Drop Gridview control from Toolbox

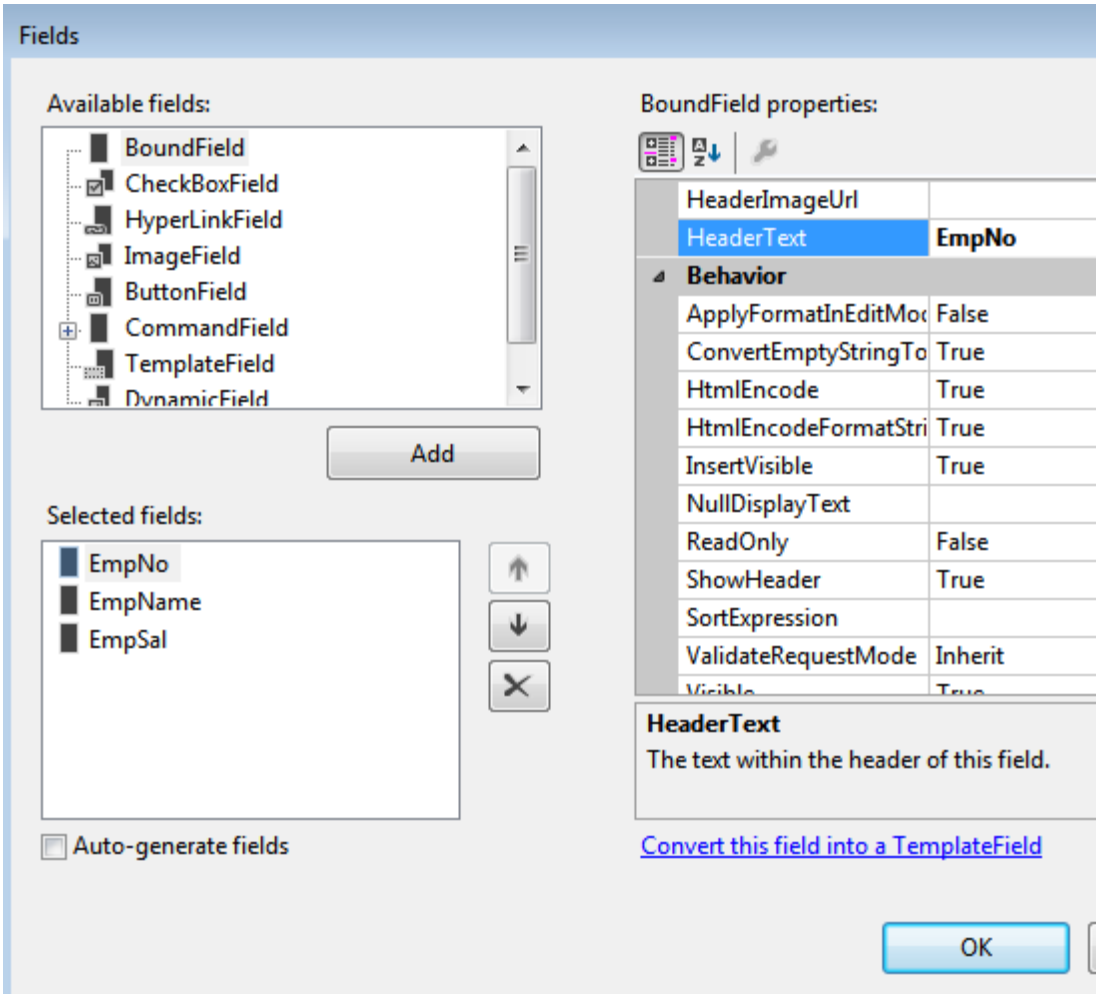
2.

```
using System.Data;
using System.Data.SqlClient;
 public partial class Page1 : System.Web.UI.Page
   {
      private void FillData()
      {
         SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=demo;data source=.");
         SqlDataAdapter da = new SqlDataAdapter("select * from emp",con);
         DataSet ds = new DataSet();
         da.Fill(ds, "emp");
         GridView1.DataSource = ds;
         GridView1.DataBind();
      }
      protected void Page_Load(object sender, EventArgs e)
      {
```

```
        if (IsPostBack==false)
        {
            FillData();
        }
    }
}
```

3. select Gridview--->properties--->AutoGenerateColumns=False
4. 

5. Select Gridview→Ellipse--→EditColumns-→Add 3 BoundFields


Fields
Available fields:
BoundField
CheckBoxField
HyperLinkField
ImageField
ButtonField
CommandField
TemplateField
DynamicField
Add
Selected fields:
EmpNo
EmpName
EmpSal
Auto-generate fields
BoundField properties:
HeaderImageUrl
HeaderText
EmpNo
Behavior
ApplyFormatInEditMo
False
ConvertEmptyStringTo
True
HtmlEncode
True
HtmlEncodeFormatStri
True
InsertVisible
True
NullDisplayText
ReadOnly
False
ShowHeader
True
SortExpression
ValidateRequestMode
Inherit
HeaderText
The text within the header of this field.
Convert this field into a TemplateField
OK

5.Select Gridview-->Ellipse-->EditColumns- Add 3 BoundFields

	HeaderText	DataField
BoundField1	Empno	eno
BoundField2	Empname	ename
BoundField3	Empsal	salary

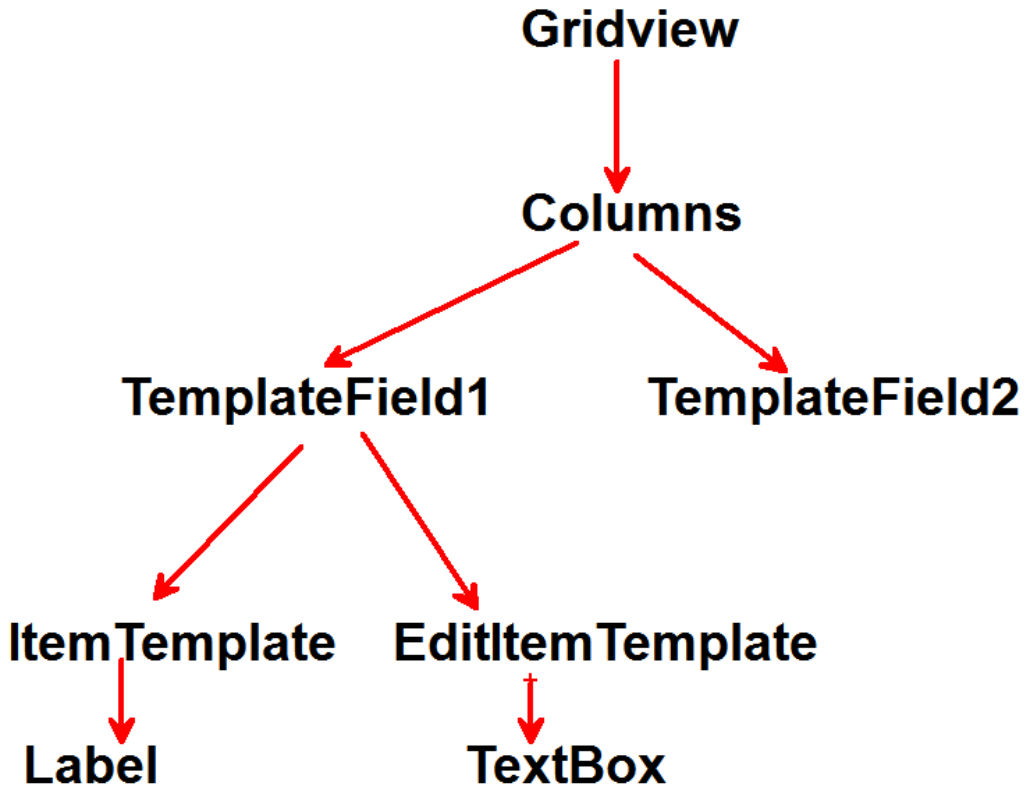
HeaderText:- it is used to display column headings within the Gridview control

DataField:- it is used to set the column name that we want to display within the Gridview

note:- DataFieldname must be same as DataSet Table column name

Select Each BoundField and click on ConvertToTemplateField

goto---->source code and check The code for Templates was generated

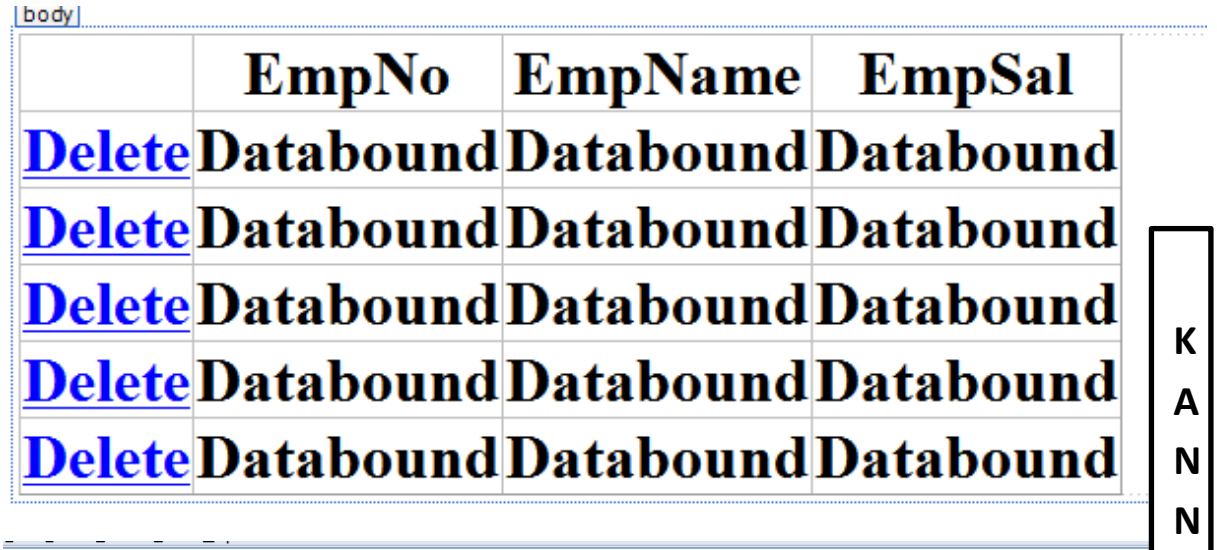


6. Goto-→source code and check

```
<asp:GridView ID="GridView1" runat="server" style="font-weight: 700;
font-size: xx-large" AutoGenerateColumns="False">
        <Columns>
          <asp:TemplateField HeaderText="EmpNo">
            <EditItemTemplate>
              <asp:TextBox ID="TextBox1" runat="server" Text='<%#
Bind("eno") %>'></asp:TextBox>
            </EditItemTemplate>
            <ItemTemplate>
              <asp:Label ID="Label1" runat="server" Text='<%# Bind("eno")
%>'></asp:Label>
            </ItemTemplate>
          </asp:TemplateField>
          <asp:TemplateField HeaderText="EmpName">
            <EditItemTemplate>
              <asp:TextBox ID="TextBox2" runat="server" Text='<%#
Bind("ename") %>'></asp:TextBox>
            </EditItemTemplate>
            <ItemTemplate>
              <asp:Label ID="Label2" runat="server" Text='<%# Bind("ename")
%>'></asp:Label>
            </ItemTemplate>
          </asp:TemplateField>
          <asp:TemplateField HeaderText="EmpSal">
            <EditItemTemplate>
              <asp:TextBox ID="TextBox3" runat="server" Text='<%#
Bind("salary") %>'></asp:TextBox>
            </EditItemTemplate>
            <ItemTemplate>
              <asp:Label ID="Label3" runat="server" Text='<%# Bind("salary")
%>'></asp:Label>
            </ItemTemplate>
          </asp:TemplateField>
        </Columns>
            </asp:GridView>
```

7. select GridView-->properties--->

AutoGenerateDeleteButton=True



8. whenever user clicks on Delete Button within the Gridview then RowDeleting event will fire
select Gridview--->Properties---->Events---->Double click on RowDeletingEvent and write the code

```
protected void GridView1_RowDeleting(object sender,
GridViewDeleteEventArgs e)
    {
        GridViewRow row = GridView1.Rows[e.RowIndex];
        Control c = row.FindControl("Label1");
        Label l1 = (Label)c;
        SqlConnection con = new SqlConnection("user
id=sa;password=abc;database=demo;data source=.");
        con.Open();
        string query="delete from emp where eno='"+l1.Text+"'";
        SqlCommand cmd = new SqlCommand(query,con);
        cmd.ExecuteNonQuery();
        con.Close();
```

```
    FillData();
  }
```

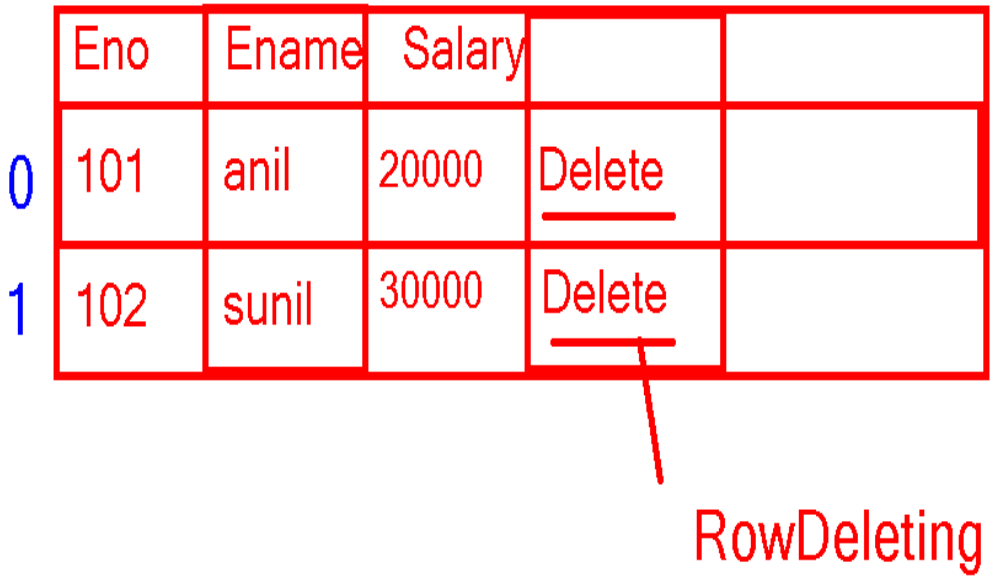
```
            class Control
            {
            }
class Label:Control                 class TextBox:Control
{                                   {
  public string Text{set;get;}        public string Text{set;get;}
}                                    }

            Control c=new Label();
            Label l1=(Label)c;
            l1.Text
```

Gridview is a collection control

Gridview will maintain the data in form of Rows and Columns

in Gridview Each and Every Row can be identified by using index no

Every Row in the Grid is called as GridViewRow



whenever user clicks on delete Button:-

1. count the no of Rows that are available within the Gridview
2. catch the RowIndex where user clicks on Delete button
3. catch the control based on which we want to perform deleting operation within the Gridview
4. Delete the Record

SATHYATECHNOLOGIES

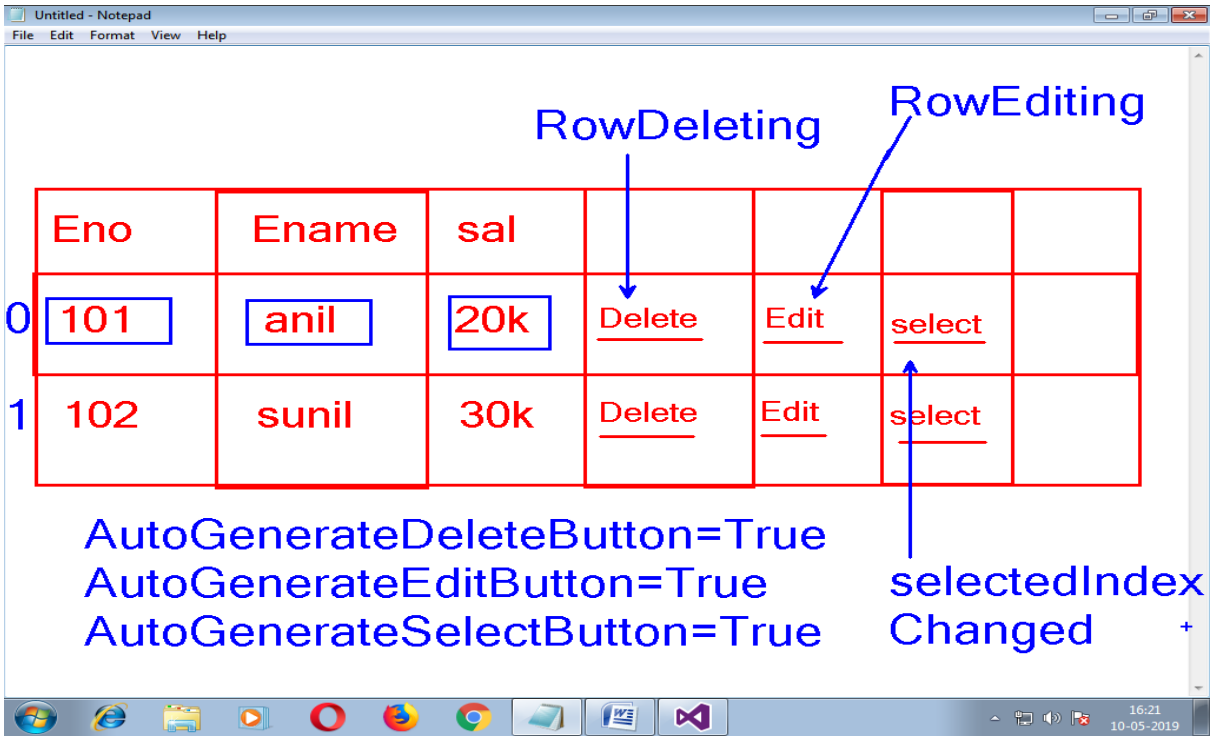
```
class GridViewRow
{
  public Control FindControl(string controlid)
  {
    This method   will return the control object based on id
  }
}
MethodName:- FindControl
returntype     :- Control
i/p                :- controlid
```

count the no of rowsinGV

catch RowIndex

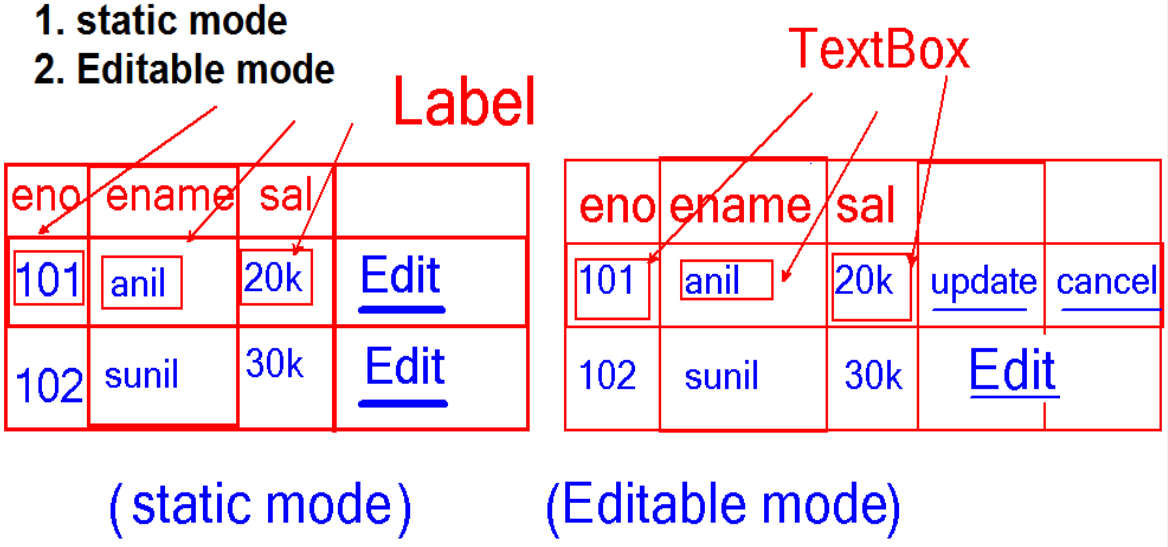
```
GridViewRow row=GridView1.Rows[e.RowIndex];
 Control c1=row.FindControl("label1");
  Label l1=(Label)c1;
```

Find the control within the GridviewRow



Performing Editing Operations within the Gridview control:-

1. Drag and Drop Gridview control
2. select Gridview--->AutoGenerateColumns=False
3. AutoGenerateEditButton=True
4. Gridview will Display the data in 2 Modes
   1. static mode
   2. Editable mode



- By default Gridview will display the data in static mode
- when Gridview is in static mode then the code that was written inside ItemTemplate will gets executed
- when Gridview is in static mode then EditIndex=-1
- when Gridview is in static mode then the data will be displayed in Label Control
- whenever user clicks on Edit Button then Gridview will display the data in Editable mode
- when Gridview is in Editable mode then the code that was written inside EditItemTemplate will gets executed
- when Gridview is in Editable mode then the data will be displayed in TextBox Control

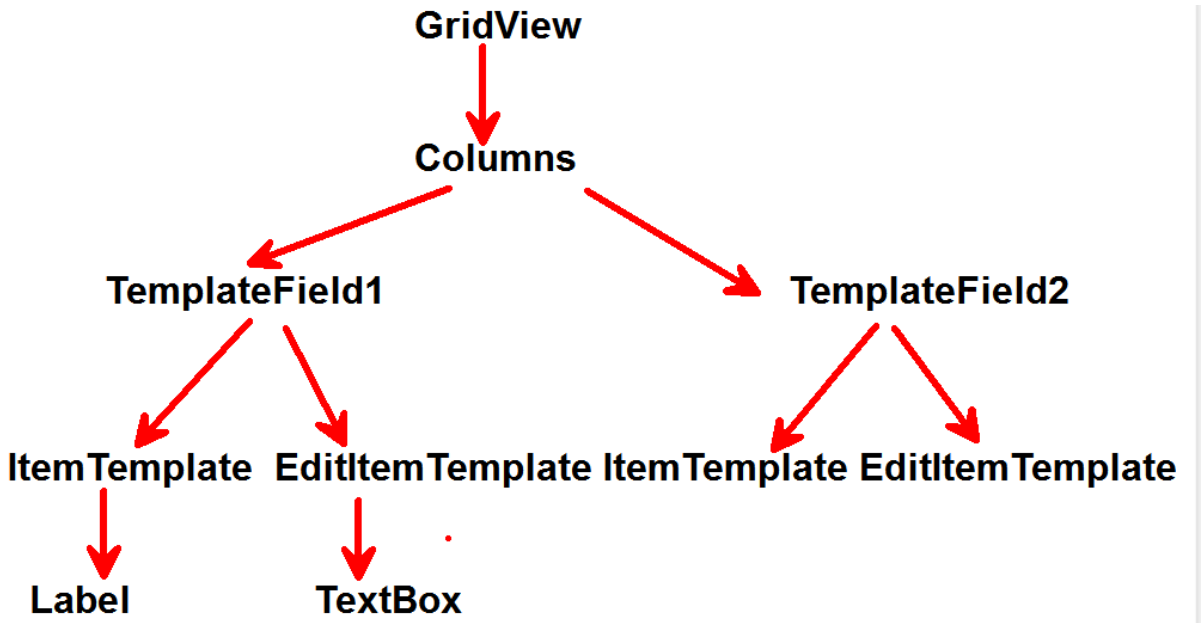
Generate the for Templates:-
select Gridview--->Ellipse--->EditColumns--->
Add 3 BoundFields

	HeaderText	DataField
BoundField1	Empno	eno
BoundField2	Empname	ename
BoundField3	Empsal	salary

select each boundfield and click on ConvertToTemplateField

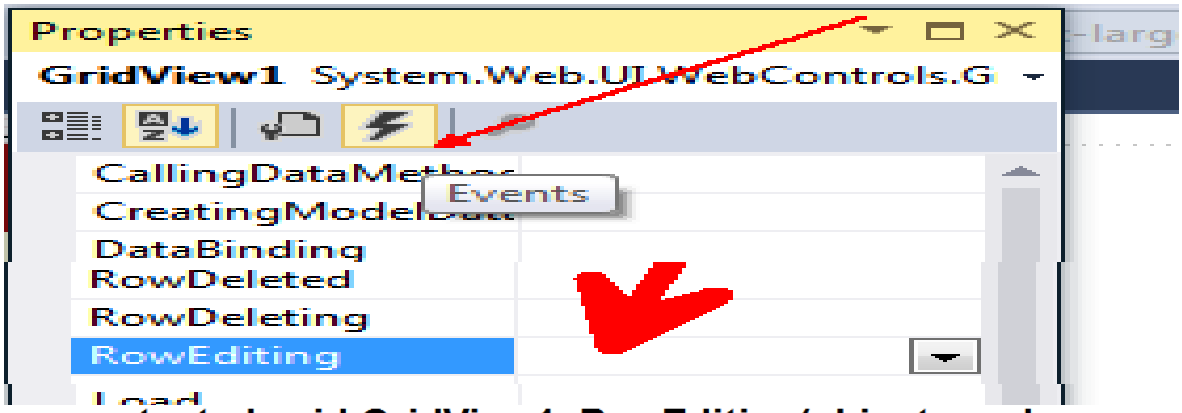
```
using System.Data;
using System.Data.SqlClient;
  public partial class Page1 : System.Web.UI.Page
   {
      private void FillData()
      {
         SqlConnection con = new SqlConnection("
          user id=sa;password=abc;database=demo;
          data source=.");
         SqlDataAdapter da = new SqlDataAdapter("select * from
          emp",con);
         DataSet ds = new DataSet();
         da.Fill(ds,"emp");
         GridView1.DataSource = ds;
         GridView1.DataBind();
      }
    protected void Page_Load(object sender, EventArgs e)
    {
       if (IsPostBack==false)
       {
          FillData();
       }
    }
 }
```

Empno	EmpName	EmpSal
Edit 101	anil	20000.0000
Edit 102	sunil	30000.0000



whenever user clicks on Edit Button then RowEditing Event will fire
whenever user clicks on Update Button then RowUpdating Event will fire
whenever user clicks on Cancel Button then RowCancellingEdit Event will fire
==========
when Gridview is in Editable mode then Edit button was divided into 2 Buttons update and Cancel

rc on Edit Button--->properties--->Events-->Double click on RowEditing Event and write the code

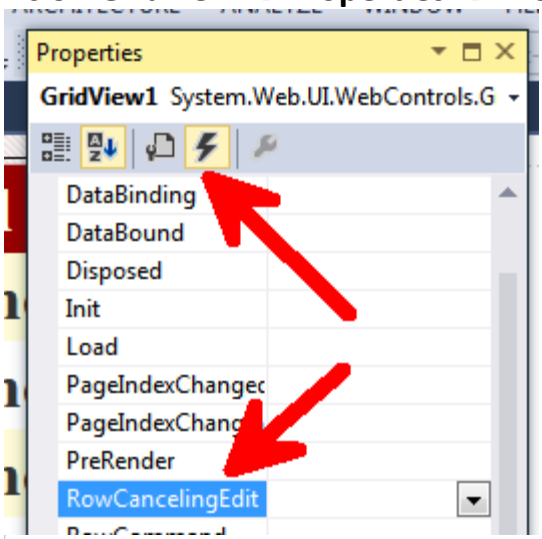



```
protected void GridView1_RowEditing(object sender,
GridViewEditEventArgs e)
{
      GridView1.EditIndex = e.NewEditIndex;
      FillData();
}
```

note:-whenever user clicks on Edit Button then RowEditing Event will fire
e.NewEditIndex will catch the index value of the row where user clicks on Edit Button and display the Rowin Editable mode

	Empno	EmpName	EmpSal
Update Cancel	101	anil	20000.0000
Edit	102	sunil	30000.0000

Rc on Gridview-→Properties-→Events -→RowCancellingEdit




```
protected void GridView1_RowCancelingEdit(object sender,
GridViewCancelEditEventArgs e)
{
        GridView1.EditIndex = -1;
        FillData();
}
```

rc on GridView--->Properties--->Events---->Double click on RowUpdating Event and write the code:-

```
protected void GridView1_RowUpdating(object sender,
GridViewUpdateEventArgs e)
  {
    GridViewRow row = GridView1.Rows[e.RowIndex];
    Control c1=row.FindControl("TextBox1");
    TextBox t1 = (TextBox)c1;
    Control c2 = row.FindControl("TextBox2");
    TextBox t2 = (TextBox)c2;
    Control c3 = row.FindControl("TextBox3");
    TextBox t3 = (TextBox)c3;
    SqlConnection con = new SqlConnection("
   user id=sa;password=abc;database=demo;
   data source=.");
    con.Open();
     string query="update emp set
     ename='"+t2.Text+"',salary='"+t3.Text+"' where

SqlCommand cmd = new SqlCommand(query,con);
cmd.ExecuteNonQuery();
con.Close();
GridView1.EditIndex = -1;
FillData();
}
```

Button Control: - Button Control will have 2 Events

1. Click Event

2. Command Event

Click Event :- This Event will fire whenever user clicks on Button control

for button clicks different Events will fire with Click event

Command Event:-This Event will fire when user clicks on Button Control

for multiple button clicks the same method will be called with Command event

Ex:- Drag and Drop 2 Buttons from Toolbox

select Button1--->properties--->Events--->Double click on Command event

select Button2-->Properties--->Events--->

Command=Button1_Command

```
protected void Button1_Command(object sender,  CommandEventArgs e)
{
        Response.Write("i am Button");
}
```

CommandEventArgs:-it is a predefined class which consists of 2 Properties

1. CommandName

2. CommandArgument

```
public class CommandEventArgs:EventArgs
{
   public object CommandArgument { get; }
   public string CommandName { get; }
}
```

CommandName:-This property is used to set the name for the button control to identify the control

CommandArgument:-This Property is used to catch the RowIndex

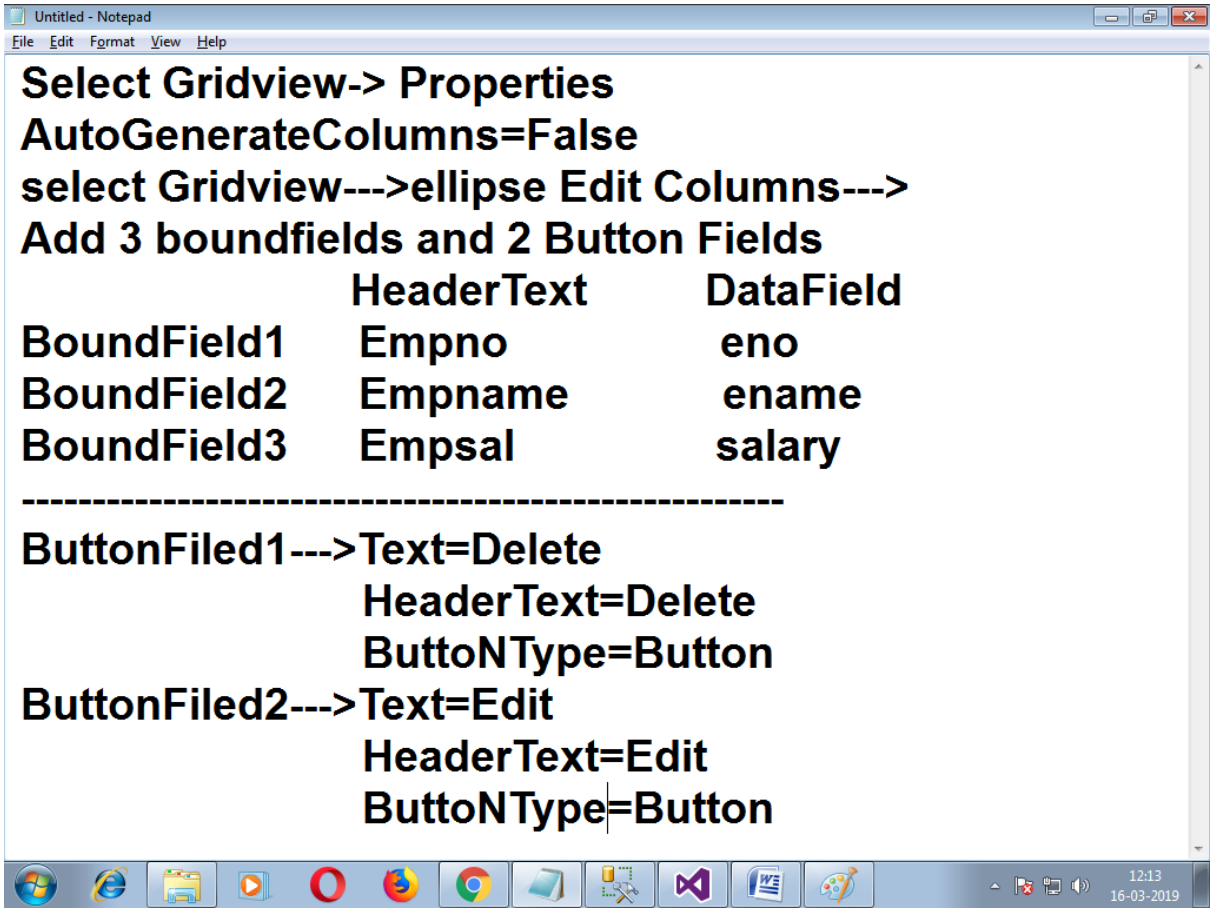
Enter Eno

Enter Ename

Salary

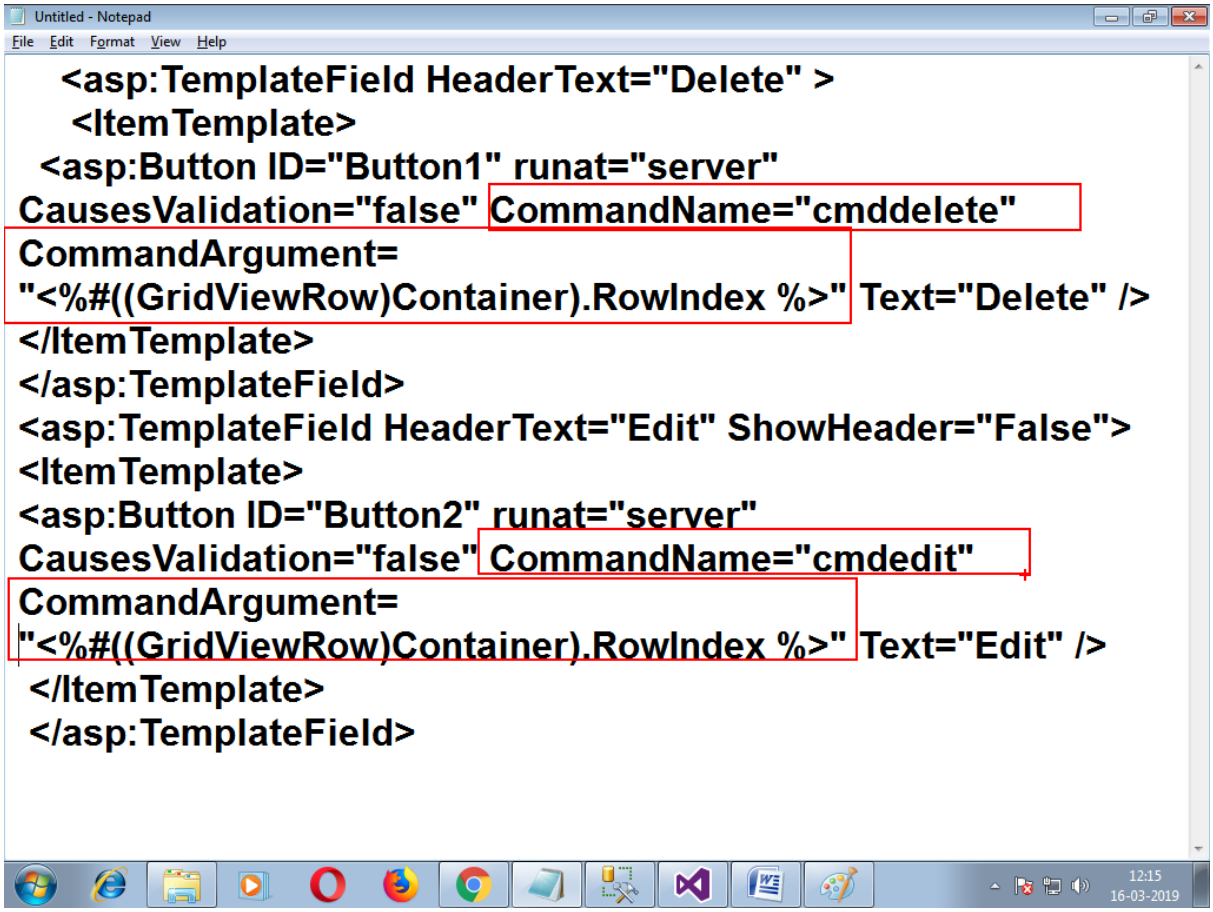
Save

Empno	EmpName	EmpSal	Delete	Edit
101	anilkumar	23000.0000	Delete	Edit
102	sunilkumar	32000.0000	Delete	Edit



Select Buttonfield and BoundField and click on ConvertTo TemplateField

Got and check in source code and modify the Button code



Code for Page1.aspx.cs

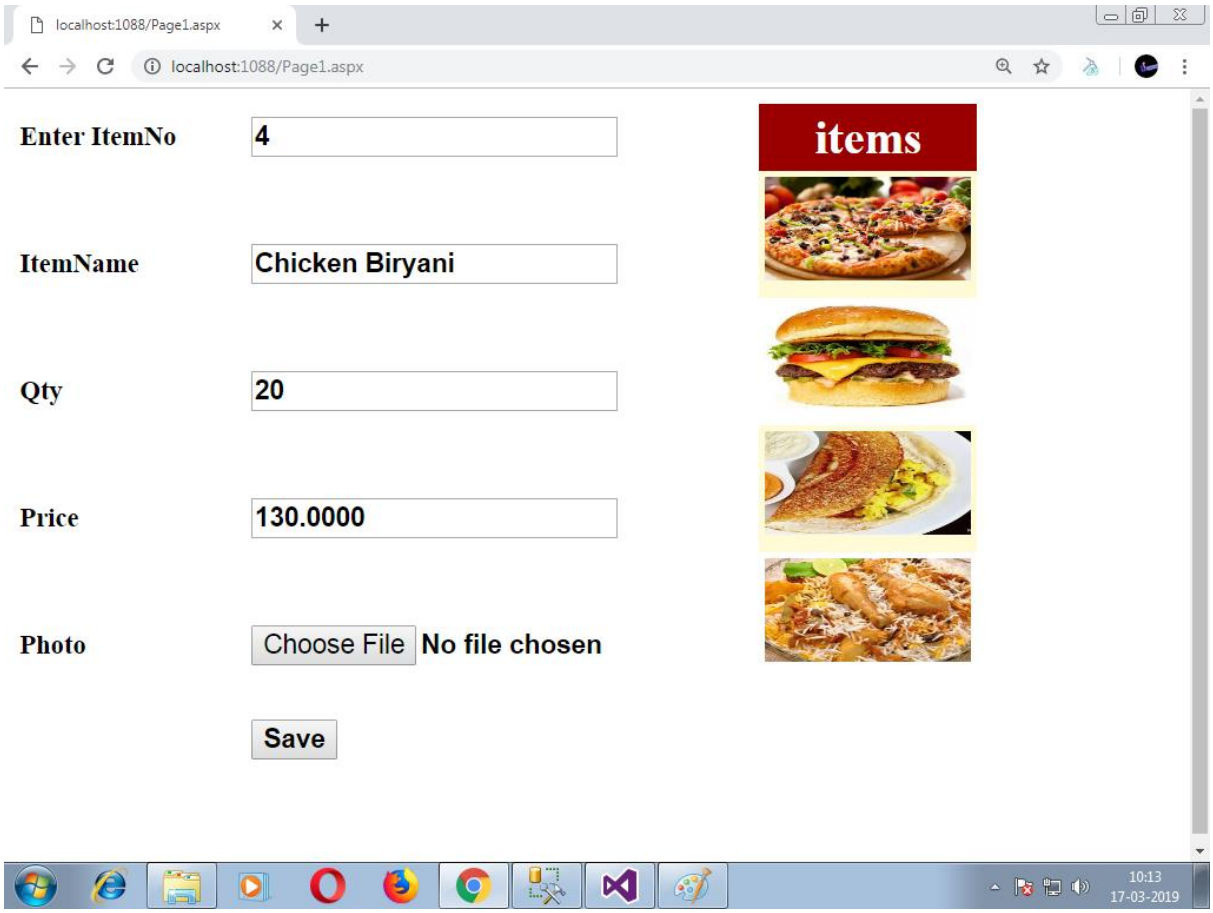
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
namespace WebApplication3
{
   public partial class WebForm1 : System.Web.UI.Page
   {
```

```
    private void FillData()
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        SqlDataAdapter da = new SqlDataAdapter("select * from emp",con);
        DataSet ds = new DataSet();
        da.Fill(ds, "emp");
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
    protected void Page_Load(object sender, EventArgs e)
    {
        if(IsPostBack==false)
        {
            FillData();
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        if (Button1.Text=="Save")
        {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        string query="insert into emp
values('"+TextBox1.Text+"','"+TextBox2.Text+"','"+TextBox3.Text+"')";
        SqlCommand cmd = new SqlCommand(query,con);
        cmd.ExecuteNonQuery();
        con.Close();
        FillData();
        }
        else if (Button1.Text == "Update")
        {
            SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
            con.Open();
```

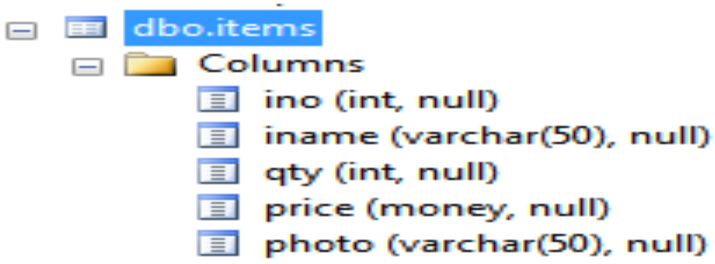
```
            string query = "update emp set
ename='"+TextBox2.Text+"',salary='"+TextBox3.Text+"' where
eno='"+TextBox1.Text+"'";
            SqlCommand cmd = new SqlCommand(query, con);
            cmd.ExecuteNonQuery();
            con.Close();
            FillData();
            TextBox1.Text = "";
            TextBox2.Text = "";
            TextBox3.Text = "";
            Button1.Text = "Save";
        }
    }

    protected void GridView1_RowCommand(object sender,
GridViewCommandEventArgs e)
    {
        if (e.CommandName=="cmddelete")
        {
            int index=Convert.ToInt32(e.CommandArgument);
          GridViewRow row=GridView1.Rows[index];
          Label l1 = (Label)row.FindControl("Label1");
          SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
          con.Open();
           string query="delete from emp where eno='"+l1.Text+"'";
          SqlCommand cmd = new SqlCommand(query,con);
          cmd.ExecuteNonQuery();
          con.Close();
          FillData();
        }
        else if (e.CommandName=="cmdedit")
        {
            int index = Convert.ToInt32(e.CommandArgument);
            GridViewRow row = GridView1.Rows[index];
            Label l1 = (Label)row.FindControl("Label1");
            Label l2 = (Label)row.FindControl("Label2");
```

```
            Label l3 = (Label)row.FindControl("Label3");
            TextBox1.Text = l1.Text;
            TextBox2.Text = l2.Text;
            TextBox3.Text = l3.Text;
            Button1.Text = "Update";
        }
    }
  }
}
```

Example to Display Image in Gridview



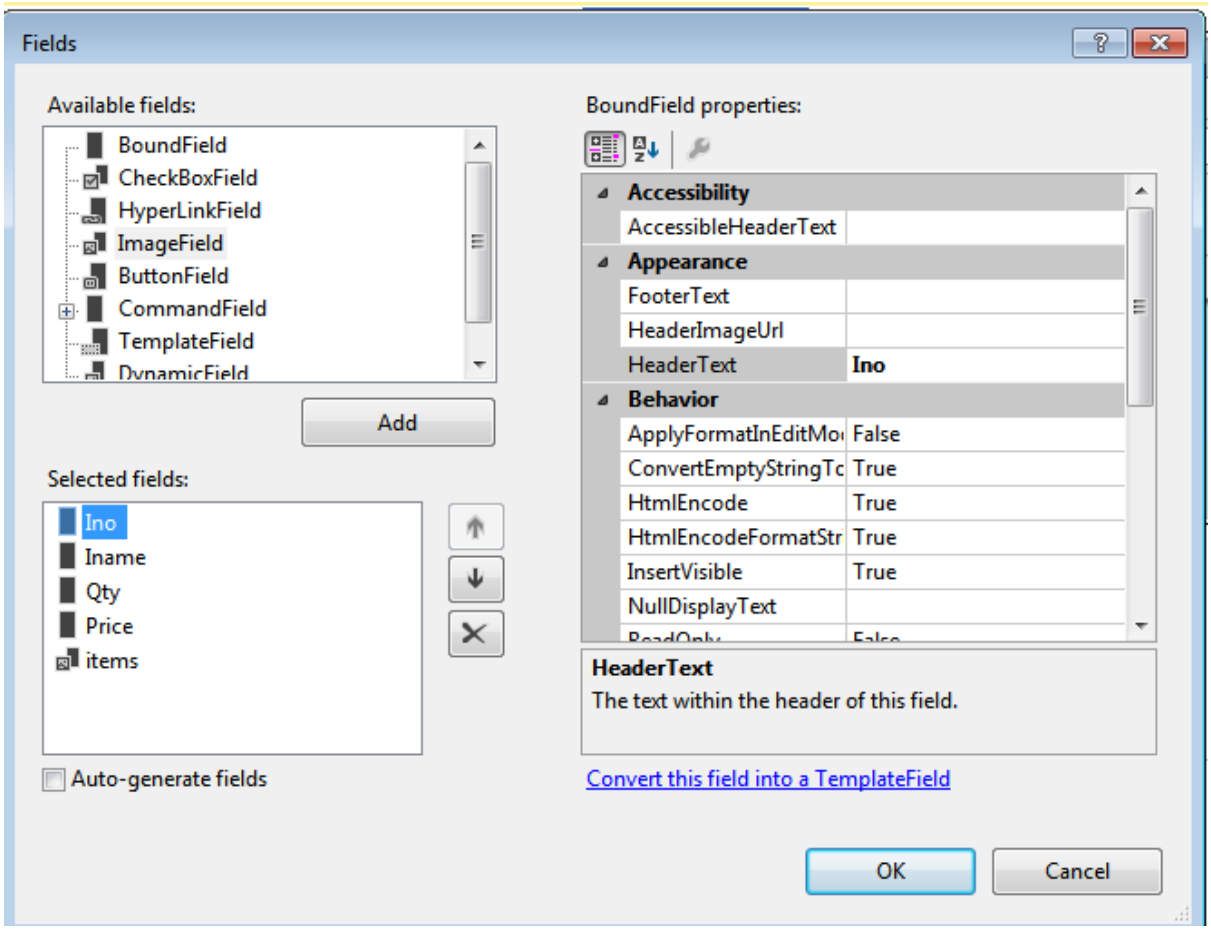
ino	iname	qty	price	photo
1	Pizaa	20	120....	~/Images/p1.jpg
NULL	NULL	NULL	NULL	NULL



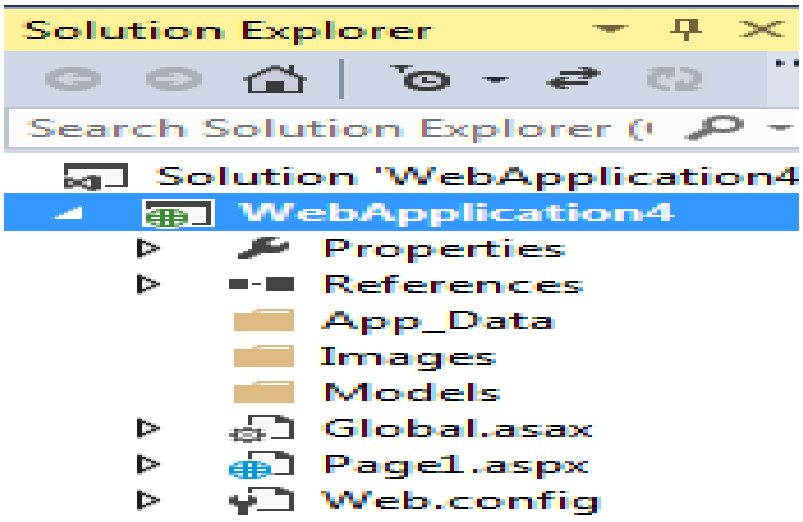
select Gridview--->Properties-->
AutoGenerateColumns=False
select Gridview--->Ellipse--->EditColumns--->
Add 4 boundFields and 1 imageField

	HeaderText	DataField	Visible
BoundField1	ino	ino	False
BoundField2	iname	iname	False
BoundField3	qty	qty	False
BoundField4	price	price	False

ImageField --->DataImageUrlField=photo
select BoundField and ImageField --->Click on ConvertToTemplateField



Goto-→SolutionExplorer-→rc on project-→Add NewFolder-→rename=Images



Goto-→Web.config file

```
 </system.web>
 <connectionStrings>
  <add name="constr" connectionString="user
id=sa;password=abc;database=kbc;data source=."/>
 </connectionStrings>
```

Goto→source code and modify the code:-

```
<ItemTemplate>
 <asp:ImageButton ID="Image1" runat="server"
ImageUrl='<%# Eval("photo") %>'
 Height="70" Width="140"
CommandArgument=
"<%#((GridViewRow)Container).RowIndex%>" />
</ItemTemplate>
</asp:TemplateField>
```

convert image to image Button

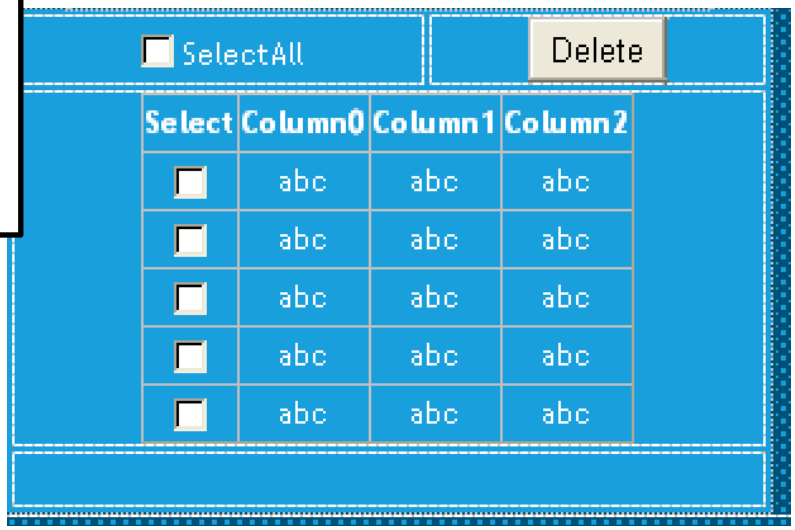
code for Page1.aspx.cs:-

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
```

```
namespace WebApplication4
{
   public partial class Page1 : System.Web.UI.Page
   {

     private void FillData()
      {
         SqlConnection con=new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].
ToString());
         string query="select * from items";
         SqlDataAdapter da = new SqlDataAdapter(query,con);
         DataSet ds = new DataSet();
         da.Fill(ds, "items");
         GridView1.DataSource = ds;
         GridView1.DataBind();
      }
      protected void Page_Load(object sender, EventArgs e)
      {
         if (IsPostBack==false)
         {
            FillData();
         }
      }
      protected void Button1_Click(object sender, EventArgs e)
      {
         string fname=FileUpload1.FileName;
         string fpath="~/Images/"+fname;
         FileUpload1.SaveAs(Server.MapPath(fpath));
         SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].
ToString());
         con.Open();
         string query = "insert into items
values('"+TextBox1.Text+"','"+TextBox2.Text+"','"+TextBox3.Text+"','"+TextBox
4.Text+"','"+fpath+"')";
         SqlCommand cmd=new SqlCommand(query,con);
```

```
        cmd.ExecuteNonQuery();
        con.Close();
        FillData();
    }


    protected void GridView1_RowCommand(object sender,
GridViewCommandEventArgs e)
    {
        int index = Convert.ToInt32(e.CommandArgument);
        GridViewRow row = GridView1.Rows[index];
       Label l1=(Label)row.FindControl("Label1");
       Label l2 = (Label)row.FindControl("Label2");
       Label l3 = (Label)row.FindControl("Label3");
       Label l4 = (Label)row.FindControl("Label4");
       TextBox1.Text = l1.Text;
       TextBox2.Text = l2.Text;
       TextBox3.Text = l3.Text;
       TextBox4.Text = l4.Text;
    }
}
```

Gridview CheckBoxes:-



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
public partial class Case59 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            filldata();
        }
    }
    public void filldata()
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        SqlDataAdapter da = new SqlDataAdapter("select * from emp", con);
        DataSet ds = new DataSet();
        da.Fill(ds, "emp");
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
    protected void Button1_Click(object sender, EventArgs e)
    {

        foreach (GridViewRow row in GridView1.Rows)
        {
            var chk = row.FindControl("CheckBox1") as CheckBox;
            if (chk.Checked)
            {
                var lblid = row.FindControl("Label1") as Label;
```

```
            Response.Write(lblid.Text + "<br>");
            SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
            con.Open();
            int id=Convert.ToInt32(lblid.Text);
            string query = " delete from emp where empno='"+ id+"'    ";
            SqlCommand cmd = new SqlCommand(query, con);
            cmd.ExecuteNonQuery();
            con.Close();
            Response.Write("empno is deleted ");
            filldata();       }      }      }
//select all query.....Delete

protected void chkselect_CheckedChanged(object sender, EventArgs e)
   {
      if (chkselect.Checked == true)
      {
         foreach (GridViewRow row1 in GridView1.Rows)
         {
            var chk1 = row1.FindControl("CheckBox1") as CheckBox;
           // chk1.Enabled = true;
            bool allcheckbox = true;
            chk1.Checked = allcheckbox;
            if (chk1.Checked )
            {
               var lblid = row1.FindControl("Label1") as Label;
               Response.Write(lblid.Text + "<br>");
               SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
               con.Open();
               int id = Convert.ToInt32(lblid.Text);
               string query = " delete from emp where empno='" + id + "'    ";
               SqlCommand cmd = new SqlCommand(query, con);
               cmd.ExecuteNonQuery();
               con.Close();
               Response.Write("empno is deleted ");
              filldata();           }          }      }   }   }
```

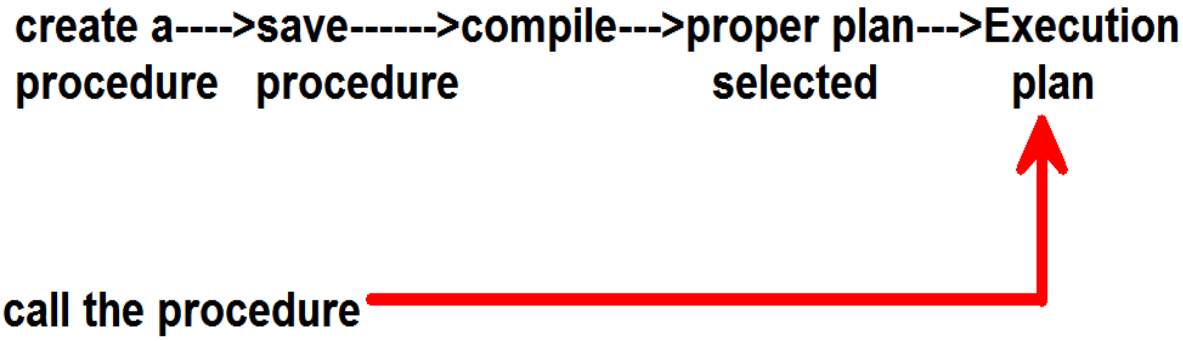
StoredProcedure:-StoredProcedure is a set of precompiled sqlstatements which will gets executed when we call it whenever we write any sqlquery 3 types of operations will done:-

1. write the query
2. query will gets compiled
3. proper plan is selected to execute the query
4. query will gets executed

Each and every time when we execute the query 2,3,4 will gets executed

Working with Storedprocedures:-

1. create the procedure
2. call the procedure



syn to create procedure:-

```
create procedure procedurename(parameters)
as begin
   sql statements
end
```

syn to call the procedure :-

```
exec procedurename
```

At the time of creating the procedure we have to declare parameters and at the time of calling the procedure we have to pass values

The no,order,type of values that we pass must match with no,order,type of parameters

```
create database demo
use demo
create table emp(eno int,ename varchar(50),
salary money)
```

```
create procedure proc_addemp(@eno int,
@ename varchar(50),@salary money)
as begin
insert into emp values(@eno,@ename,
@salary)
End
```

```
create procedure proc_deleteemp(@eno int)
as begin
delete from emp where eno=@eno
end
```

```
create procedure proc_getemp
as begin
select * from emp
end
```

```
create procedure proc_searchemp(@eno int)
as begin
select * from emp where eno=@eno
end
```

```
create procedure proc_updateemp(@eno int,
@ename varchar(50),@salary money)
as begin
update emp set ename=@ename,salary=@salary
where eno=@eno
end
```

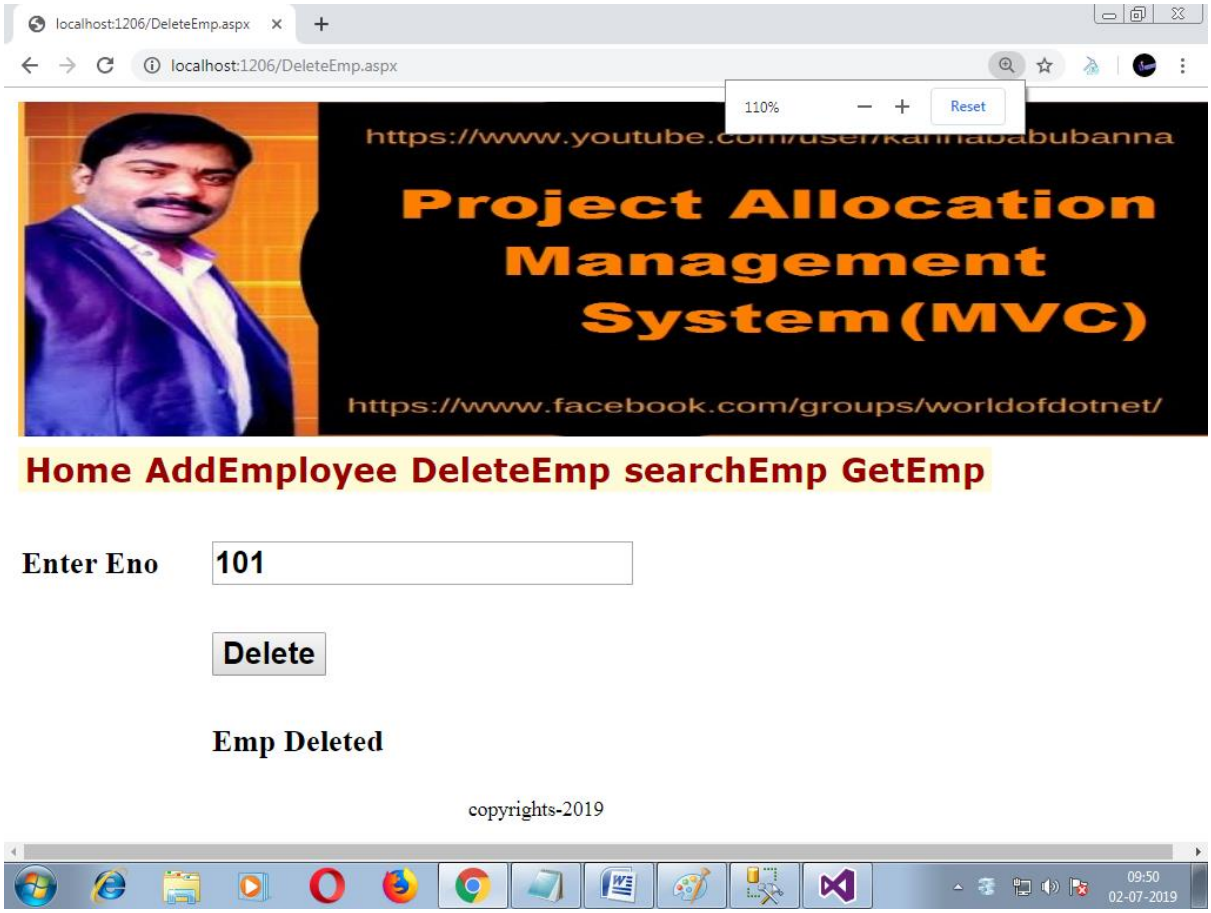
declare connectionstring globally in web.config:-

```
</system.web>
  <connectionStrings>
    <add name="constr" connectionString="user
id=sa;password=abc;database=demo;data source=."/>
  </connectionStrings>
```

AddEmp.aspx.cs:-

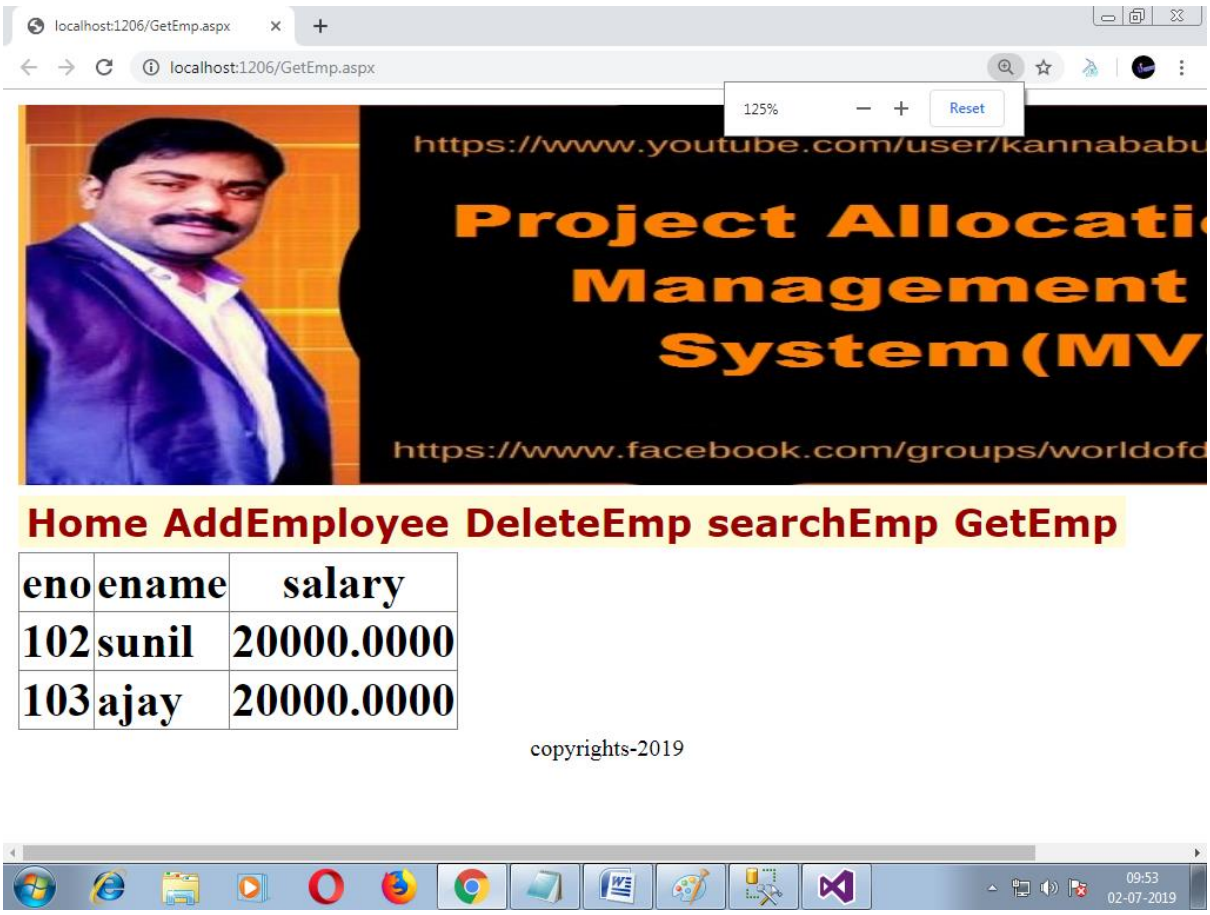
```
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

  protected void Button1_Click(object sender, EventArgs e)
     {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        //pass the procedure
        SqlCommand cmd = new SqlCommand("proc_addemp",con);
        //mention that we are working with SP
        cmd.CommandType = CommandType.StoredProcedure;
        //pass the values to parameters
        cmd.Parameters.AddWithValue("@eno", TextBox1.Text);
        cmd.Parameters.AddWithValue("@ename", TextBox2.Text);
        cmd.Parameters.AddWithValue("@salary", TextBox3.Text);
        //execute the procedure
       int i=cmd.ExecuteNonQuery();
```

```
        con.Close();
        if (i==1)
        {
           Label1.Text = "Emp added";
        }
      }
```




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

    protected void Button1_Click(object sender, EventArgs e)
     {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        SqlCommand cmd = new SqlCommand("proc_deleteemp",con);
```

```
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@eno", TextBox1.Text);
       int i= cmd.ExecuteNonQuery();
        con.Close();
        if (i==1)
        {
           Label1.Text = "Emp Deleted";
        }
    }
```




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

protected void Page_Load(object sender, EventArgs e)
    {
```

```
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        SqlDataAdapter da = new SqlDataAdapter("proc_getemp",con);
        DataSet ds = new DataSet();
        da.Fill(ds, "emp");
        GridView1.DataSource = ds;          GridView1.DataBind();       }
```

Statemanagement Techniques

B.Kannababu
(Sathyatechnologies)

State Management Techniques:

Q)What is WebBrowser?

A **web browser** (commonly referred to as a **browser**) is a software application for accessing information on the World Wide **Web**.

Eg:Internet Explorer,Mozilla Firefox,Googlechrome etc...

Q)What is WebServer?

Web server refers to server software that can serve contents to the World Wide Web.

Eg:Asp.net Development Server

IIS(Internet Information services)

Apache

Whenever we develop and webapplication we have to deploy the application on webserver.

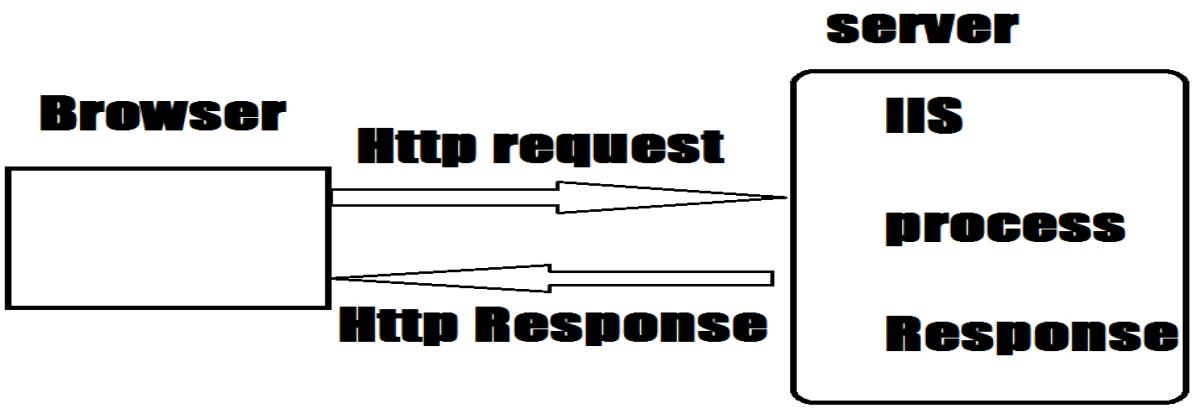
So that end user will access the application via webserver.

We need to follow 2 types:-

1.We need to purchase Domainname

2.We need to purchase space on webserver.

Client Server Architecture:--



1.Client will send request to server.

2.Webserver will accept the request,process and generate the response and forward the response back to Browser.

3.The communication between client and server is due to http protocol.

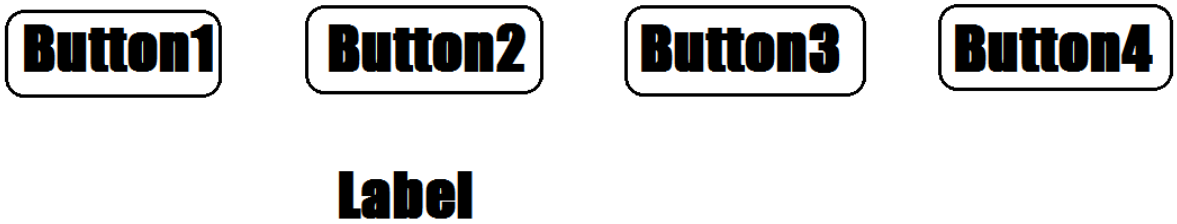
4.Http is a communication protocol which is used to create communication between client and server.

Http is responsible to send request to server and it is responsible to get response from server.

Http is a stateless protocol.

i.e. it will not maintain the state of the response.

Eg:-



```
public Partial class page1:System.Web.Ui.Page
{
int x;
Protected void Button1_click()
{
X=X+5000
Label1.Text=X.ToString();
}
Protected void Button2_click()
{
X=X+3000
Label1.Text=X.ToString();
}
Protected void Button3_click(){
X=X+1000
Label1.Text=X.ToString();
}
Protected void Button4_click(){
X=X+10
Label1.Text=X.ToString();
}}
```

OBSERVATION:In the above program when user clicks on Button1 request 1 will go to server response is 5000.when user clicks on Button2 request will go to server when req2 is going to server the previous reponse is destroyed so final output is 10.

--->Asp.net is a technology which works on Http protocol.Http is a stateless protocol.i.e will not maintain state of the response either on Browser or on server as Asp.net works on Http protocol.Asp.net is also called as stateless Technology, In order to maintain the state of the response either on Browser or web server we have to use State Management Techniques.

StateManagementTechniques:--It is a mechanism of maintaining the state of the response either on Browser or on Web Server.

StateManagementTechniques are two types:--

1.Client side State Management Technique.

2.Server side State Management Technique.

1.<u>Client side State Management Technique</u>:-It is a mechanism of maintaining the state of the Response on Browser.

Different Types of Client side State Management Techniques are:--

1.ViewState

2.QueryString

3.HiddenFields

4.Cookies

<u>Server side State Management Technique</u>:--It is a mechanism of maintaining the state of the response on web server.

Different Types of Server side State Management Techniques are:--

1.Sessions

2.Application

3.Caching

<u>VIEW STATE:-</u>

points to Remember:-

1. viewstate is client side state management Technique which is used to maintain the state of the response on Browser.

2. The scope of viewstate is within the page i.e we cannot maintain the values between multiple webpages.

3. Viewstate will maintain the data in encrypted format within the browser,any hacker can convert encrypted format in to plain text format so it is not recomended to store sensitive information like password in viewstate

4. Viewstate will automatically retain values of the controls during the postbacks of the same page

5. All the control values will store in Viewstate before Page_Load event

6. if EnableViewstate property is set to False then Viewstate will not maintain the values

7. As Viewstate will maintain the value in Browser it will reduce the burden on server.

8. we can store only limited amount of data in Viewstate the maximum capacity of the browser is 4kb(Internet Explorer) so we cannot store huge amount of data in viewstate.

9. we can store object in viewstate but we have to declare the class with [Serializable] attribute.

10. if the page consits of more controls then viewstate might effect the performance on the Page.

Syntax to store the value in ViewState:-

ViewState[”varname”]=value

Syntax to read the value from viewstate:--

View State[”varname];

By default view state will maintain the data in the form of object.

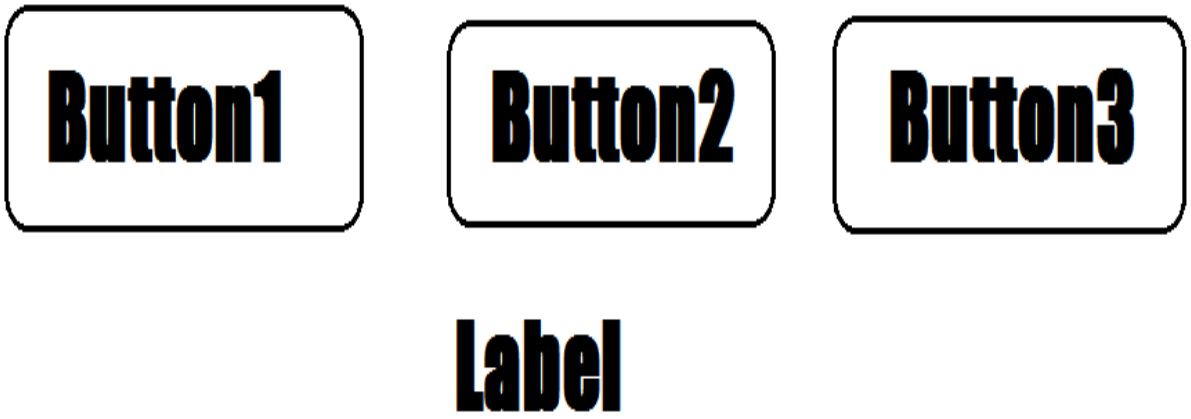
So we can store any type of value in viewstate.

Object a=10;

int i=(int)a;

Eg:-

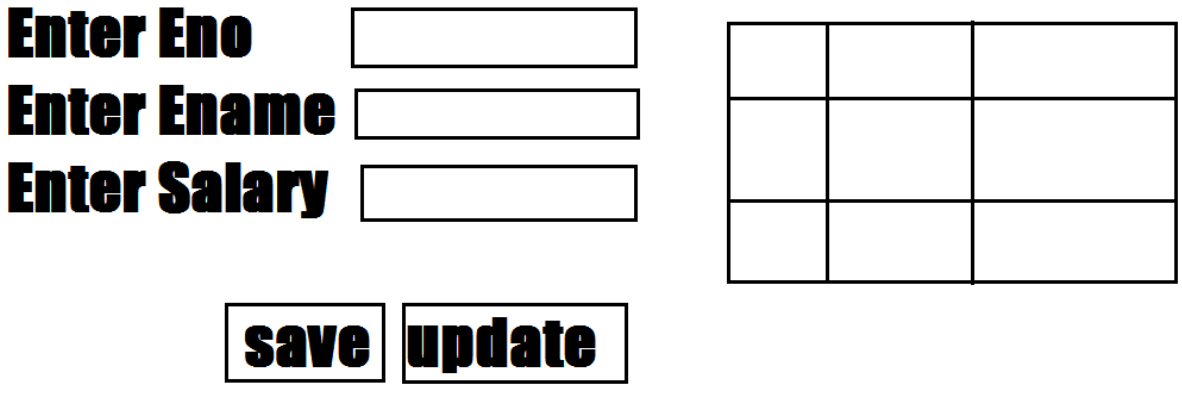

```
Protected void Button1_click()
{  ViewState["x']="SathyaTech";
Label1.Text="Value is stored in viewstate";
}
Protected void  Button2_click()
{  Label1.Text=ViewState["x].ToString();  }
```




```
public Partial class WebForm1:System.Web.Ui.Page
{
int x=0;
Protected void Button1_click(){
x=x+5000;
ViewState["x"]=x;
Label1.Text=X.ToString[];
```

```
}
protected void Button2_click(){
x=(int)ViewState["x"];
x=x+1000;
ViewState["x"]=x;
Label1.Text=x.ToString();
}
protected void Button3_click(){
x=(int)ViewState["x"];
x=x+10;
ViewState["x"]=x;
Label1.Text=x.ToString();
}
}
```

Click on the save Button will display the data in Grid View without using database using dataset:-



Requirement:- whenever user clicks on save button store the details in DataSet whenever user clicks on update button update the data from Dataset to Database


```
Using System.Data;
public Partial class page1:System.Web.Ui.Page
{
```

```
DataSet ds; DataTable=dt; DataRow dr;
Protected void Page_Load(){
if(IsPostBack==false)
{
        //create DataSet
    ds=new DataSet();
        //create DataTable
     dt=new DataTable();
        //design the name for data table
    dt.TableName="emp";
        //create columns
   dt.columns.Add("eno");
   dt.columns.Add("ename");
   dt.columns.Add("salary");
   ViewState["mydt"]=dt;
   ViewState["myds"]=ds;
}}
protected void Button1_click()
{
            dt=(DataTable)ViewState["mydt"];
            ds=(DataSet)ViewState["myds"];
                //create a new row
            dr=dt.NewRow();
                //insert the record in New Row
            dr[0]=int.Parse(TextBox1.Text);
            dr[1]=TextBox2.Text;
            dr[2]=double.Parse(TextBox3.Text);
                //add the row to Data Table
            dt  .Rows.Add(dr);
                //Merge DataTable to DataSet
          ds.Merge(dt);
            dr.delete();
```

```
//Display data in GridView
GridView.DataSource=ds;
GridView1.DataBind();
```

ick on Update Button DataSet Table Data is stored in DataBase By Using ommandBuilder:

ode for Update Button:--

```
otected void Button2_click()

dt=(DataTable)ViewState["mydt"];
ds=(DataSet)ViewState["myds"];
SqlConnection con=new SqlConnection();
SqlDataAdapter ds=new SqlDataAdapter("select * from emp",con);
SqlCommand Builder cb=new SqlCommandBuilder(da);
la.Update(ds,"emp");
```


ueryString:

Querystring will maintain the values between one page to another page.

ynatx to pass the value in QueryString:

```
Reponse.Redirect("destinationurl? QueryStringname=value");
```

yntax to Read the value from QueryString:

```
Request.QueryString["querystringname"]
```

Enter Username

Button

Code for page1.aspx.cs:--

```
 Protected void Button1_click)
{
Respose.Redirect("page2.aspx? uname='"+TextBox1.Text+"'");
}
```

code for page2.aspx.cs:--

```
 Protected void Page_Load()
{
Label1.Text="Welcome "+Request.QueryString[0];
}
```

Points To Remember:

1.QueryString will maintain the values from one page to another page only.

2.QueryString will display the data within the Browser url so it is not recommended to send sensitive information like passwords using query string.

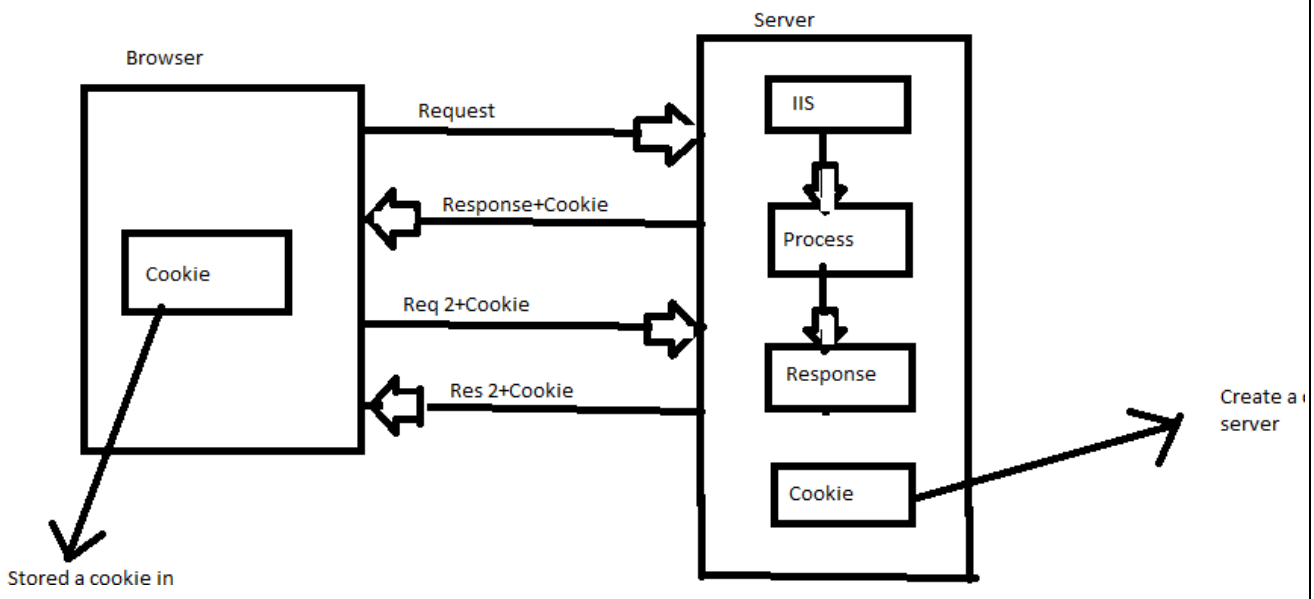
3.Querystring will send the data via browser url so the maximum capacity of the browser is 4kb we cannot send more than 4kb of information using querystring.

Q)How to hide the query string data within browser url?

By using Server. Transfer();

COOKIES:

Cookie is a client side statemanagement Technique which is used to maintain the reponse on browser in plain text format.



1.Whenever client sends the request IIS will accept the request,process and generate the response and store the response in cookie.

2.Along with res1+cookie will travel to Browser.

3.Along with req2+cookie will travel to server.

4.i.e between multiple client requests and responses cookie will travel between client and server and finally cookie will store on Browser in the form of plainText format.

Q)Where Cookie is created?

Web Server

Q)Where cookie is stored?

Browser

Q)What is the scope of cookie?

Throughout the website.

Q)What is the default Expiry timeout of cookie?

30 Minutes.

Q)What are Different Types of Cookies?

1.Temporary cookie (or) inmemory cookie(or) Non PersistentCookie.

2.Permanent cookie (or) outmemory cookie(or) PersistentCookie.

Steps To work With Cookie:

1.Create a cookie by using Httpcookie class

```
Httpcookie obj=new Httpcookie("cookiename");
```

2.Store the value in cookie object:

```
Cookieobjectname.value=Somevalue;
    obj.value="Sathya Tech";
```

3.Add the cookie object to cookies collection table:

```
Response.cookies.Add(cookiename);
```

Syntax to Read the value from cookie:-

```
Request.cookies["cookiename"].value;
```


```
Protected void Button1_click(){
   HttpCookie obj=new HttpCookie("name");
      obj.value="SathyaTechnologies";
  Response.Cookies.Add(obj);
  Label1.Text="value is stored in cookie";
}
```

```
Protected void Button2_click()
{
  Label1.Text=Request.cookies["name"].value;
}
```

Difference Between Inmemory cookie and outmemory:-

In memory	Out memory
1.Cookie value will store on RAM.	1.Cookie value will store on Harddisk.
2.The life span of In memory cookie Until user close Browser.	2.The life span of Outmemory cookie after expiry Time.
3.We cannot set expiry time for in memory cookie	3.We can set expiry time for outmemory cookie
4.In memory cookie is also called as Temporary called non persistent cookie.	4.Out memory is also as permanent cookie(or) (or)Persistent cookie.

Setting Expiry Time for persistent Cookie:--

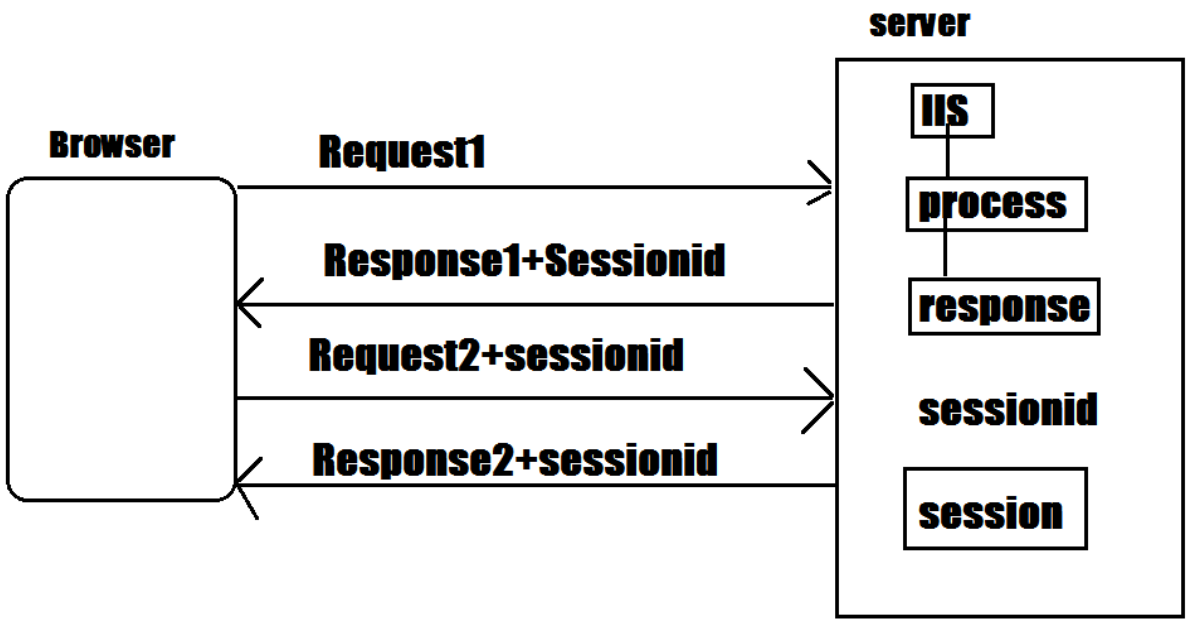
```
 Protected void Button1_click()
{
    HttpCookie obj=new HttpCookie("name");
       obj.value="SathyaTechnologies";
    obj.Expires=System.Data.Now.AddHours(5);
   Response.Cookies.Add(obj);
   Label1.Text="value is stored in cookie";
}
```

Serverside StateManagement Techniques:---It is a mechanism of maintaining the state of the response on webserver.

Different Types of serverside Techiniques are:

1.Session
2.Application
3.Caching

Session:Session is a server side statemanagement technique which is used to maintain the data on webserver within a particular time period between login and logout.



1.Whenever client sends the request webserver will accept the request,process and generate the response apart from that a session is created on webserver and that session was assigned with some id i.e session id.
2.Along with response1+sessionid will travel to Browser.
3.Between multiple client requests and responses sessionid will travel between client and server and finally sessionid is stored on Browser in the form of coockie.

Q)Where session is created?
On Webserver

Q)Where Sessionid is created?
OnWebServer.

Q)Where sessionid is stored?
on webserver in the form of cookie.

Q)what will happen when we delete cookies?
Sessionid will delete so we cannot access session object.

Q)How to kill the session?
Session.Abandon()

Q)What is the default expiry time of session?
20 Minutes

Q)What are the Different session state modes?
1.inproc
2.Off
3.SqlServer
4.StateServer
5.Custom

ASP.net Page Life Cycle Execution:-

1. Application level Events (Global.asax)
Application_Start
Session_Start
Application_Error
Session_End
Application_End
2. Page Level Events(webform1.aspx.cs)
Page_PreInit
Page_Init
Page_InitComplete
Page_PreLoad
Page_Load
Page_LoadComplete
Page_PreRender
Page_PreRenderComplete
Page_Unload
3. Control Level Events(webform1.aspx.cs)

1. whenever we develop any webapplication we have to host the webapplication on webserver otherwise the enduser cannot access webapplication
   in order to host the webappn on webserver(IIS)
   we need to buy the space on remote server
2. whenever client sends request IIS will accept the request
   and check whether the request is ASP.net request or not
   based on the extension
3. if the request is ASP.net request then IIS will launch
   ISAPI.dll file
   internet server application programming interface
4. ISAPI.dll file is responible to launch w3wp.exe file
   w3wp.exe file is worker process which will under IIS
   This process is reposnible for processing ASP.net request
   Asp.net webapplication will run within this process
   note:- a webfarm consits of multiple worker process
   Every server may have worker process or a group of
   servers may have a single worker process
   workerprocess is a seperate memory that was allocated
   on server
5. within the worker process for every webaapplication a seperate memory is allocated which is called as AppDomain
   1 webapplication =1 AppDomain
   multiple AppDomains may exist in worker process
   within the Appdomain for every request a seperate object(thread) is created
6. CLR will execute the code that is available within the AppDomain and different pagelevel events will fire
   and finally output is rendered to browser in the form of HTML

Q)**what is Rendering?**

Rendering means converting server side code into client side code

1. Page_Preinit:- This event is used to set master page Dynamically
   This event will check whether IsPostBack property
   i.e it will check whether the page is processed for
   firsttime or not

it is used create the controls dynamically
it is used to apply themes and skins dynamically

2. Page_Init:- This event will fire after each control has been initialized
in this event a unique id was set for every control

3. Page_InitComplete:- until now the viewstate values are not loaded hence this event is used to make changes to
the viewstate

4. PreLoad:-This event will raised after the page loads the viewstate i.e befor PageLoad wil fire all the control values are stored in viewstate

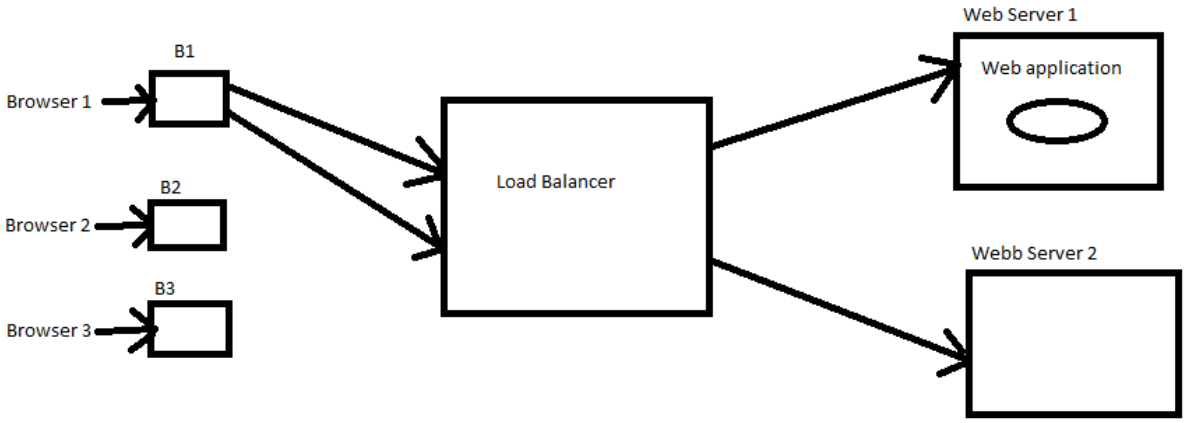
5. Page_Load:- this event will fire whenever the page is postabck to server

6. Prerender:-in this event the response is generated
and the server side code is converted into Html code and render to Browser

7. Page_UnLoad:- This event is used to destroy the objects
Garbage collector will destroy unreferenced objects

Q)what is webfarm?

webfarm means webapplication deployed on multiple webservers



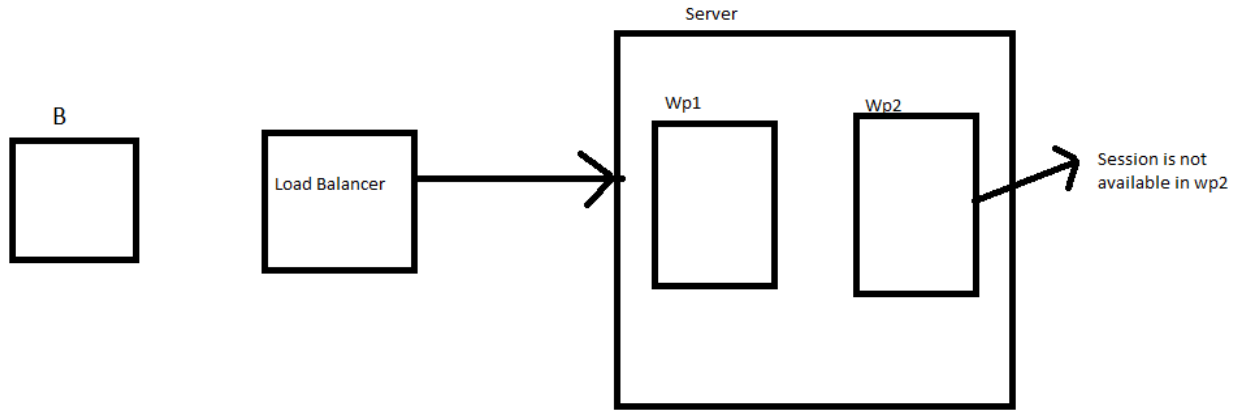
Q)what is webgarden?

Webappn deployeed on multiple worker process within

the same webserver

session state mode="off":- when session state mode is off

sessions will not work



Sessionstate modes:-

1. **Inproc**

2. **Sqlserver**

3. **Stateserver**

4. **Off**

5. **Custom**

Inproc:

session state mode="inproc":-This is the Default session state mode

Q)where to declare the session state mode?

web.config file

<sessionstate mode="inproc" timeout="2">

</sessionstate>

whenever we set sessionstate mode="inproc" then session informataion is stored within the workerprocess of local server

Advantages of Inproc mode:-

1. inproc is very easy to implement just declaring sessionstate mode="inproc" in web.config file

2. performance is good because session information is stored within The Appdomain of local server

3. inproc mode is recomended for the websites where the no of users are less

4. inproc mode is recomended for webappns which was hosted on single server

5. in this mode we can directly store the objects without serialization

Limitations of inproc mode:-

1. session data is lost when the worker process is restarted or when we do any modification in web.config.

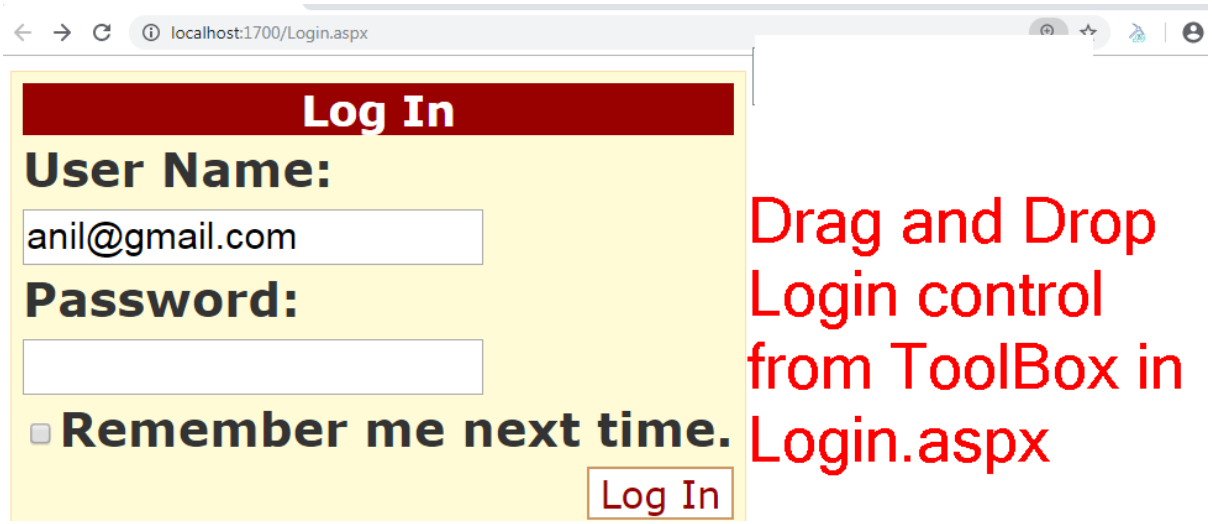
2. inproc mode is not suitable for webgarden and webfarm.

3. scalability could be an issue.

4. it is not recomended for websites where no of users are more.

```
use Demo
create table login(username varchar(50),password varchar(50))
```

username	password
anil@gmail.com	anil
sunil@gmail.com	sunil



Drag and Drop Label control in Success.aspx

Goto-→web.config file:-

```
<system.web>
  <compilation debug="true" targetFramework="4.5.1" />
  <sessionState mode="InProc" timeout="2"></sessionState>
 </system.web>

 <connectionStrings>
  <add name="constr" connectionString="user
id=sa;password=abc;database=demo;data source=."/>
 </connectionStrings>
```

Double click on Login control and write the code:-

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

    protected void Login1_Authenticate1(object sender, AuthenticateEventArgs
e)
     {
```

```
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        string query = "select * from login where username='" +
Login1.UserName + "' and password='" + Login1.Password + "'";
        SqlCommand cmd = new SqlCommand(query, con);
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
           if (dr.Read())
           {
              Session["uname"] = Login1.UserName;
              Response.Redirect("Success.aspx");
           }
        }
        else
        {
 Login1.FailureText = "invalid user";
        }
     }
```

Code for success.aspx.cs:-

```
  protected void Page_Load(object sender, EventArgs e)
     {
        if (Session["uname"]!=null)
        {
           Label1.Text = "Welcome " + Session["uname"];

        }
        else
        {
           Response.Redirect("Login.aspx");
        }
     }
```

session state mode="off":- when session state mode is off

sessions will not work

Q)How to Disable the Session?

we can Disable the session in 2 ways

1. At PageLevel
2. At Website Level

Q)How to Disable the session at PageLevel?

By declaring EnableSessionState=False in

page1.aspx[source]

Q)How to Disable the session at website level?

By declaring session state mode=Off in web.config file .

Sql Server:--

Session state mode="sqlserver" in this mode session information will store in sql server database.Session information like session id,timeout,Expiry etc..

Advantages of Sql Servermode:--

1.It supports webfarm and webgarden.

2.Scalability is good.

3.Realibilitty is good.

4.Security is good.

DisAdvantages:-

1. performance is poor when compare with in proc mode.

2. in sqlserver mode if we want to store session information in sqlserver database then we need to purchase the license of sql server.

3. we need to serialize and deserialize when we store objects in session in sqlserver mode.

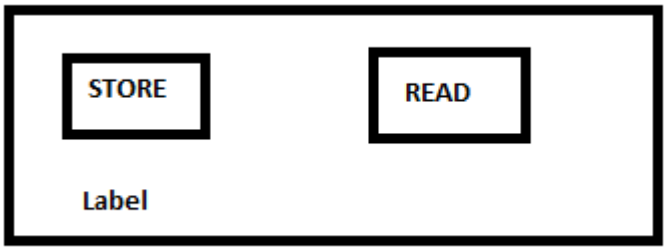
Working in SQL server mode:--

1.Inorder to work with sqlserver mode Ms has given some predefined Tables under Asp state database

2.In order to generate the tables Ms has given a tool i.e aspnet_regsql.

3.This tool is responsible to generate session state tables.

Eg:




```
Button1_click()
{
 Session["x"]="sathyaTech";
Label1.Text="value is stored in session";
}
Button2_click()
{
Label1.Text=session["x"].ToString();
}
```

Generating session state tables:--

goto--->start-->Allprograms--->Visualstudio-2013-->

VisualstudioTools-->Visualstudiocommandprompt-2013

Right click Run as Administrator

In command propt write aspnet_regsql/?

It will display the list of properties to generate session state tables,

-s servername

-u username

-p password

-E connection to sqlserver using windows authentication.

-ss add:- adding support for sqlserver mode.

-ss remove:-Remove the support for sql server mode

-sstype t|p|c:-

t Temporary-session information is stored in Tempdb and stored procedures related to session will store in Asp state database.

P Persisted- Session information will store in Aspstate database

C custom-session information will store in custom database.

When sstype is c we have to use d

d-database

Open commandprompt

aspnet_regsql –s –U -–P abc -ssadd -sstypep

goto-->web.config file

```
<system.web>
    <sessionState mode="SQLServer" allowCustomSqlDatabase="true"
sqlConnectionString="user id=sa;password=abc;
database=ASPState;data source=DOTNET42-PC">
```

</sessionState>

State Server:-

Session state mode=StateServer:-In this mode session information will store in Remote server.

Advantages:-

1. session information will not lost if workerprocess is restarted

2. it supports webgarden and webfarm

3. more scalable compare with inproc mode

DisAdvantages:-

1.Serialization and Deserialization is required.

2.Performance is less when compare to inproc mode.

3.When server is busy sessions will destroy

sessionstate mode="custom":- in this mode session information is stored in custom database like oracle,mysql etc...

sessionstate mode="stateserver"

in this mode session infromation is stored in remote server

<sessionState mode="StateServer"
stateConnectionString="tcpip=localhost:4354">

</sessionState>

Application Memory:- whenever we develop any Webapplication ,we have to deploy the application on webserver, then the first user who is trying to access the appn for first time then a common memory was allocated on webserver i.e

Application memory and that memory was common to all the users who are accessing the application

Different Types of Application Level Events are:-

- **Application_Start**
- **Session_Start**
- **Session_End**
- **Application_Error**
- **Application_End**

Application_Start:-This event will fire whenever the application started on server

session_start:- This event will fire whenever a new user will login

Application_Error:- This event will fire when an unhandled error occurs

Q)where Application Level Events are available?

Global.asax file

Q)How many Global.asax files are available in single application?

Only one

APPLICATION_Ex:--

Ex:-

1. goto--->project---->Addnewitem--->select Global Application class--->name=global.asax

2.
```
protected void Application_Start()
    {
 Application["shc"] = 0;
 }
   protected void Session_Start()
 {
Application["shc"]=Convert.ToInt32(Application    ["shc"])+1;
```

```
    }
    protected void Application_Error()
    {
 Response.Redirect("Demo.aspx");
 }
```

code for Page_Load:-

```
 protected void Page_Load()
 {  Label1.Text = "no of visitors are" + Application["shc"].ToString();      }
  protected void Button1_Click()
    {
Response.Redirect("page2.aspx");
  }
```

Advantages of Application:-

1. multiple users can access same data

2. Easy to access

3. performance is good

4. we can retrieve the data fastly

Limitations:-

1. whenever we restart IIS or whenever we modify the Application then Again Application_Start will fire

2. it is not suitable for webfarm and webgarden scenarios

3. it is not thread safe i.e when multiple users was trying to

access application memory then concurrency problems

may occur

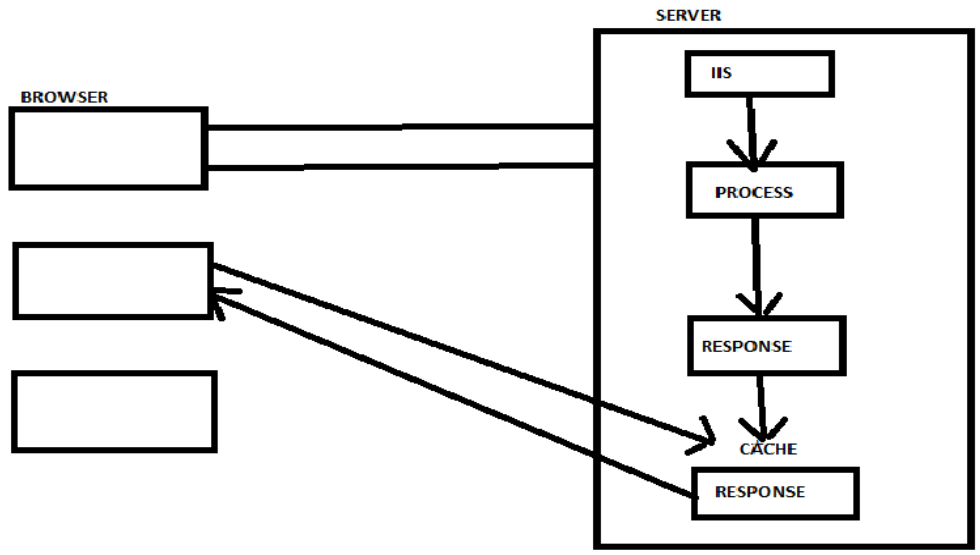
Session	Application
1. session will allocate a seperate memory for every user	1. will allocate a common memory for all the users
2. session will have expiry time	2. Application will not have Expiry time
3. Session supports webfarm and webgarden	3. Application doesnot support webfarm and webgarden

Caching:-Caching is server side state management technique which is used to store frequently used data

in a temporary cache memory

Caching is of 3 Types:-

1. Output Caching
2. Fragment Caching
3. Data Caching



-→Whenever client sends the request web server will accept request,process and generate the response and store the response in cache memory.

→When the next user is submitting the same request response will directly generate from cache memory.

Output Caching:-it is used to store the entire page output

in a temporary cache memory

Ex:-

goto-->webform1.aspx-->drag and Label control from toolbox

```
protected void Page_Load()
    {
        Label1.Text = System.DateTime.Now.ToString();
    }
```

observation:- press F5 each and everytime when the page is refereshed IIS will accept,process and generate respponse

goto--->source code and write the code:-

```
<%@ Page Language="C#" AutoEventWireup="true"  --%>
<%@ OutputCache Duration="10" VaryByParam="none" %>
```

observation:- press F5

when we referesh the page then output will display from cache memory upto 10 seconds and later again the request is considerd as fresh request

Fragment Caching or Partial page caching:-it is used to apply caching for the portion of the page

in order to work with Fragment Caching we have to make use of webuser controls

webuser controls are userdefined controls which can be

Developed by programmer depending on the user requirement

Ex:- goto--->page1.aspx[Design]-->

Drag and Drop Label control from toolbox in page1

goto--->project--->Addnewitem--->select webusercontrol

---->name=mycontrol.ascx--->Add

Drag and Drop Label control from toolbox in mycontrol.ascx[Design]

goto---->page1.aspx.cs:-

```
protected void Page_Load()
{
Label1.Text="i am from  webpage"
  +System.DateTime.Now.ToString();
}
```

goto---->mycontrol.ascx[source]-->

```
<%@ Control Language="C#"   %>
<%@ OutputCache Duration="10" VaryByParam="none" %>
```

goto--->

mycontrol.ascx.cs:-

```
protected void Page_Load()
{   Label1.Text ="i am from webpage"
    +System.DateTime.Now.ToString();
 }
```

DataCaching:- it is used to cache the data

syn to store the value:-

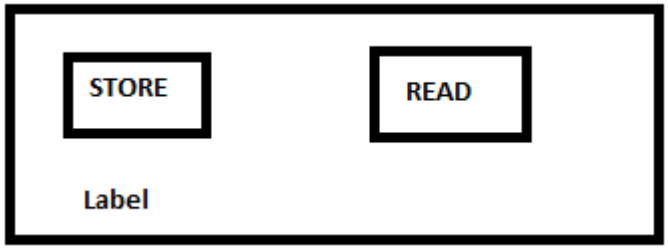
```
Cache["variablename"]=value;
```

syn to retrive the value:-

```
Cache["variablename"];
```

Advantages:-

1. it reduces the burden on server
2. it will performance of the appn because it is not required to process the request everytime by server
3. it increases reliabity



Ex:

```
Protected void Button1_click()
{  cache["x"]="sathya Technologies";  }
```

```
Protected void Button2_click()
{
   Label.Text=cache["x"].ToString();
}
```

Fragment caching (or) partial page caching:--

Apply caching for portion of webpage is called as Fragment caching.We can achieve fragment caching by using web user controls .

WebUSer controls:Creatint userdefined controls.

Eg:

1.goto→page1.aspx→Drag and Drop Label

Code for page1.aspx.cs

```
Protected void Page_Load(){
    Label1.Text=sysem.DateTime.Now.ToString();
}
```

2.goto project on menu bar→Add new item→select webform usercontrol→

Name=mycontrol.ascx

Drag and Drop Label in mycontrol.ascx[Design]

Code for mycontrol.ascx.cs

```
Protected  void Page_Load(){
Label.Text="web user control"+system.Date Time.Now.ToString();
}
```

3.goto→mycontrol.ascx[source] write the code in 2nd Line

```
<%@ Control Language="c#"%>
```

<% @ OutputCache Duration="10" varyByParam="none"%>

4.goto→page1.aspx[Design]

Goto→Solution Explorer→Drag and Drop mycontrol.ascx in page1.aspx

Press F5 and check the output.

Hidden Field:-- Hidden field is client side management Techinique which is used to maintain the response in hidden format.

Ex:-

```
Protected void  Page_Load)
{   HiddenField1.Value="Sathyatech";   }
```

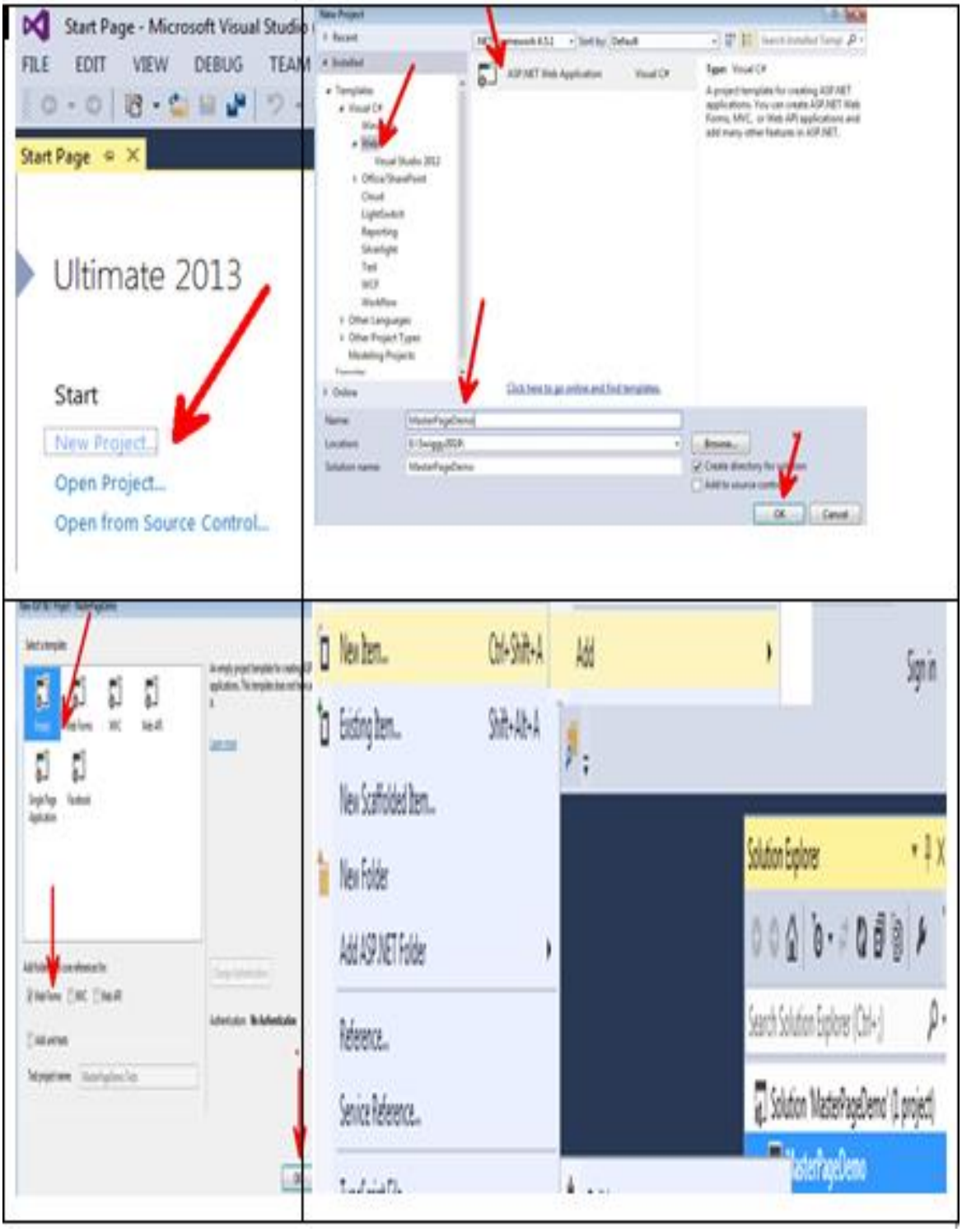
Chapter-8

MasterPages

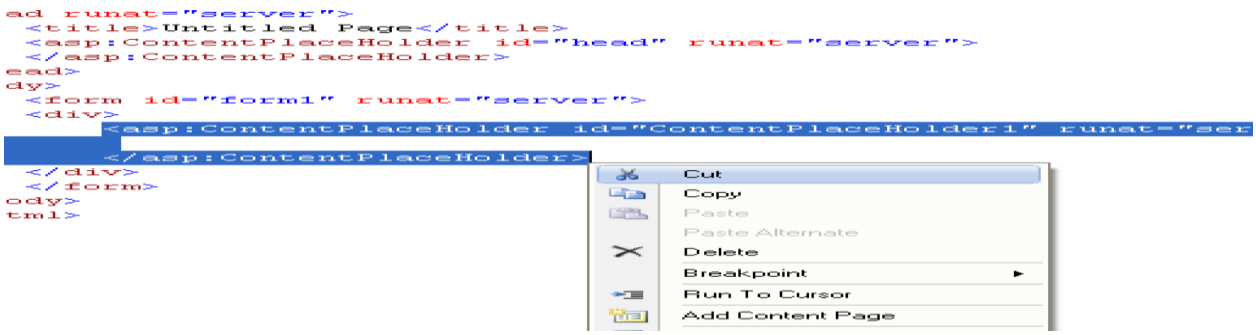
B.Kannababu
(Sathyatechnologies)

STEP-1: GOTO--➐START-----➐RUN-----

DEVENV➐OK



STEP-5: GO TO SOURCE CODE AND CUT THE CONTENT PLACEHOLDER AND GO TO DESIGN WINDOW AND DRAG AND DROP HTML TABLE

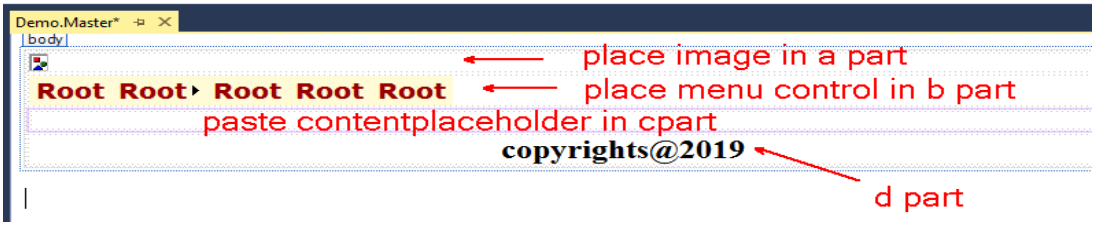


STEP-6:FOR BETTER UNDERSTAND I HAVE NAMED EACH

AND EVERY ROW DON'T CONSIDER THE NAMES IN

MASTERPAGE JUST FOR YOUR UNDERSTANDING ONLY I

HAVE given



PLACE MENU CONTROL IN A SELECT ELIPSE BUTTON IN MENU CONTROL AND CLICK ON AUTOFORMAT SELECT ONE STYLE.

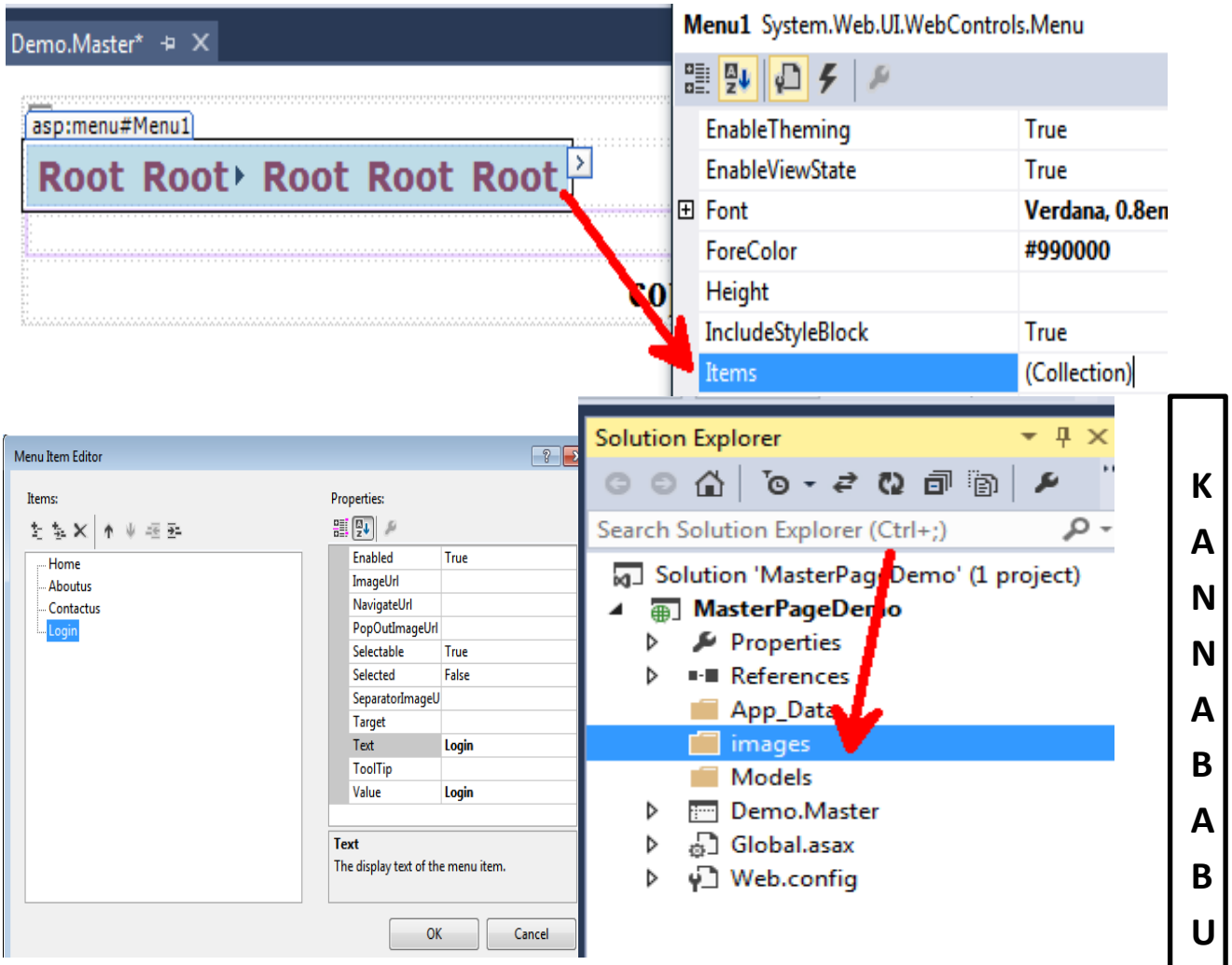
PASTE CONTENT PLACEHOLDER IN E

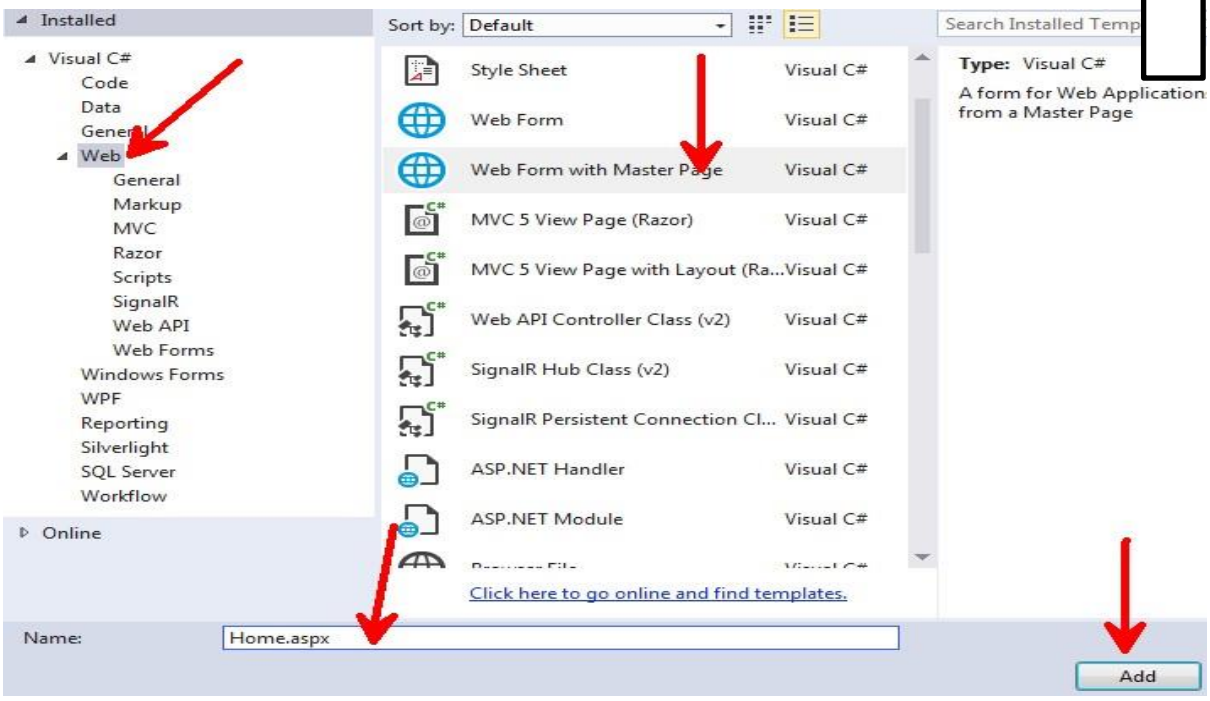
PLACE IMAGE CONTROL IN B NAME=IMAGE1

PLACE IMAGE CONTROL IN C NAME=IMAGE2

TO PLACE IMAGES IN IMAGE CONTROL1 AND IMAGE CONTROL2 GOTO SOLUTION EXPLORER RIGHT CLICK ON PATH

ITEMS=SELECT ROOT ITEM — ❼TEXT=HOME SIMILARLY ADD ABOUTUS, CONTACTUS, JOBSEEKER AND JOBPROVIDER SELECT JOBSEEKER AND ADD CHILD ITEM TEXT=LOGIN JOBSEEKER SIMILARLY SELECT JOB PROVIDER AND ADD CHILD ITEM TEXT=LOGIN JOB PROVIDER





