**Aim:** Design a webpage to display the use of LINQ.

### Code:

1. Consider a scenario where you are developing a web application using ASP.NET and MySQL.

* Extract Data from data from a table named employees in your MySQL database with columns Fname,Lname,mobile,city Email into a gridview.
* Using this gridview extract data using LINQ to display only employees in Mumbai city

Webconfig file :

<connectionStrings>

<add name="constring" connectionString="Server=localhost;Database=awt;Uid=root;Pwd=root;" providerName="MySql.Data.MySqlClient" />

</connectionStrings>

webform1.aspx.cs:

using System;

using System.Collections.Generic; using System.Data.SqlClient; using System.Data;

using System.Linq; using System.Web; using System.Web.UI;

using System.Web.UI.WebControls; using System.Configuration;

using MySql.Data.MySqlClient;

namespace prac666

{

public partial class WebForm1 : System.Web.UI.Page

{

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

{

BindGridView(); ExtractAndBindFilteredData();

}

private void BindGridView()

{

string connectionString = ConfigurationManager.ConnectionStrings["constring"].ConnectionString;

string query = "SELECT Fname, Lname, mobile, city, email FROM employee";

using (MySqlConnection con = new MySqlConnection(connectionString))

{

MySqlCommand cmd = new MySqlCommand(query, con); con.Open();

DataTable dt = new DataTable(); dt.Load(cmd.ExecuteReader());

GridView1.DataSource = dt; GridView1.DataBind();

}

}

private void ExtractAndBindFilteredData()

{

// Get the data source of the source GridView

DataTable sourceDataTable = (DataTable)GridView1.DataSource;

// Optionally, perform LINQ operations on the source DataTable var filteredData = sourceDataTable.AsEnumerable()

.Where(row => row.Field<string>("City") == "New York")

CopyToDataTable();

// Bind the filtered data to the destination GridView GridView2.DataSource = filteredData; GridView2.DataBind();

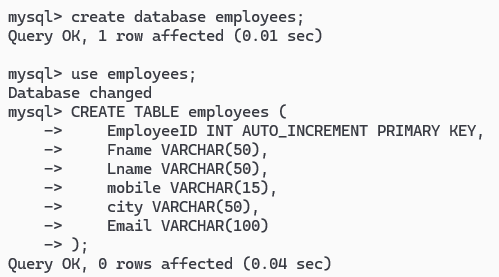
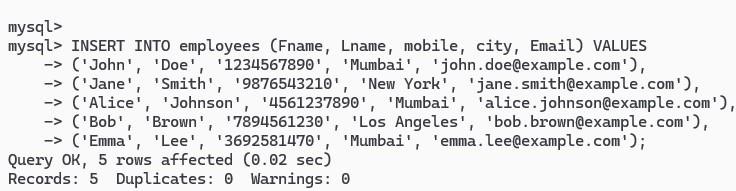
}

}

}

### Output:





1. Consider a scenario where you are developing a web application using ASP.NET and JSON

* Extract Data from data from a JSON String named Product having productname and type
* Extract data using LINQ to display only products of type Electronic Code:

using System;

using System.Collections.Generic; using System.Linq;

using System.Web.UI.WebControls; using ewtonsoft.Json;

namespace prac6\_2

{

public partial class Yourflage : System.Web.UI.flage

{

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

{

if (!IsflostBack)

{

string jsonString = @"

{

},

""electronic"" },

},

""electronic"" }, electronic"" }

}";

""products"": [

{ ""product ame"": ""Laptop"", ""type"": ""electronic""

{ ""product ame"": ""Smartphone"", ""type"":

{ ""product ame"": ""Book"", ""type"": ""non-electronic""

{ ""product ame"": ""Headphones"", ""type"":

{ ""product ame"": ""Watch"", ""type"": ""non-

]

// Deserialize JSO string into a list of flroduct objects List<flroduct> products =

JsonConvert.DeserializeObject<flroductList>(jsonString).flroducts;

// Filter products of type "electronic"

var electronicflroducts = products.Where(p => p.Type == "electronic").ToList();

// Bind the filtered data to the GridViewflroducts control GridViewflroducts.DataSource = electronicflroducts; GridViewflroducts.DataBind();

}

}

// Define a class to represent a flroduct public class flroduct

{

public string flroduct ame { get; set; } public string Type { get; set; }

}

// Define a class to represent the JSO structure public class flroductList

{

public List<flroduct> flroducts { get; set; }

}

}

}

Web.aspx

<%@ flage Language="C " AutoEventWireup="true" CodeBehind="Yourflage.aspx.cs" Inherits="prac6\_2.Yourflage" %>

<!DOCTYflE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<asp:GridView ID="GridViewflroducts" runat="server" AutoGenerateColumns="false">

<Columns>

<asp:BoundField DataField="flroduct ame" HeaderText="flroduct

ame" />

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

</Columns>

</asp:GridView>

</div>

</form>

</body>

</html>

### Output:

