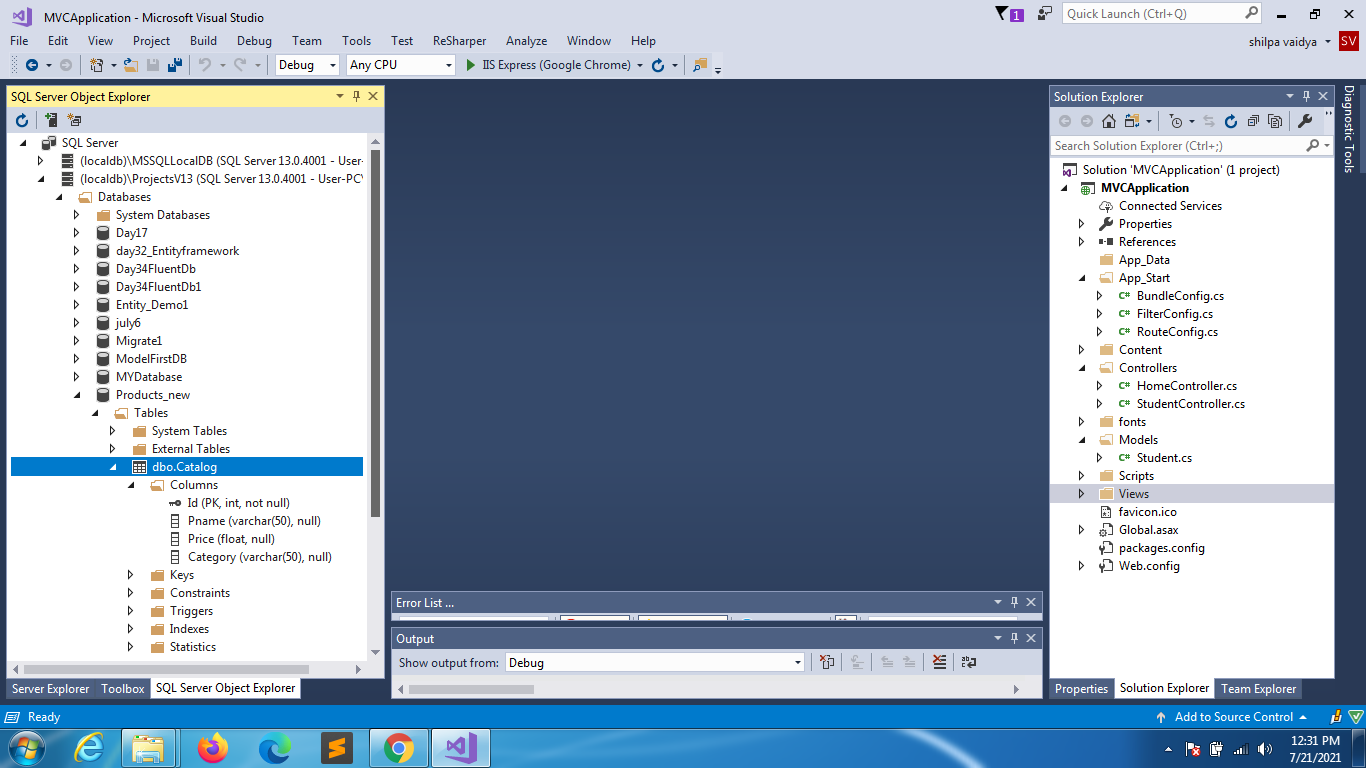
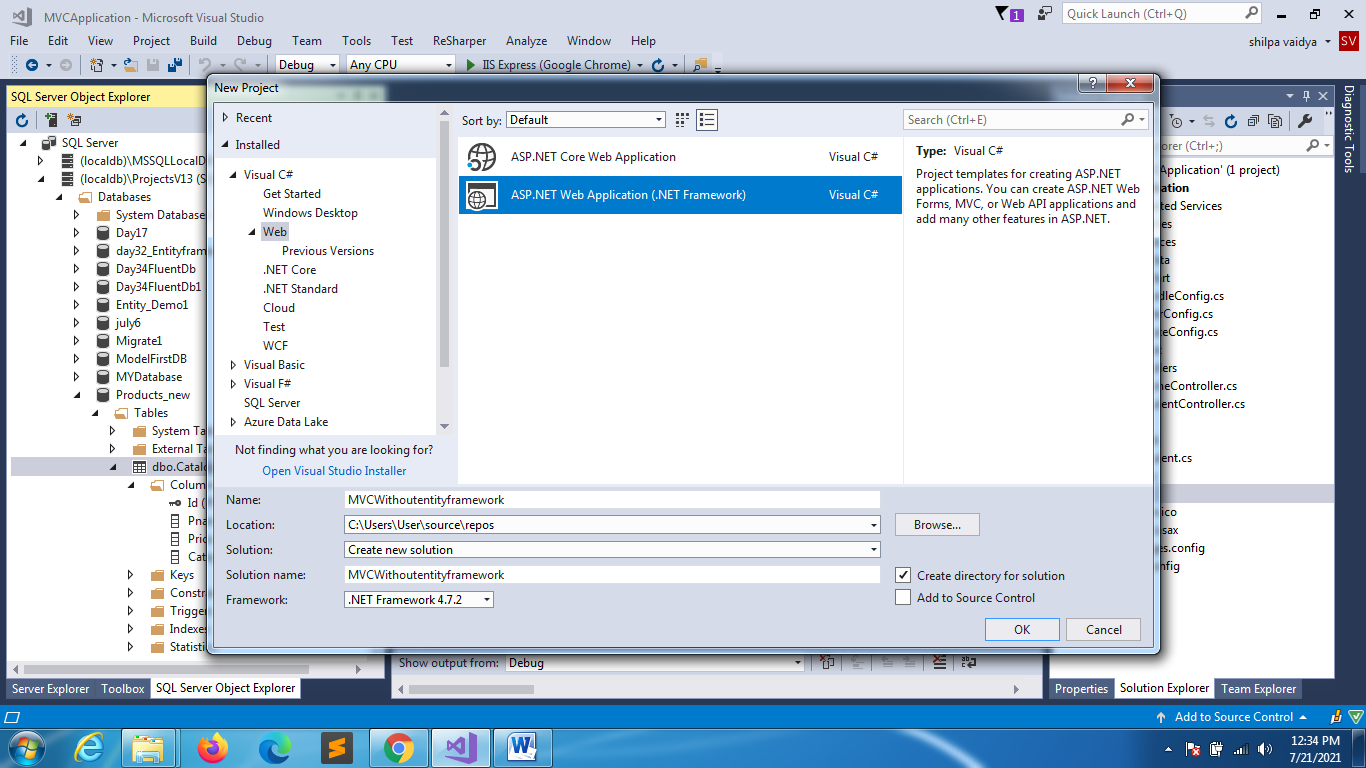
