Without Class and Object

//Create Database

CREATE TABLE CURD(

Roll INT,

Name VARCHAR(50),

City VARCHAR(50)

)

//Print Database

SELECT \* FROM CURD;

//Insert Query Procedure

CREATE PROCEDURE curd\_op(@Roll1 AS INT,

@Name1 as VARCHAR(50),

@City1 as VARCHAR(50))

AS

BEGIN

INSERT INTO CURD(Roll, Name, City) VALUES (@Roll1, @Name1, @City1);

END;

//Show Query Procedure

CREATE PROCEDURE Show\_curd1(@Roll1 AS INT)

AS

BEGIN

SELECT \* FROM CURD;

END;

//Find Query Procedure

CREATE PROCEDURE Find\_curd1(@Roll1 AS INT)

AS

BEGIN

SELECT \* FROM CURD WHERE Roll = @Roll1;

END;

//Update Query Procedure

CREATE PROCEDURE Update\_curd1(@Roll1 AS INT,

@Name1 AS VARCHAR(50),

@City1 AS VARCHAR(50))

AS

BEGIN

UPDATE CURD SET Name = @Name1, City = @City1 WHERE Roll = @Roll1;

END;

//Delete Query Procedure

CREATE PROCEDURE Delete\_curd1(@Roll1 AS INT)

AS

BEGIN

DELETE FROM CURD WHERE Roll = @Roll1;

END;

File Name – default.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="default.aspx.cs" Inherits="WebApplication1.\_default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h1>CURD Operation</h1>

Enter Roll : <asp:TextBox ID="txtRoll" runat="server"></asp:TextBox>&nbsp;&nbsp; <asp:Button ID="btnFind" runat="server" Text="Find" OnClick="btnFind\_Click" />

<br />

Enter Name : <asp:TextBox ID="txtName" runat="server"></asp:TextBox>

<br />

Enter City : <asp:TextBox ID="txtCity" runat="server"></asp:TextBox>

<br />

<br />

<asp:Button ID="btnSubmit" runat="server" Text="Submit" OnClick="btnSubmit\_Click" />

&nbsp;&nbsp;

<asp:Button ID="btnUpdate" runat="server" Text="Update" OnClick="btnUpdate\_Click" />

&nbsp;&nbsp;

<asp:Button ID="btnDelete" runat="server" Text="Delete" OnClick="btnDelete\_Click" />

&nbsp;&nbsp;

<asp:Button ID="btnShow" runat="server" Text="Show" OnClick="btnShow\_Click" />

<br />

</div>

<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False">

<Columns>

<asp:BoundField DataField="Roll" HeaderText="Roll No." />

<asp:BoundField DataField="Name" HeaderText="Name" />

<asp:BoundField DataField="City" HeaderText="City" />

</Columns>

</asp:GridView>

</form>

</body>

</html>

File Name – Default.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.SqlClient;

using System.Configuration;

using System.Data;

namespace WebApplication1

{

public partial class \_default : System.Web.UI.Page

{

SqlConnection con;

SqlCommand com;

SqlDataReader dr;

protected void Page\_Load(object sender, EventArgs e)

{

if(!IsPostBack)

{

}

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("curd\_op", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", txtRoll.Text);

com.Parameters.AddWithValue("Name1", txtName.Text);

com.Parameters.AddWithValue("City1", txtCity.Text);

com.ExecuteNonQuery();

Response.Write("Data Insert");

}

protected void btnShow\_Click(object sender, EventArgs e)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Show\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", txtRoll.Text);

dr = com.ExecuteReader();

GridView1.DataSource = dr;

GridView1.DataBind();

dr.Close();

}

protected void btnFind\_Click(object sender, EventArgs e)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Find\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", txtRoll.Text);

dr = com.ExecuteReader();

if(dr.Read())

{

txtRoll.Text = dr["Roll"].ToString();

txtName.Text = dr["Name"].ToString();

txtCity.Text = dr["City"].ToString();

}

dr.Close();

}

protected void btnUpdate\_Click(object sender, EventArgs e)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Update\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", txtRoll.Text);

com.Parameters.AddWithValue("Name1", txtName.Text);

com.Parameters.AddWithValue("City1", txtCity.Text);

com.ExecuteNonQuery();

Response.Write("Data Update");

}

protected void btnDelete\_Click(object sender, EventArgs e)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Delete\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", txtRoll.Text);

com.ExecuteNonQuery();

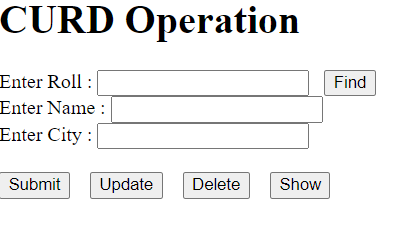
Response.Write("Data Delete");

}

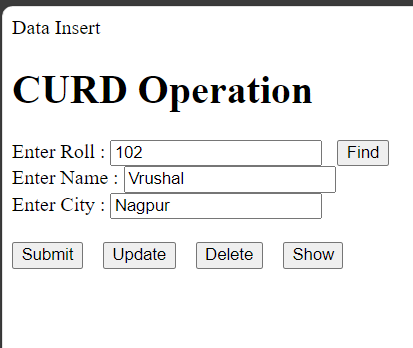
}

}

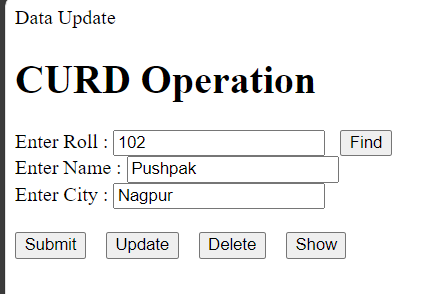
Output –



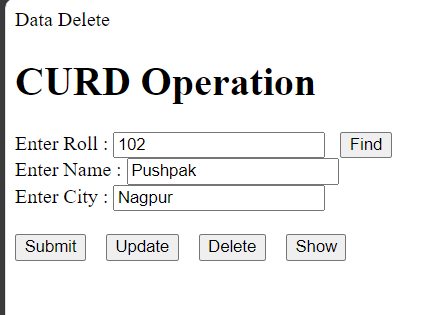
Screen 1 - Data Insert



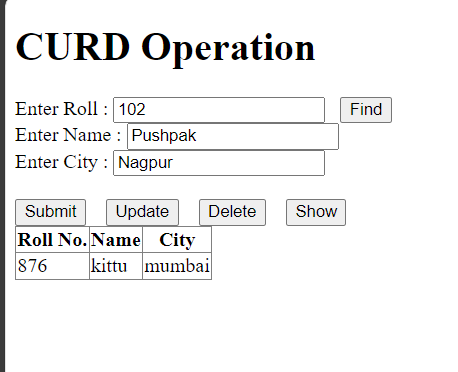
Screen 2 - Data Update



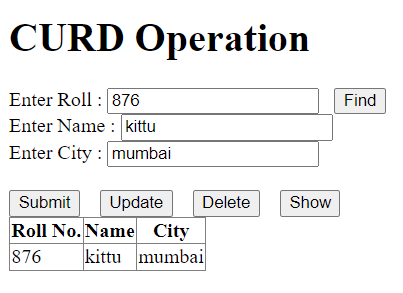
Screen 3 - Data Delete



Screen 4 - Show Grid



Screen 5 - Find record



CURD operation using class and object

File Name – classobj.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.SqlClient;

using System.Configuration;

using System.Data;

namespace WebApplication1

{

public partial class classobj : System.Web.UI.Page

{

SqlConnection con;

SqlCommand com;

SqlDataReader dr;

protected void Page\_Load(object sender, EventArgs e)

{

if(!IsPostBack)

{

}

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

operation\_curd o1 = new operation\_curd();

o1.insert\_q(int.Parse(txtRoll.Text), txtName.Text, txtCity.Text);

}

protected void btnFind\_Click(object sender, EventArgs e)

{

operation\_curd f1 = new operation\_curd();

f1.find\_q(int.Parse(txtRoll.Text), txtName.Text, txtCity.Text);

if(f1.dr.Read())

{

txtRoll.Text = f1.dr["Roll"].ToString();

txtName.Text = f1.dr["Name"].ToString();

txtCity.Text = f1.dr["City"].ToString();

}

f1.dr.Close();

Response.Write("Done");

}

protected void btnShow\_Click(object sender, EventArgs e)

{

operation\_curd s1 = new operation\_curd();

s1.show\_q();

GridView1.DataSource = s1.dr;

GridView1.DataBind();

s1.dr.Close();

}

protected void btnUpdate\_Click(object sender, EventArgs e)

{

operation\_curd u1 = new operation\_curd();

u1.update\_q(int.Parse(txtRoll.Text), txtName.Text, txtCity.Text);

}

protected void btnDelete\_Click(object sender, EventArgs e)

{

operation\_curd d1 = new operation\_curd();

d1.delete\_q(int.Parse(txtRoll.Text));

}

}

}

File Name – operation\_curd.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Data.SqlClient;

using System.Configuration;

using System.Data;

namespace WebApplication1

{

public class operation\_curd

{

SqlConnection con;

SqlCommand com;

public SqlDataReader dr;

public string insert\_q(int f\_roll, string f\_name, string f\_city)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("curd\_op", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", f\_roll);

com.Parameters.AddWithValue("Name1", f\_name);

com.Parameters.AddWithValue("City1", f\_city);

com.ExecuteNonQuery();

HttpContext.Current.Response.Write("Data Insert");

return "Insert";

}

public string find\_q(int f\_roll, string f\_name, string f\_city)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Find\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", f\_roll);

dr = com.ExecuteReader();

return "Done";

}

public string show\_q()

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("show\_class", con);

com.CommandType = CommandType.StoredProcedure;

dr = com.ExecuteReader();

return "Done";

}

public string update\_q(int f\_roll, string f\_name, string f\_city)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Update\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", f\_roll);

com.Parameters.AddWithValue("Name1", f\_name);

com.Parameters.AddWithValue("City1", f\_city);

com.ExecuteNonQuery();

HttpContext.Current.Response.Write("Data Update");

return "Done";

}

public string delete\_q(int f\_roll)

{

//Connection

string path = ConfigurationManager.AppSettings["crud"];

con = new SqlConnection(path);

con.Open();

//query

com = new SqlCommand("Delete\_curd1", con);

com.CommandType = CommandType.StoredProcedure;

com.Parameters.AddWithValue("Roll1", f\_roll);

com.ExecuteNonQuery();

HttpContext.Current.Response.Write("Data Delete");

return "Done";

}

}

}