KANNABABU

SATHYATECHNOLOG

KANNABABU

STEP-9: GO TO HOME.ASPX AND DRAG AND DROP SOME IMAGE IN CONTENT PLACEHOLDER

NOTE: WE CANNOT EXECUTE MASTERPAGE MASTERPAGE IS ONLY USED TO IMPORT INTO ANY OF THE ASPX PAGES

NOW GO TO WEBSITE IN HOME.aspx PAGE AND SELECT SET AS STARTPAGE

PRESS F5 AND CHECK THE OUTPUT

STEP-10: SIMILARLY ADD SOME DATA IN CONTENT PLACEHOLDER IN ABOUTUS.aspx AND CONTACTUS.aspx PAGES

STEP-11: GO TO JOBSEEKERLOGIN.aspx DESIGN

Chapter-9

Validation Controls

B.Kannababu
(Sathyatechnologies)

A validation control enables us to validate an input and display an error message if necessary. It is very much like other server-side controls, with certain additional methods and properties. First, the server treats it as an invisible control. After the user has entered erroneous data, it becomes visible. All the validation controls inherit from the base class `BaseValidator`, which is part of the class library namespace. **`System.Web.UI.WebControls.BaseValidator`** exposes a series of properties and methods that are common to all the validation controls.

Different validation Controls

1. Required Field Validator
2. CompareValidator
3. RangeValidator
4. RegularExpressionValidator
5. CustomValidator
6. Validation Summary

- **Common Properties for All validation Controls:**
- **RequiredFieldValidator**: Required Field validator control is used to make a field as mandatory in the form. Without filling the field user can't submit the form.
- **Properties**

PropertyName	**Property Description**
ControlToValidate	Gets or sets the input control to validate (eg: we have to set the control id for which we want to perform validation).
ErrorMessage	This Property is used to display error message

Enabled	true/false. Gets or sets whether to enable the validation control or not.
ValidationGroup	Gets or sets the validation group it belongs to. This is used to group a set of controls.
SetFocusOnError	true/false. Used to move focus on the control that fails the validation.

- **Ex:**

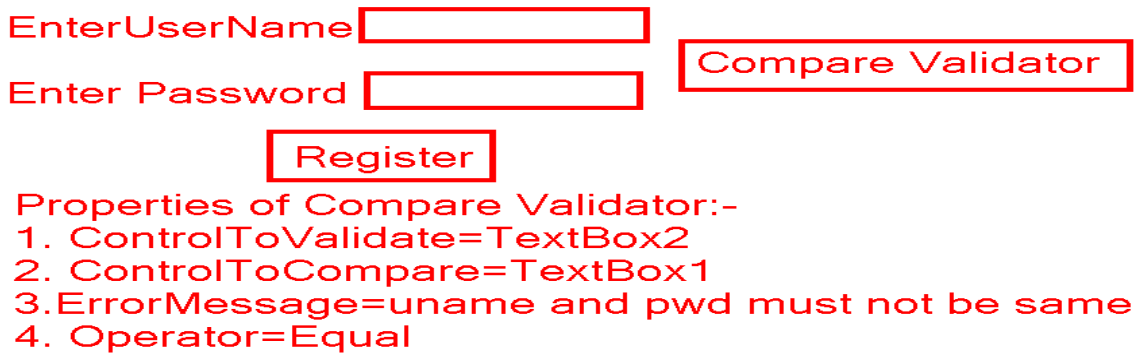
CompareValidator: CompareValidator control is used to compare two values. The value to compare can be either a value of another control or a constant specified. There are predefined data types that can be compared like string, integer etc.

Properties

PropertyName	**Property Description**
ControlToValidate	Gets or sets the input control to validate (eg. The ID value of asp:TextBox control).
ControlToCompare	Gets or sets the ID of the control whose value will be compared with the currently entered value

Text	Gets or sets the description of the error message text.
Enabled	true/false. Gets or sets whether to enable the validation control or not.
ValidationGroup	Gets or sets the validation group it belongs to. This is used to group a set of controls.
SetFocusOnError	true/false. Used to move focus on the control that fails the validation.
Operator	DataTypeCheck/Equal/GreaterThan/GreaterThanEqual/LessThan/LessThanEqual/NotEqual. Used to specify the comparison operation to peform. In case of DataTypeCheck, ControlToCompare properties are ingnored.

Ex:



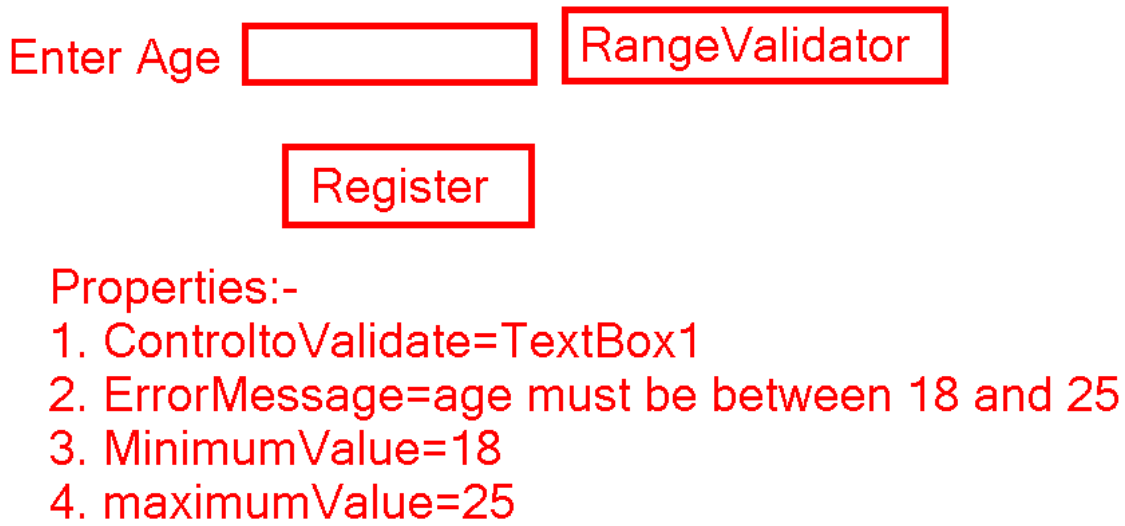
RangeValidator: RangeValidator is used to validate if the given data is in between the specified range or not.

Properties

PropertyName	**Property Description**
ControlToValidate	Gets or sets the input control to validate (eg. The ID value of asp:TextBox control).

ErrorMessage	Gets or sets the text of the error message that will be displayed when validation fails (This is displayed when ValidationSummary validatoin control is used.).
Text	Gets or sets the description of the error message text.
ValidationGroup	Gets or sets the validation group it belongs to. This is used to group a set of controls.
MinimumValue	Gets or sets the minimum value of the range.
MaximumValue	Gets or sets the maximum value of the range.

Ex:-



RegularExpressionValidator:-RegularExpressionValidator is used to make sure that a textbox will accept a predefined format of characters. This format can be of any type like abc@gmailcom (a valid email address).

Properties

PropertyName	Property Description

ValidationExpression	Gets or sets the regular expression that will be used to validate input control data.
ControlToValidate	Gets or sets the input control to validate (eg. The ID value of asp:TextBox control).
Text	Gets or sets the description of the error message text.
Enabled	true/false. Gets or sets whether to enable the validation control or not.
ValidationGroup	Gets or sets the validation group it belongs to. This is used to group a set of controls.
SetFocusOnError	true/false. Used to move focus on the control that fails the validation.

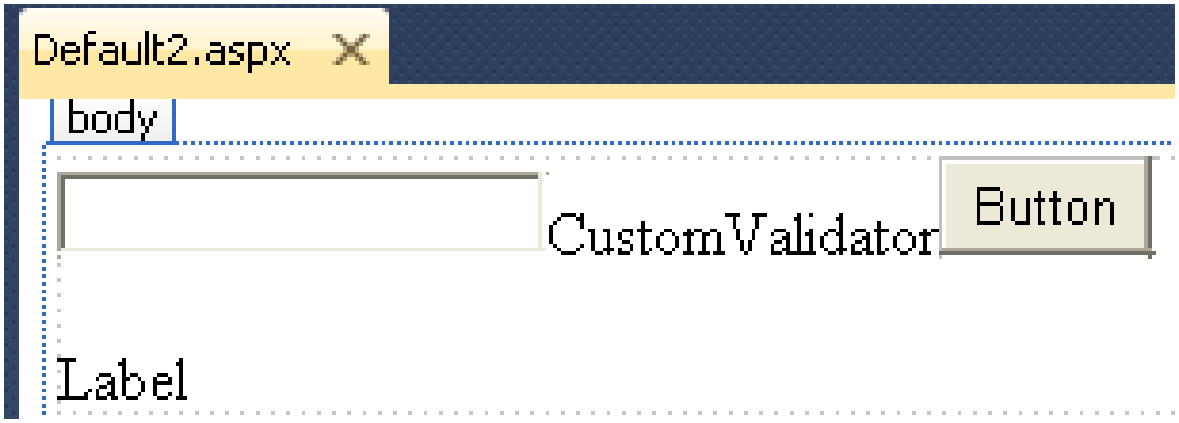
Ex:

CustomValidator control it is used to validate a input control with user-defined function either from server side or client side. Generally this control is used when you feel that no other validation controls fit in your requirement.

Properties

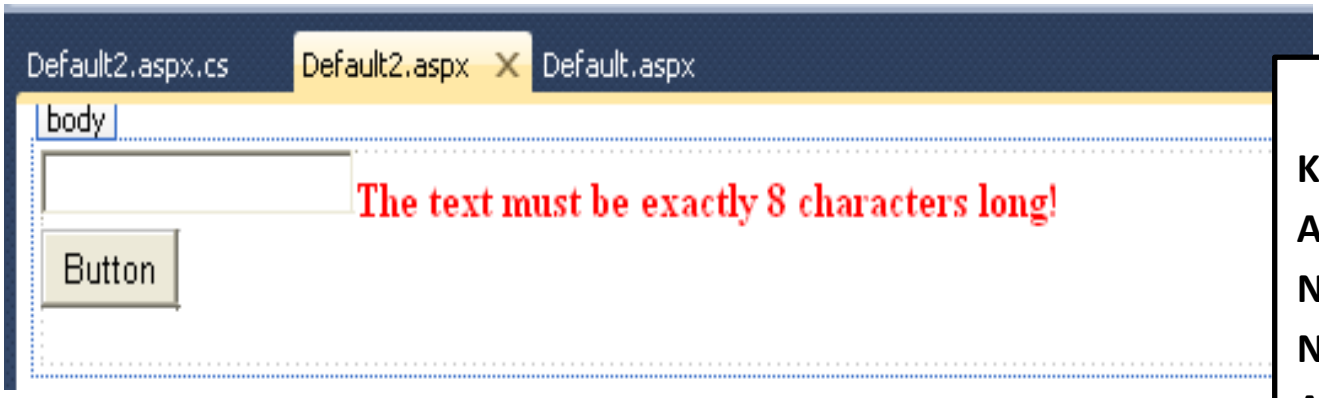
PropertyName	Property Description
ClientValidationFunction	Gets or sets the validation function that will be used in client side (JavaScript function).
OnServerValidate	Method that fires after post back.
ControlToValidate	Gets or sets the input control to validate (eg. The ID value of asp:TextBox control).
ErrorMessage	Gets or sets the text of the error message that will be displayed when validation fails (This is displayed when ValidationSummary validatoin control is used.).
Text	Gets or sets the description of the error message text.
Enabled	true/false. Gets or sets whether to enable the validation control or not.
ValidationGroup	Gets or sets the validation group it belongs to. This is used to group a set of controls.
SetFocusOnError	true/false. Used to move focus on the control that fails the validation.

Customervalidator (Client side)



```
<head runat="server">
    <title></title>
    <script type="text/javascript"
language="javascript">
        function IsEven(source, args)
        {
            if (args.Value % 2 == 0) {
                args.IsValid = true;}
            else {  args.IsValid = false;}
        }
    </script>
</head>
<div>
        <asp:TextBox ID="TextBox1"
runat="server"></asp:TextBox>
        <asp:CustomValidator
ID="CustomValidator1" runat="server"
ControlToValidate="TextBox1"
            ErrorMessage="CustomValidator"
ClientValidationFunction="IsEven"></asp:Custo
mValidator>
        <asp:Button ID="Button1"
runat="server" Text="Button"
OnClientClick="IsEven" />
        <br />
        <br />
        <asp:Label ID="Label1" runat="server"
Text="Label"></asp:Label>
    </div>
```

Customervalidator (Server side)




```
Custom text:<br />
<asp:TextBox runat="server"
id="txtCustom"/>
<asp:CustomValidator runat="server"
id="cusCustom"
controltovalidate="txtCustom"
onservervalidate="cusCustom_ServerValidat
e" errormessage="The text must be exactly
8 characters long!" />
<br /><br/>
```

```
protected void
CustomValidator1_ServerValidate(object
source, ServerValidateEventArgs args)
    {
        if (args.Value.Length== 8)
            args.IsValid = true;
        else
            args.IsValid = false;
    }
==========================================
```

Validation Summary control:-it is used to summarize all validation errors on the page and display. Displays a summary of all current validation errors. In other words, reports a summary of all errors.

Properties

PropertyName	Property Description
ShowMessageBox	true/false. Popup alert box with all validation error, if true.
ShowSummary	rue/false. Display summary of all errors on the page, if true.
DisplayMode	BulletList/List/SingleParagraph. Used to display all validation errors in specified format.
ValidationGroup	Used to specify the group name of input controls for which summary will be displayed.

Ex:-

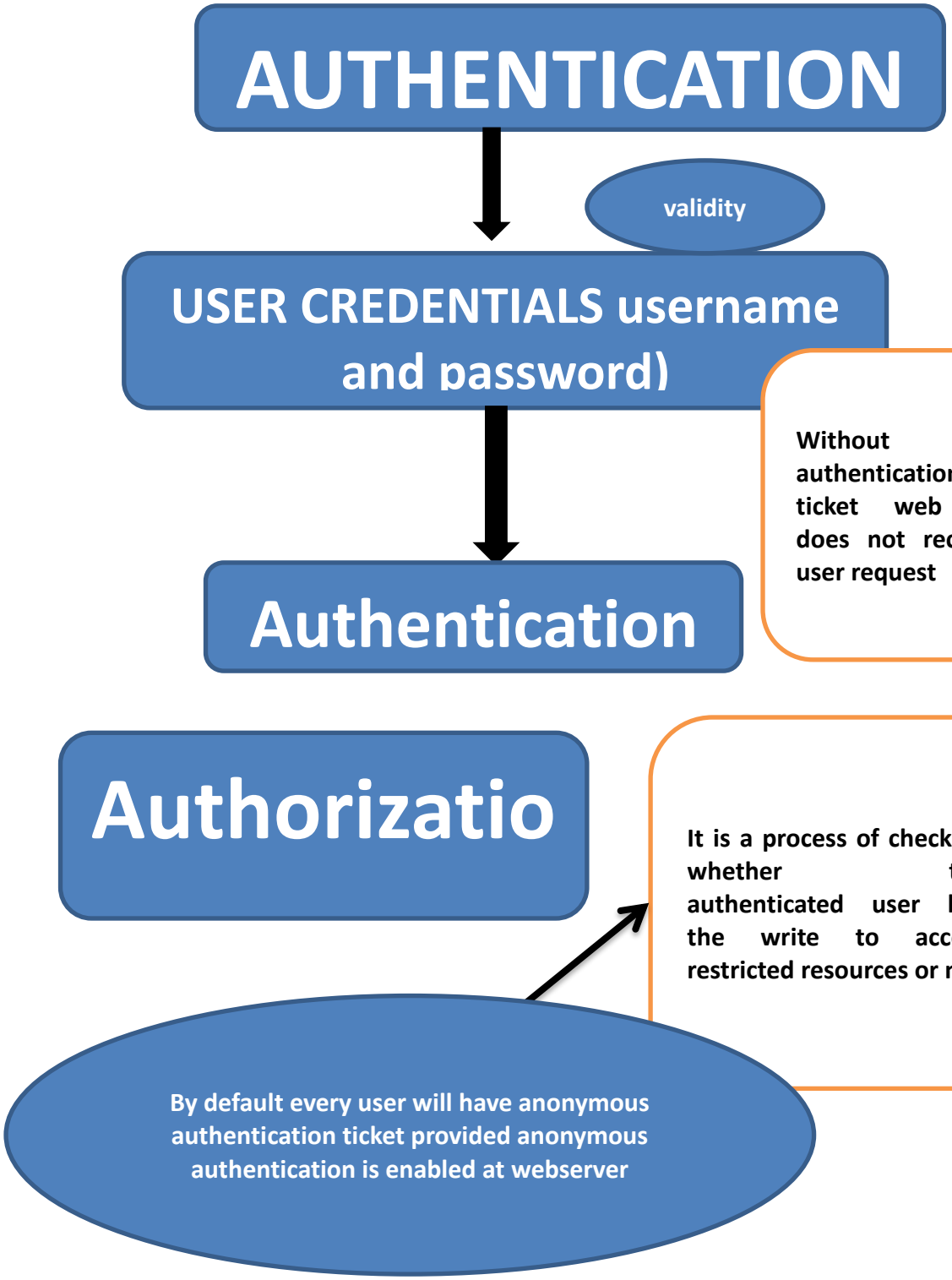
```
<asp:ValidationSummary id="valSummary" runat="server"
 headerText="Please correct the following errors"  display="static"
showSummary= "True" />
```

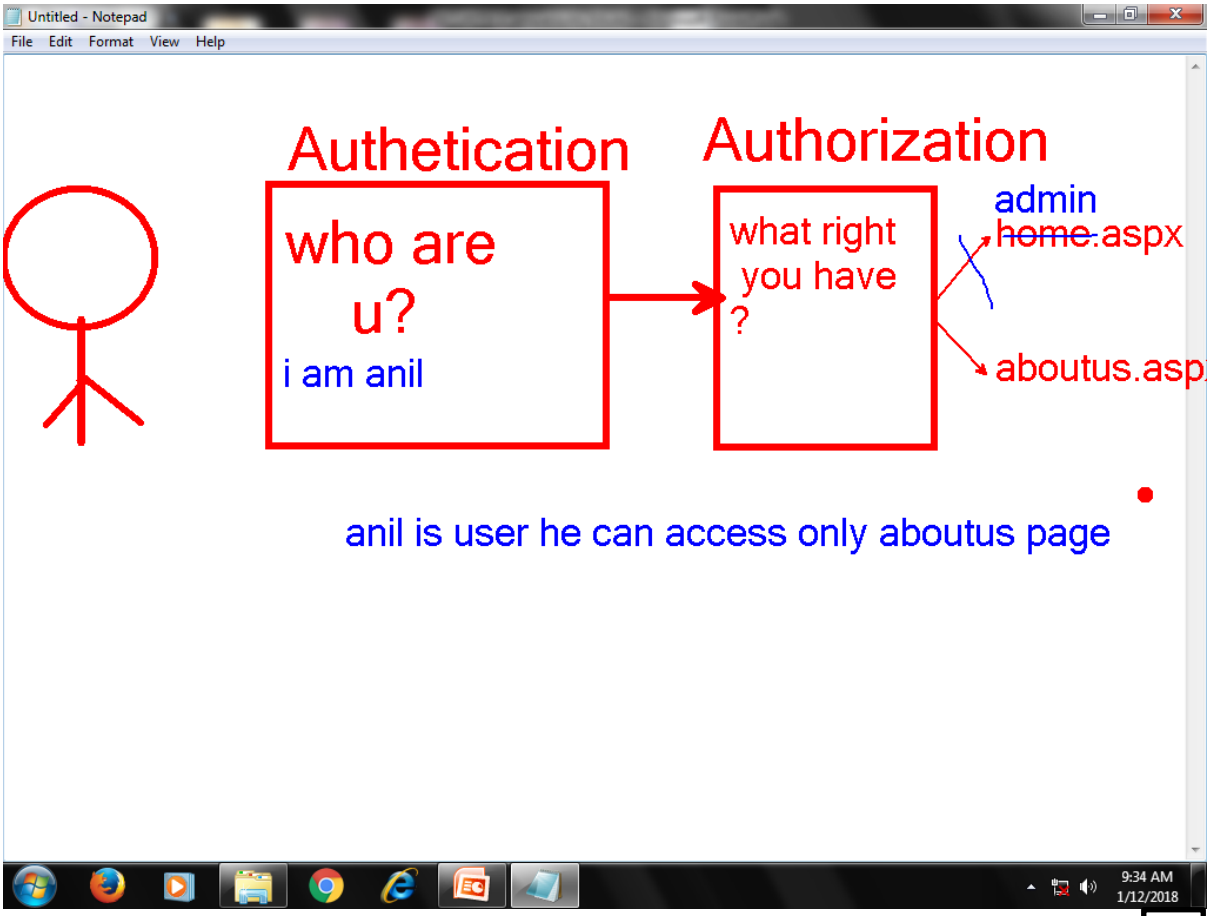
KANNABABU

Chapter-10

Security in ASP.net

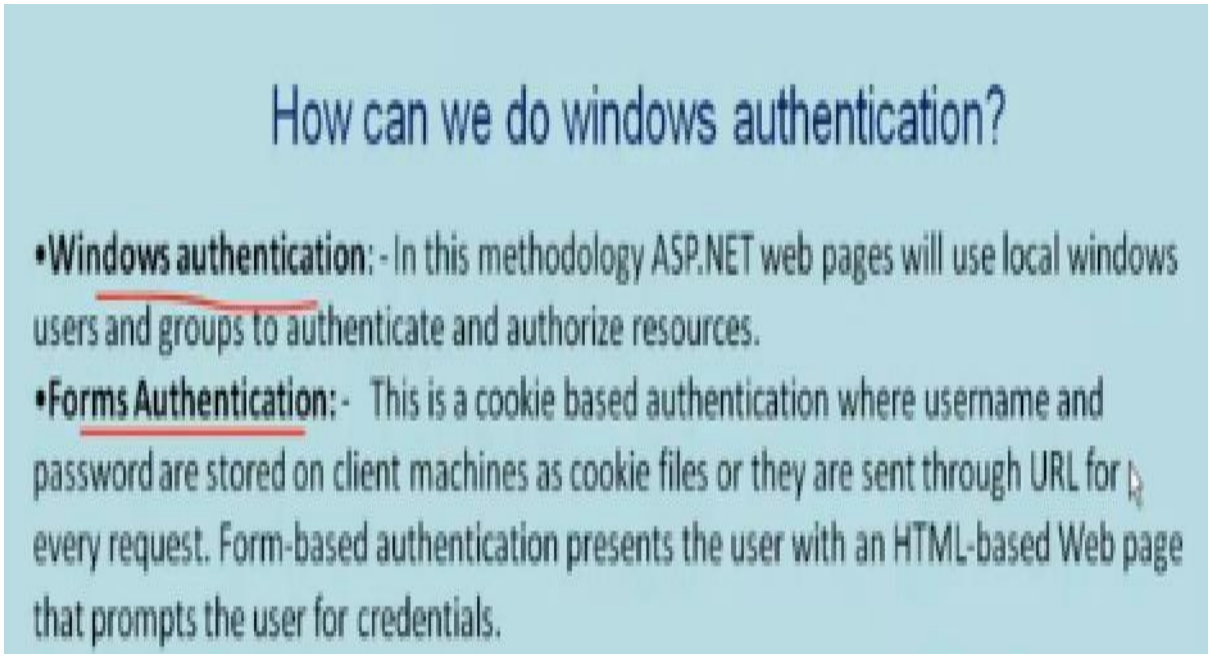
B.Kannababu
(Sathyatechnologies)

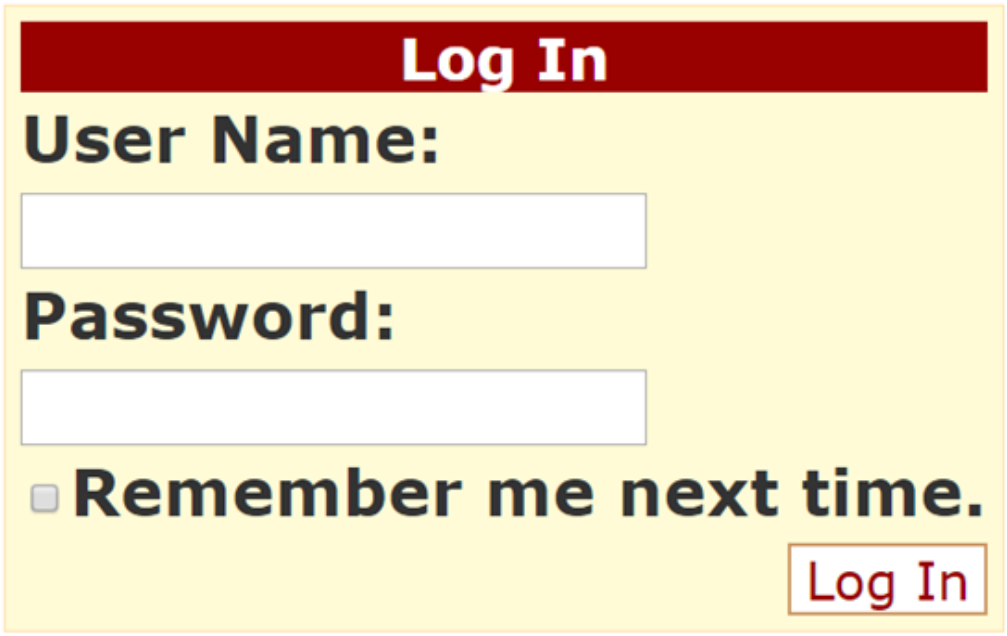




Types of Authentication modes:-

- **Windows Authentication**
- **Forms Authentication**
- **Passport Authentication**

- 

- 

Web.config:-

```
<authentication mode="Forms">
 <forms defaultUrl="Success.aspx" loginUrl="Login.aspx">
  <credentials passwordFormat="Clear">
   <user name="anil" password="anil"/>
   <user name="sunil" password="sunil"/>
  </credentials>
 </forms>
</authentication>
<authorization>
 <deny users="anil"/>
</authorization>
</system.web>
```

Codee for Login.aspx.cs

```
using System.Web.Security;
  protected void Login1_Authenticate(object sender, AuthenticateEventArgs e)
  {
    if(FormsAuthentication.Authenticate(Login1.UserName,Login1.Password))
    {
       FormsAuthentication.RedirectFromLoginPage(Login1.UserName, true);
    }
    else
    {
       Login1.FailureText = "invalid user";
    }
  }
```

Chapter-11

3-Tier Architecture

B.Kannababu
(Sathyatechnologies)

Storedprocedure:-Storedprocedure is a set of precompiled sql statements which will gets executed when we call it whenever we write any sql query:-

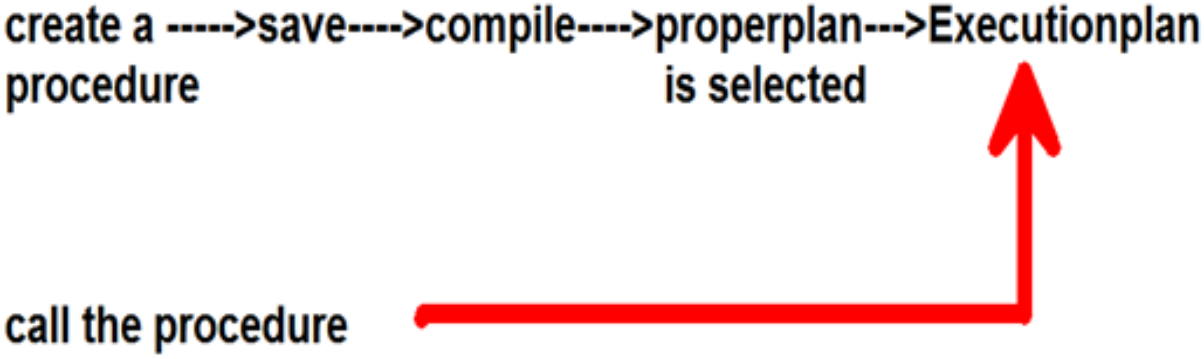
SqlQuery--->compile the query---->proper plan is selected ---->execute the query

Steps to work with Storedprocedure :-
1. create the procedure
2. execute the procedure

syn to create procedure:-

```
create procedure procedurename(parameters)
as begin
  sql statements
end
```

whenever we create aprocedure

create a procedure -----> 

Ex:- create a procedure to insert emp details?

```
create procedure proc_addemp(@eno int,@ename varchar(50),
@salary money)
as begin
  insert into emp values(@eno,@ename,@salary)
end
```

syn to call procedure:-

```
exec procedurename values
exec proc_addemp 101,'anil',20000
```

Ex:- create a procedure to delete emp details based on eno?

```
create procedure proc_deleteemp(@eno int)
as begin
  delete from emp where eno=@eno
end
```

call the procedure:-

```
exec proc_deleteemp 101
```

Ex:- create a procedure to update emp details? create procedure proc_updateemp(@eno int,@ename varchar(50),

```
@salary money)
as begin
update emp set ename=@ename,salary=@salary where eno=@eno
end
```

Call the procedure:-

exec proc_updateemp 102,'anilkumar',30000

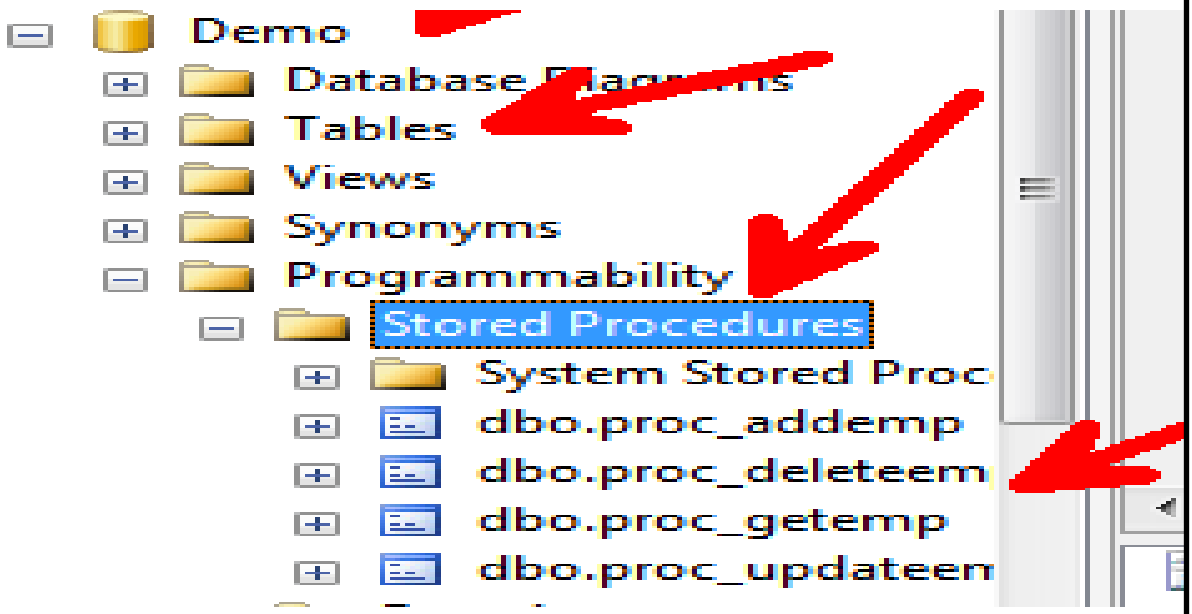
Ex:- create a procedure to Display emp details?

```
create procedure proc_getemp
as begin
 select * from emp
end
```

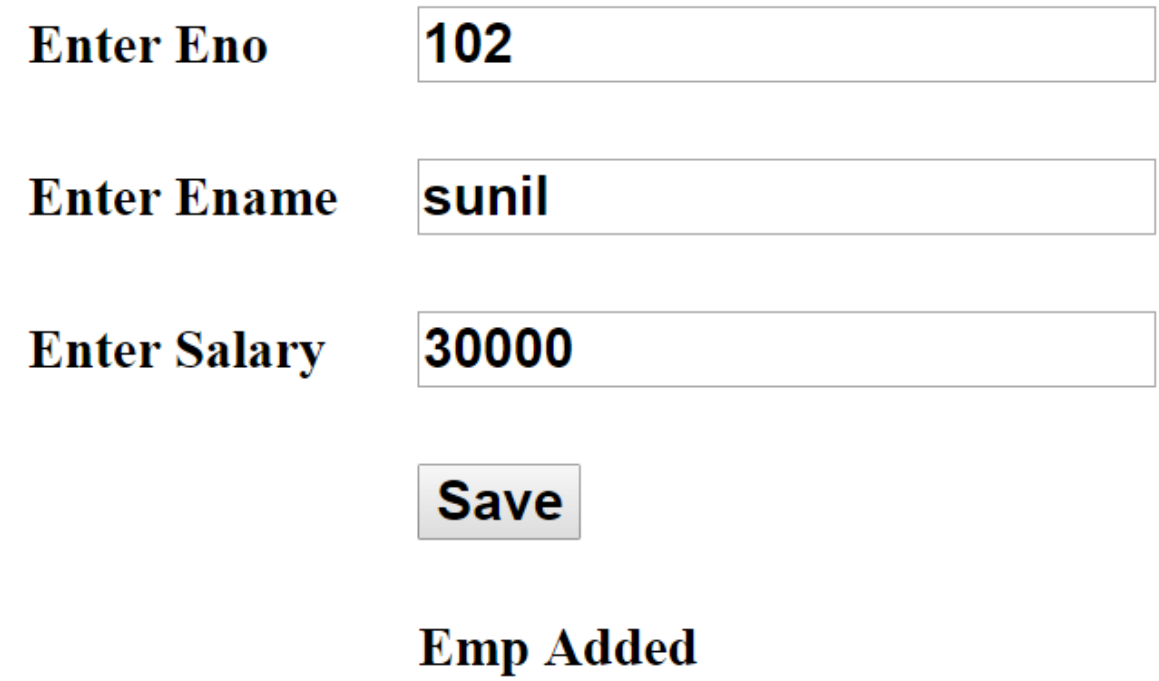
call the procedure:-

```
exec proc_getemp
```

Q)How to view the Storedprocedure?



KANNABABU

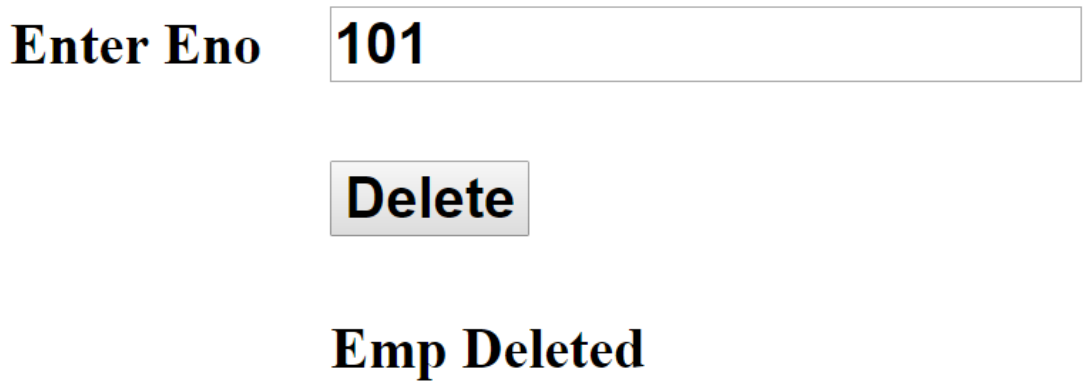



```
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

  protected void Button1_Click(object sender, EventArgs e)
      {
          //create the connection
          SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
          //open the connection
          con.Open();
          //pass the procedure
          SqlCommand cmd = new SqlCommand("proc_addemp",con);
          //mention that we are working with SP
          cmd.CommandType = CommandType.StoredProcedure;
          //pass the values to parameters
cmd.Parameters.AddWithValue("@eno", TextBox1.Text);
cmd.Parameters.AddWithValue("@ename", TextBox2.Text);
cmd.Parameters.AddWithValue("@salary", TextBox3.Text);
```

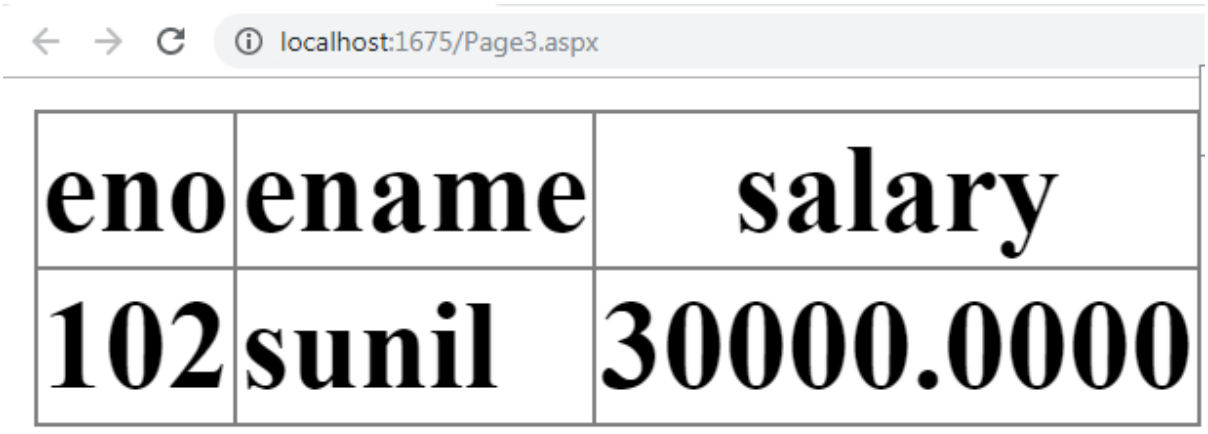
```
        //Execute the procedure
        int i=cmd.ExecuteNonQuery();
        //close the connection
        con.Close();
        if(i==1)
        {
            Label1.Text = "Emp Added";
        }
        else
        {
            Label1.Text = "Failed";
        }
    }
```

Declare Connectionstring Globally in Web.config file:-

```
</system.web>
  <connectionStrings>
    <add name="constr" connectionString="user
id=sa;password=abc;database=demo;data source=."/>
  </connectionStrings>
```



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


    protected void Button1_Click(object sender, EventArgs e)
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        SqlCommand cmd = new SqlCommand("proc_deleteemp",con);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@eno", TextBox1.Text);
       int i=cmd.ExecuteNonQuery();
       con.Close();
       if(i==1)
       {
          Label1.Text = "Emp Deleted";
       }
       else
       {
          Label1.Text = "Failed";
       }
    }
```




```
using System.Data;
```

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

 protected void Page_Load(object sender, EventArgs e)
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        SqlDataAdapter da = new SqlDataAdapter("proc_getemp", con);
        DataSet ds = new DataSet();
        da.Fill(ds, "emp");
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
```

Advantages of StoredProcedures:-

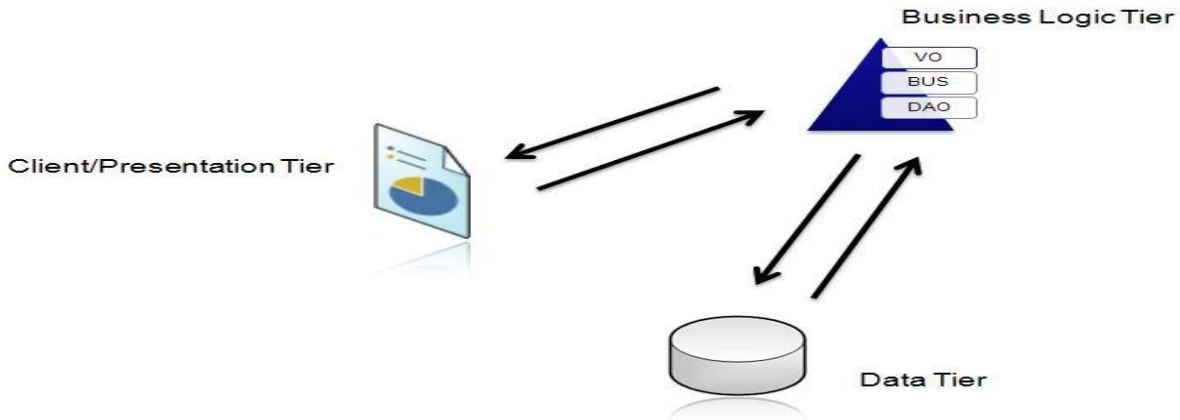
- Execution plan retention and reuse
- Query auto-parameterization
- Encapsulation of business rules and policies
- Application modularization
- Sharing of application logic between applications
- Access to database objects that is both secure and uniform
- Consistent, safe data modification
- Network bandwidth conservation
- Support for automatic execution at system start-up
- Enhanced hardware and software capabilities
- Improved security
- Reduced development cost and increased reliability
- Centralized security, administration, and maintenance for common routines

3Tier-Architecture

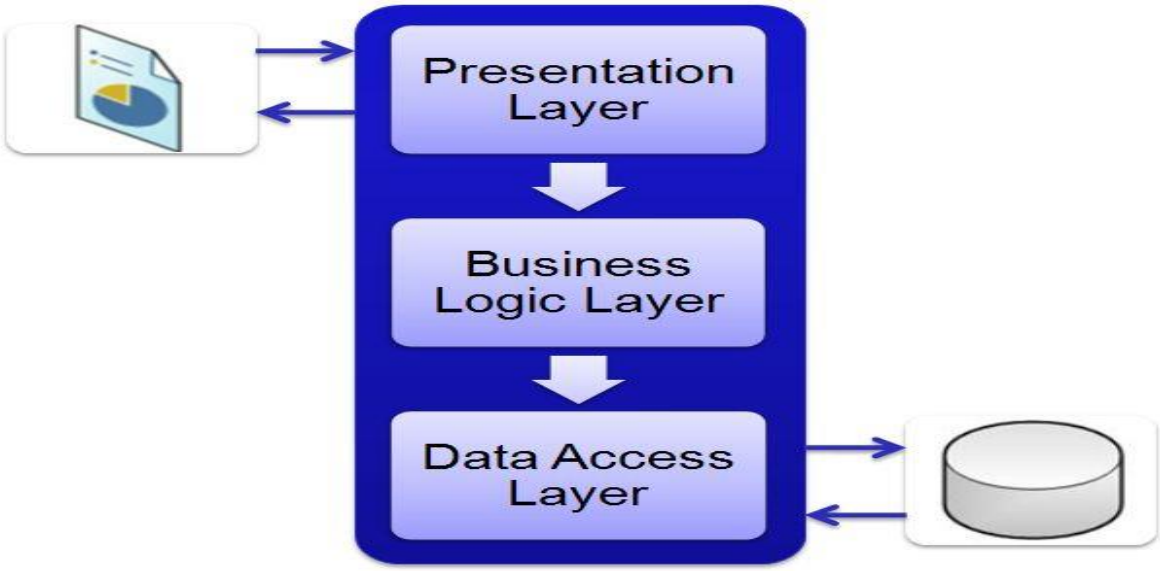
Tier and Layer

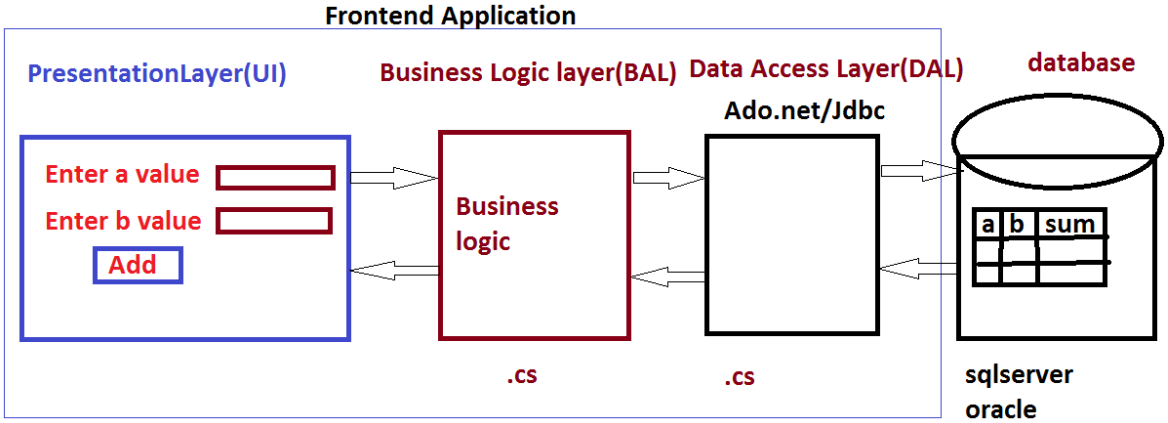
Firstly we need to clarify the difference between two terms in N-Tier architecture: tier and layer. Tier usually means the physical deployment computer. Usually an individual running server is one tier. Several servers may also be counted as one tier, such as server failover clustering. By contrast, layer usually means logic software component group mainly by functionality; layer is used for software development purpose. Layer software implementation has many advantages and is a good way to achieve N-Tier architecture. Layer and tier may or may not exactly match each other. Each layer may run in an individual tier. However, multiple layers may also be able to run in one tier.