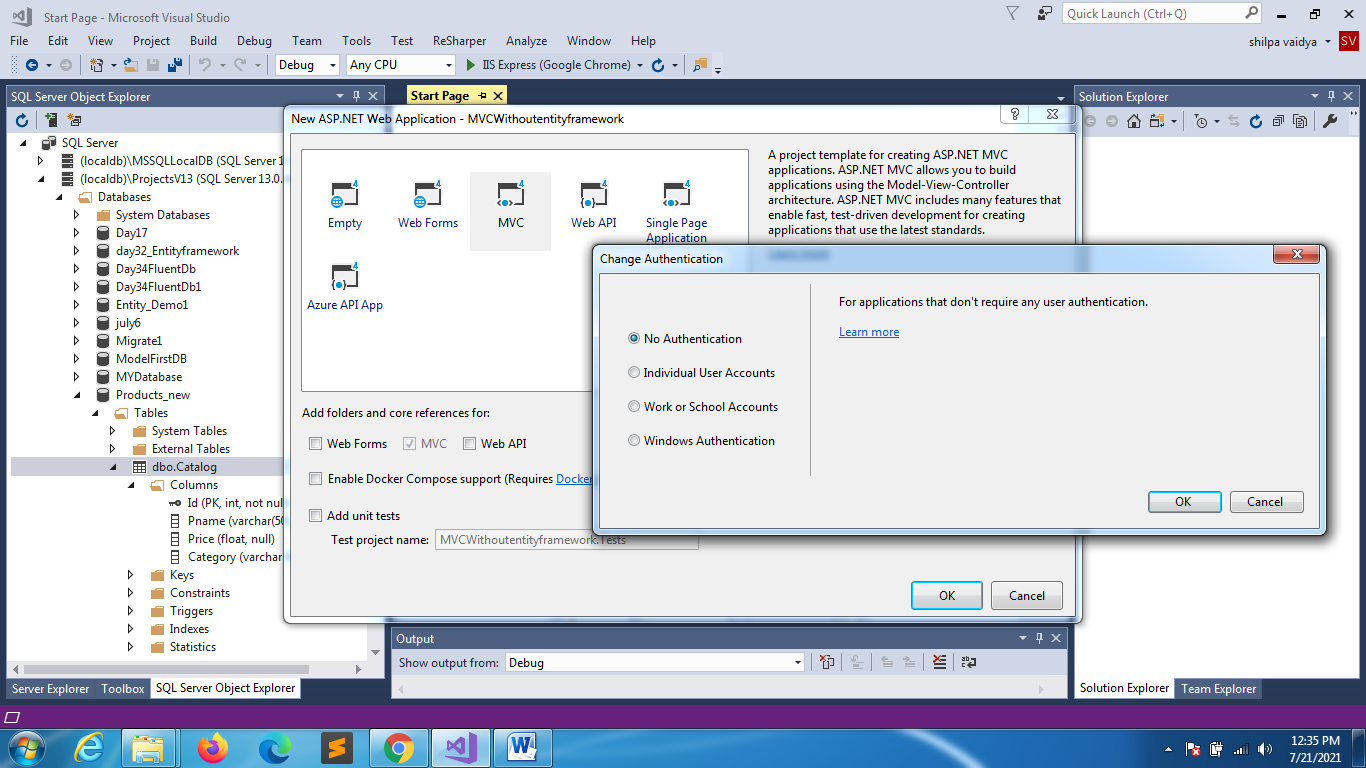
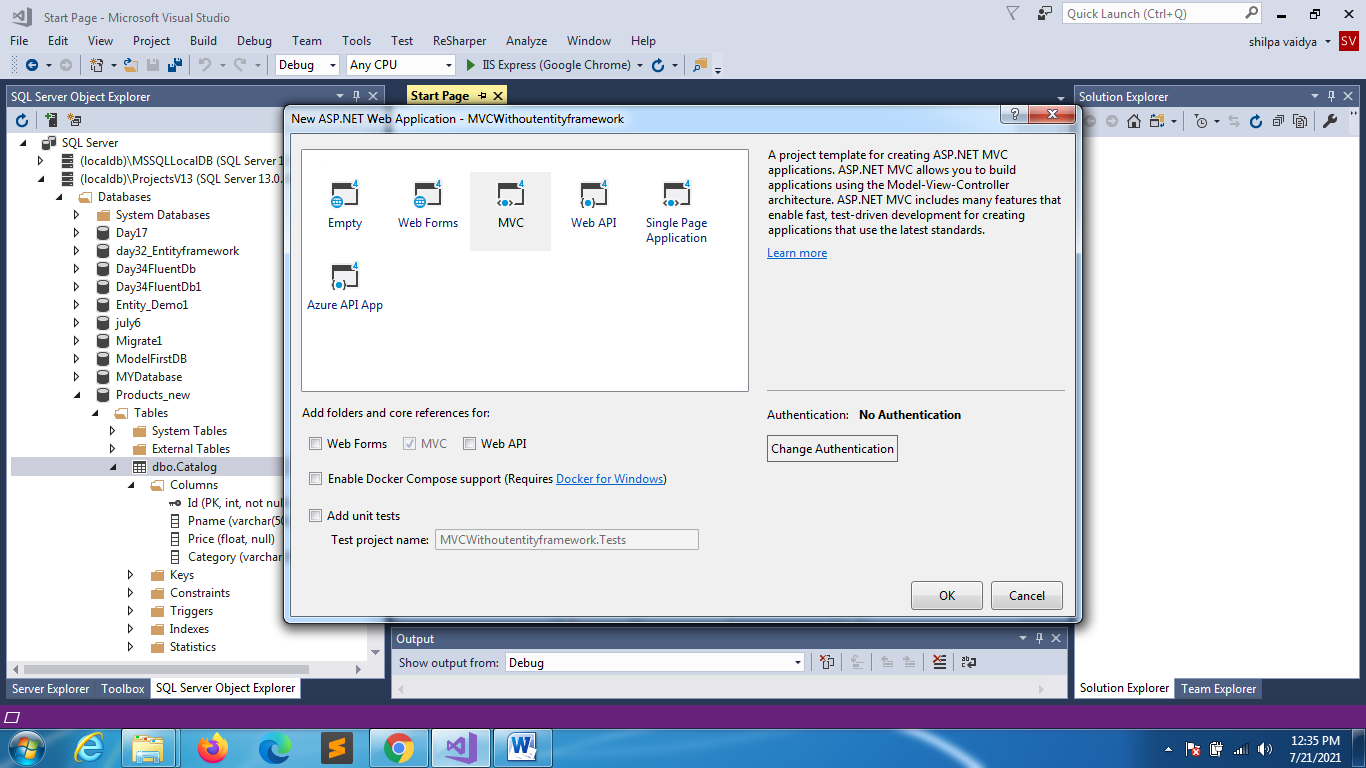
CREATE A DATABASE PRODUCTS NEW -> CATALOG WITH BELOW COLUMNS