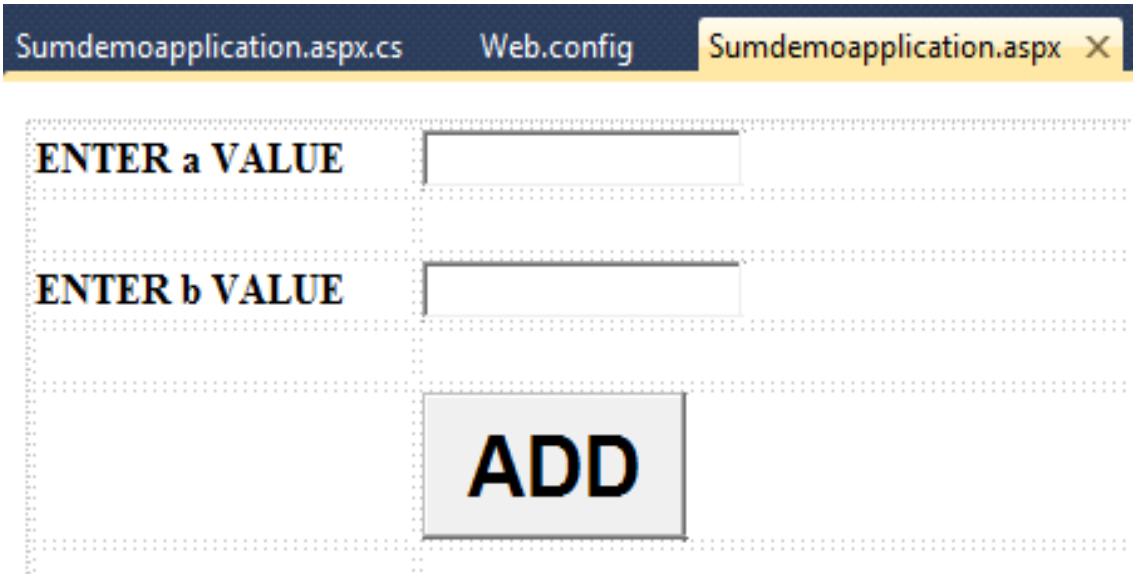
Tier indicates a physical separation of components, which may mean different assemblies such as DLL, EXE, etc. on the same server or multiple servers.



Layer indicates logical separation of components, such as having distinct namespaces and classes for the Database Access Layer, Business Logic Layer and User Interface Layer.



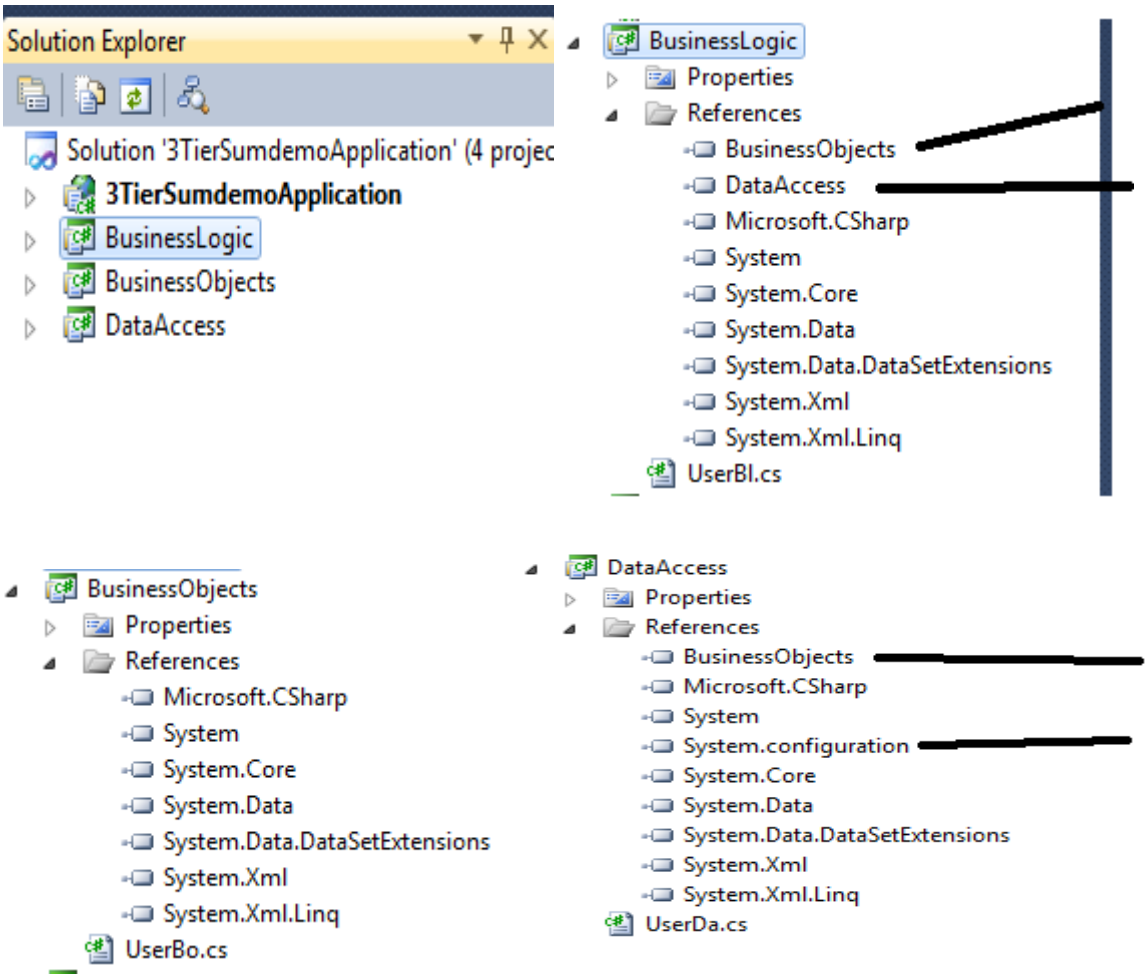




SQLQuery2.sql - K...C.KANNA (sa (53))* KANNA-PC.KANNA

a	b	sum
10	20	30
10	20	30

```
create procedure proc_add(@a int,@b int,@sum int)
as begin
insert into sumdemo values(@a,@b,@sum)
end
```



Code for Web.Config File and declare the connectionstring

```
<connectionStrings>
  <add name="constr"
      connectionString="user id=sa;password=abc;database=KANNA;data
source=KANNA-PC"
      providerName="System.Data.SqlClient" />
 </connectionStrings>
```

Double click on Button and write the Code in sumdemoapplication.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using BusinessObjects;
namespace _3TierSumdemoApplication
{
  public partial class Sumdemoapplication : System.Web.UI.Page
  {

    protected void Button1_Click(object sender, EventArgs e)
    {
      if (txtavalue.Text=="")
      {
        ScriptManager.RegisterStartupScript(this, this.GetType(), "alert",
"alert('Please entera value')", true);
      }
      else if (txtbvalue.Text == "")
      {
        ScriptManager.RegisterStartupScript(this, this.GetType(), "alert",
"alert('Please enter b value')", true);
      }
      else
      {
```

```
            BusinessObjects.UserBo objUBo = new UserBo();
           //pass the values from Controls through properties
            objUBo.A = int.Parse(txtavalue.Text);
            objUBo.B = int.Parse(txtbvalue.Text);
            BusinessLogic.UserBl objUbl = new BusinessLogic.UserBl();
           int r= objUbl.Addmethod(objUBo);
           if (r>0)
           {
            ScriptManager.RegisterStartupScript(this, this.GetType(), "alert",
"alert('Record is inserted')", true);
           }
         }
      }
   }
}
```

Code for UserBo.cs


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace BusinessObjects
{
    public class UserBo
   {
      //variables
      private int a;
      private int b;
      private int sum;
      //Properties
      public int A
      {
         set { a = value; }
         get { return a; }
      }
      public int B
```

```
        {
            get { return b; }
            set { b = value; }
        }
        public int Sum
        {
            get { return sum; }
            set { sum = value; }
        }
    }
}
```

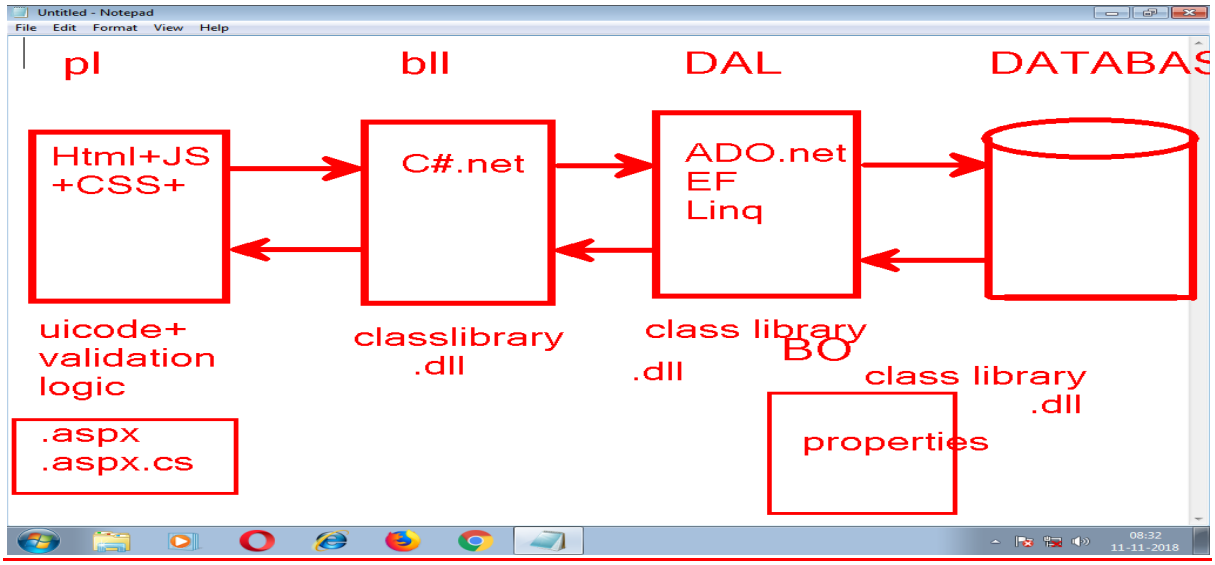
Code for userBl.cs

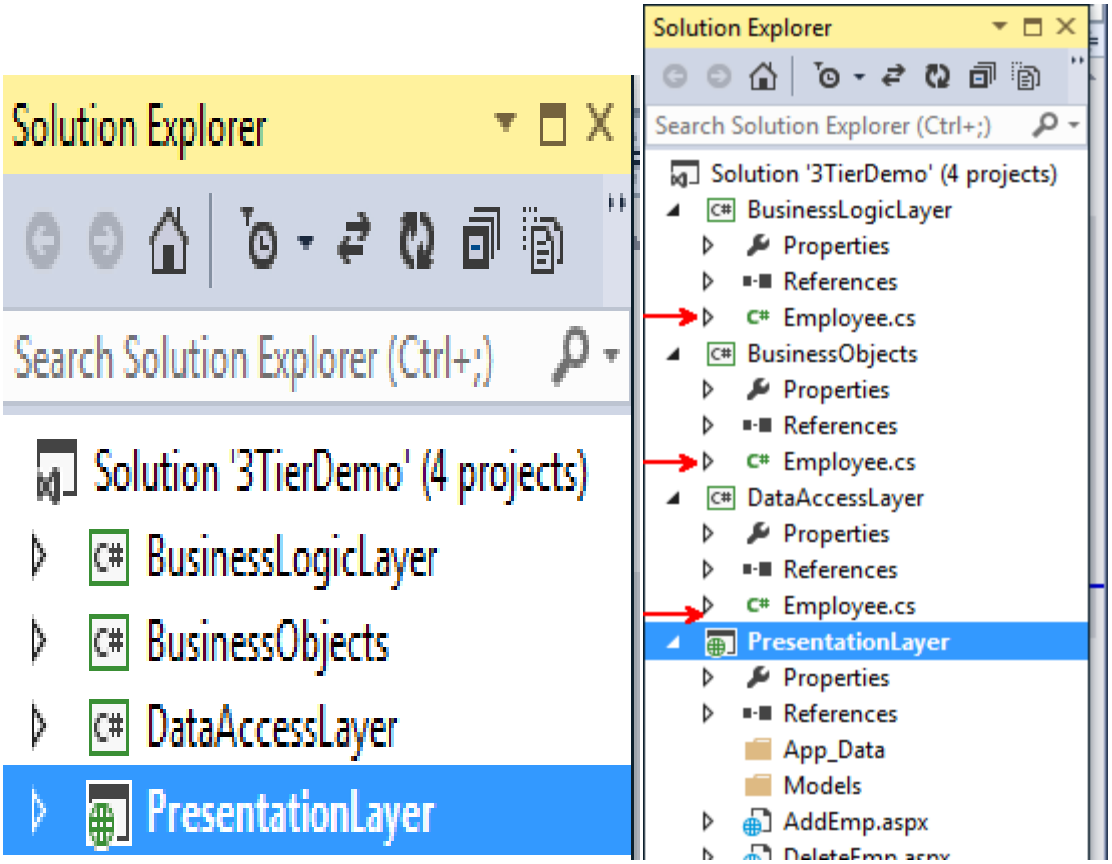
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using BusinessObjects;
using DataAccess;
namespace BusinessLogic
{
  public  class UserBl
   {
     public int Addmethod(BusinessObjects.UserBo objUBo)
     {
        objUBo.Sum = objUBo.A + objUBo.B;
        DataAccess.UserDa objUserDa = new UserDa();
      return objUserDa.InsertRecords(objUBo);

     }
   }
}
```

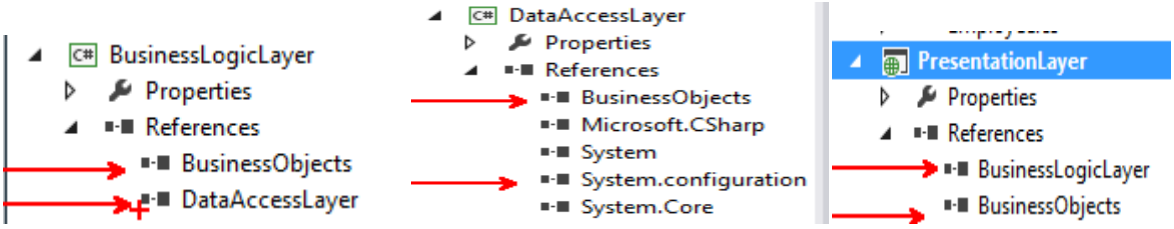
Code for UserDa.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```
using System.Configuration;
using BusinessObjects;
using System.Data.SqlClient;
using System.Data;
namespace DataAccess
{
  public class UserDa
   {
      public int InsertRecords(BusinessObjects.UserBo objBo)
      {
         SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
         con.Open();
         SqlCommand cmd = new SqlCommand("proc_add",con);
         cmd.CommandType = CommandType.StoredProcedure;
         cmd.Parameters.AddWithValue("@a", objBo.A);
         cmd.Parameters.AddWithValue("@b", objBo.B);
         cmd.Parameters.AddWithValue("@sum", objBo.Sum);
        int result= cmd.ExecuteNonQuery();
        con.Close();
        return result;
      }
   }
}
```







Code for BusienssObjects.cs:-

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

```
namespace BusinessObjects
{
   public class Employee
   {
      public int Eno { get; set; }
      public string Ename { get; set; }
      public double Salary { get; set; }
   }
}
```

Code for DataAccessLayer.cs:-


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data.SqlClient;
using System.Configuration;
using BusinessObjects;
using System.Data;
namespace DataAccessLayer
{
   public class Employee
   {
      public int AddEmployee(BusinessObjects.Employee objboemp)
      {
         SqlConnection con=new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
         con.Open();
         string query="insert into emp
values('"+objboemp.Eno+"','"+objboemp.Ename+"','"+objboemp.Salary+"')";
         SqlCommand cmd=new SqlCommand(query,con);
         int i=cmd.ExecuteNonQuery();
         con.Close();
         return i;
      }
```

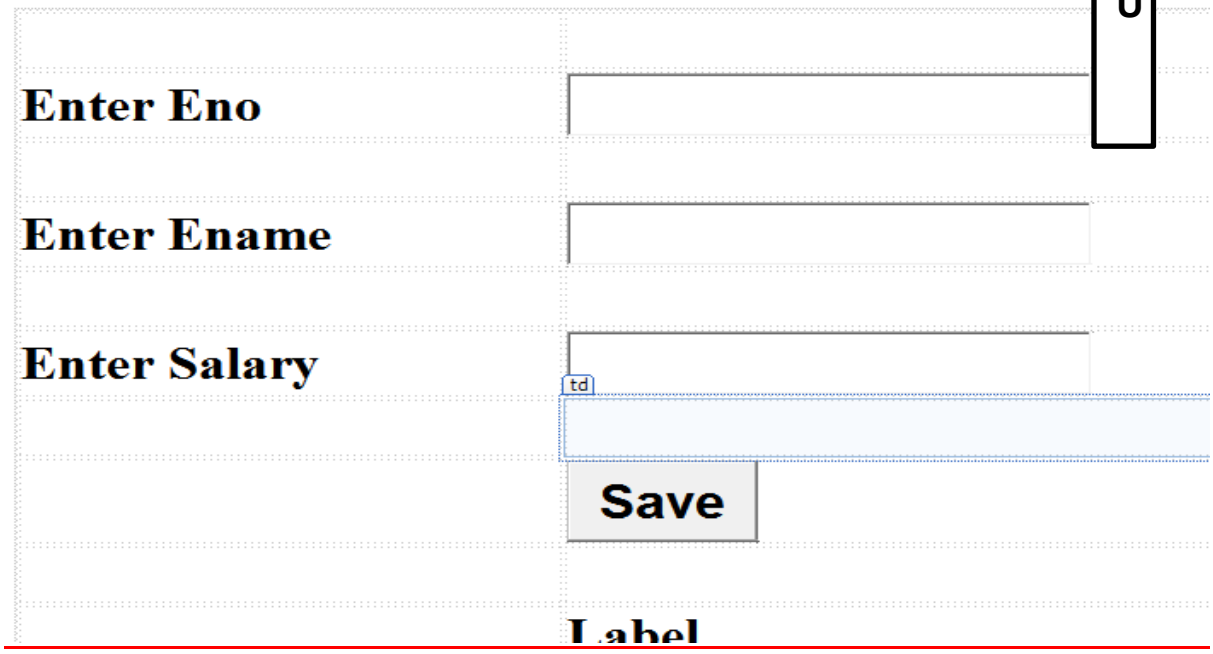
```
    public int DeleteEmployee(BusinessObjects.Employee objboemp)
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        con.Open();
        string query="delete from emp where eno='"+objboemp.Eno+"'";
        SqlCommand cmd = new SqlCommand(query,con);
       int i=cmd.ExecuteNonQuery();
        con.Close();
        return i;
    }
    public DataSet FillData()
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["constr"].ToString());
        SqlDataAdapter da = new SqlDataAdapter("select * from emp",con);
        DataSet ds = new DataSet();
        da.Fill(ds, "emp");
        return ds;
    }
  }
}
```

Code for BusienessLogicLayer.cs:-

```
using System;
using DataAccessLayer;
using BusinessObjects;
using System.Data;
namespace BusinessLogicLayer
{
  public class Employee
  {
    DataAccessLayer.Employee objdalemp = new DataAccessLayer.Employee();
    public int AddEmployee(BusinessObjects.Employee objboemp)
    {
      int i=objdalemp.AddEmployee(objboemp);
      return i;      }
```

```
    public int DeleteEmployee(BusinessObjects.Employee objboemp)
    {
      int i=objdalemp.DeleteEmployee(objboemp);
      return i;
    }
    public DataSet FillData()
    {
      DataSet ds=objdalemp.FillData();
      return ds;
    }
  }
}
```

AddEmp.aspx:-



Code for AddEmp.aspx.cs:-

```
using System;
using BusinessObjects;
using BusinessLogicLayer;
namespace PresentationLayer
```

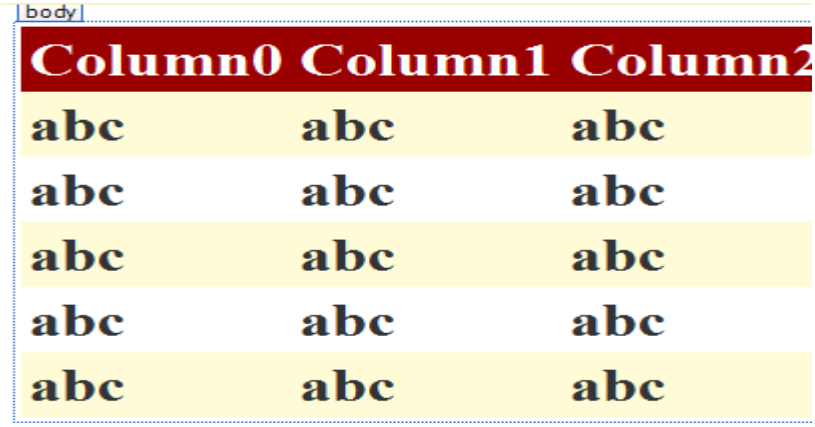
```
{
   public partial class AddEmp : System.Web.UI.Page
   {
      protected void Button1_Click(object sender, EventArgs e)
      {
         BusinessObjects.Employee objboemp = new
BusinessObjects.Employee();
         objboemp.Eno = int.Parse(TextBox1.Text);
         objboemp.Ename = TextBox2.Text;
         objboemp.Salary = double.Parse(TextBox3.Text);
         BusinessLogicLayer.Employee objbllemp = new
BusinessLogicLayer.Employee();
        int i=objbllemp.AddEmployee(objboemp);
        if (i == 1)
        {
           Label1.Text = "Record Inserted";
        }
        else
        {
           Label1.Text = "Failed";
        }
      }
   }
}
```

DeleteEmp.aspx:-


```
using System;
using BusinessLogicLayer;
using BusinessObjects;
namespace PresentationLayer
{
   public partial class DeleteEmp : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }
      protected void Button1_Click(object sender, EventArgs e)
      {
         BusinessObjects.Employee objboemp = new
BusinessObjects.Employee();
         objboemp.Eno = int.Parse(TextBox1.Text);
         BusinessLogicLayer.Employee objbllemp = new
BusinessLogicLayer.Employee();
        int i=objbllemp.DeleteEmployee(objboemp);
        if(i==1)
        {
           Label1.Text = "Record Deleted";
        }

        else
        {
```

```
            Label1.Text = "Failed";
        }

    }
  }
}
```

GetEmp.aspx:-




```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using BusinessLogicLayer;
using System.Data;
namespace PresentationLayer
{
   public partial class GetEmp : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         BusinessLogicLayer.Employee objbllemp = new Employee();
        DataSet ds=objbllemp.FillData();
        GridView1.DataSource = ds;
       GridView1.DataBind();      }   } }
```