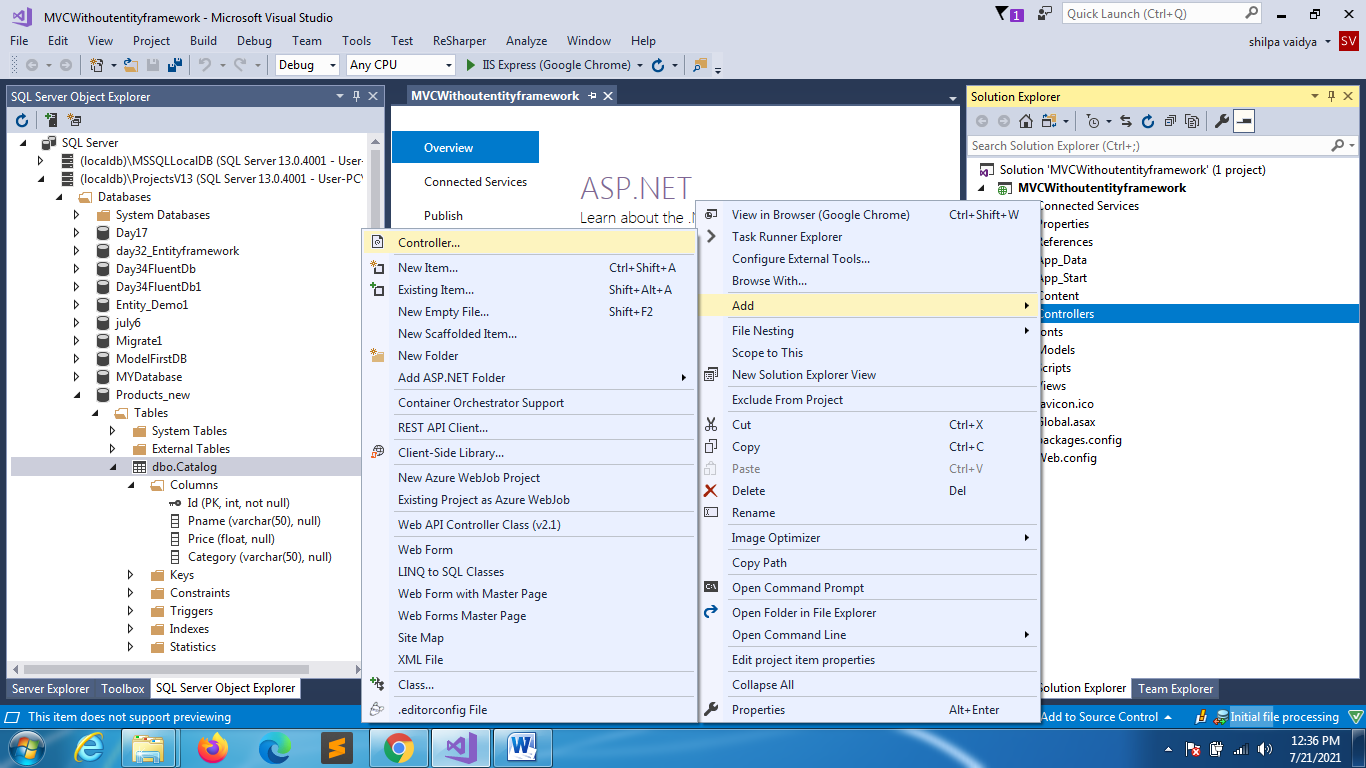


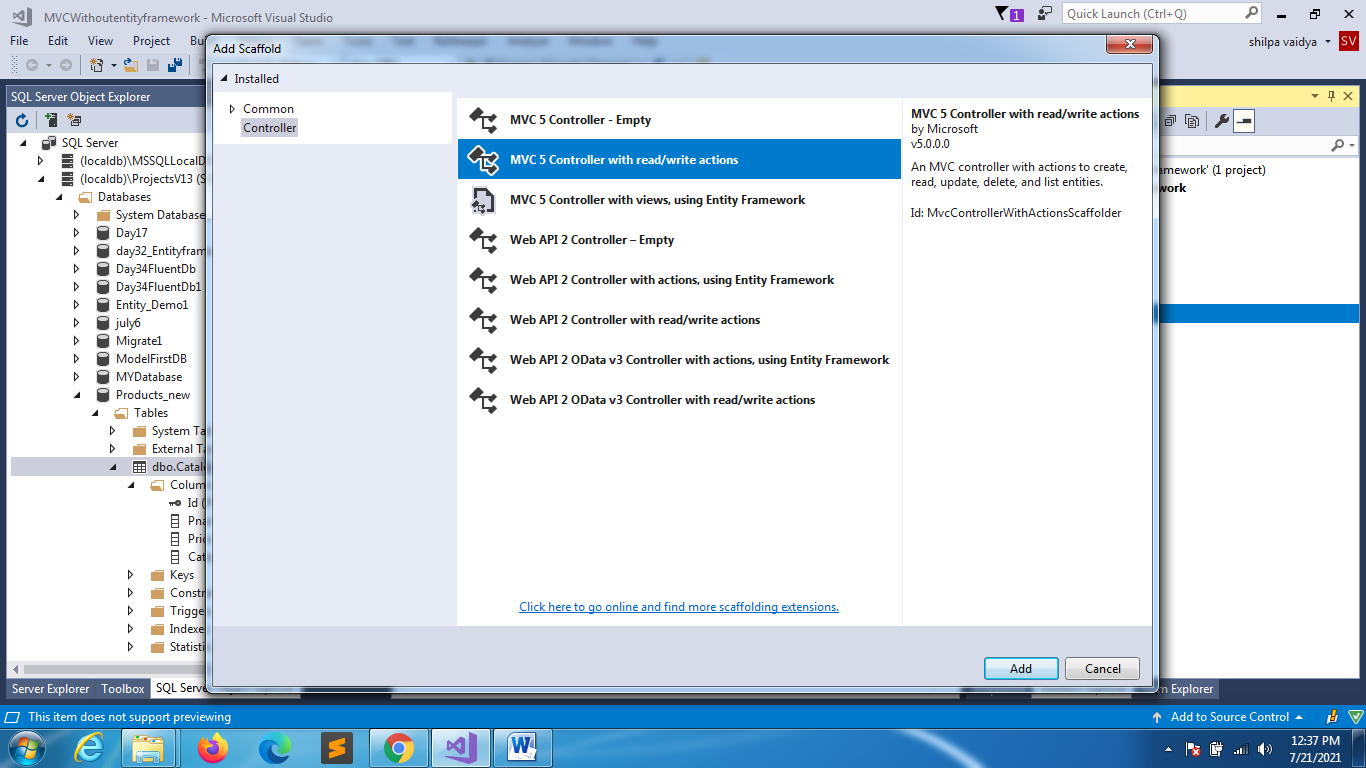


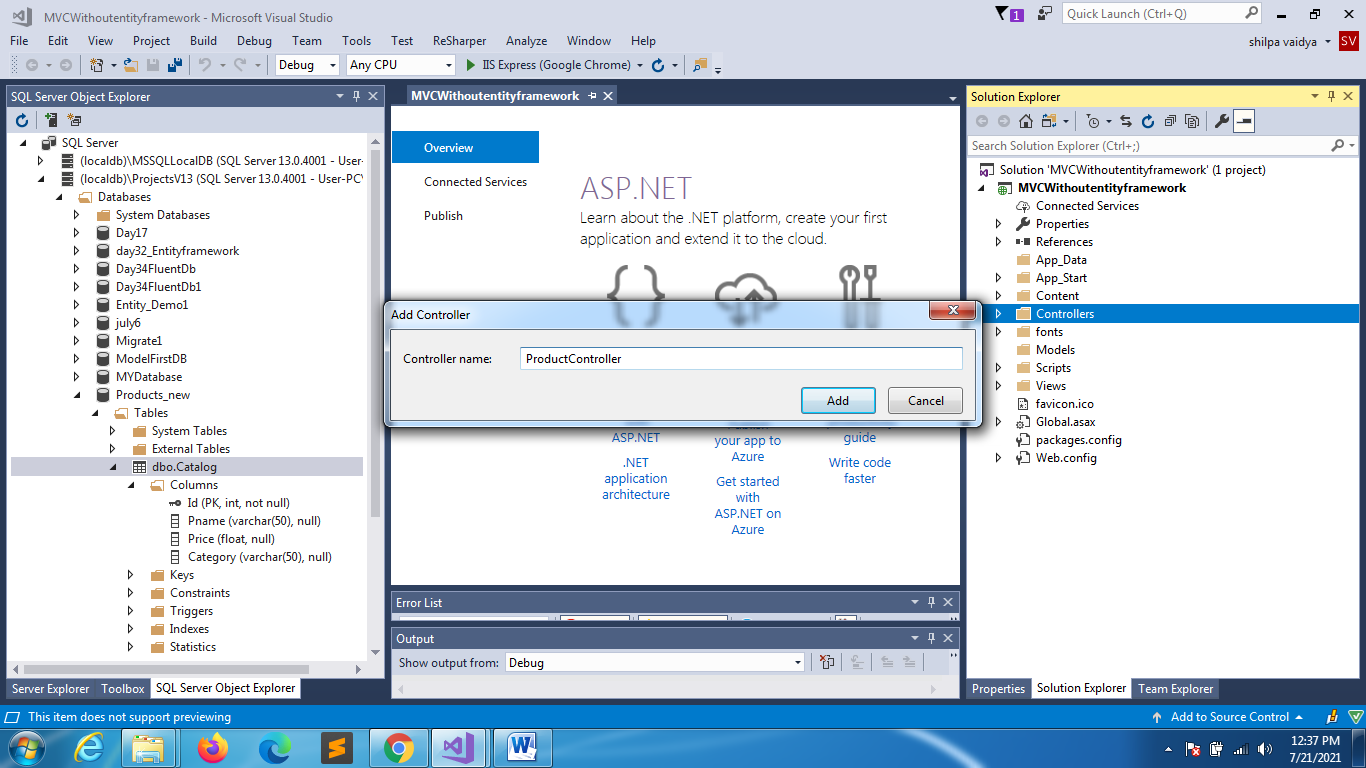




ADD CONTROLLERS







using System;

using System.Web.Mvc;

using System.Data;

using System.Data.SqlClient;

namespace MVCWithoutentityframework.Controllers

{

    public class ProductController : Controller

    {

        string connectionString = @"Data Source=(localdb)\ProjectsV13;Initial Catalog=Products\_new;Integrated Security=True;Connect Timeout=30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSubnetFailover=False";

        [HttpGet]

        public ActionResult Index()

After typing this much rightclick in the function and add view

        {

            DataTable dttbCatalog = new DataTable();

            using (SqlConnection conn = new SqlConnection(connectionString))

            {

                conn.Open();

                string sql = "select \* from Catalog";

                SqlDataAdapter da = new SqlDataAdapter(sql,conn);

                da.Fill(dttbCatalog);

            }

            return View(dttbCatalog);

        }

        // GET: Product/Details/5

        public ActionResult Details(int id)

        {

            return View();

        }

        // GET: Product/Create

        public ActionResult Create()

        {

            return View();

        }

        // POST: Product/Create

        [HttpPost]

        public ActionResult Create(FormCollection collection)

        {

            try

            {

                // TODO: Add insert logic here

                return RedirectToAction("Index");

            }

            catch

            {

                return View();

            }

        }

        // GET: Product/Edit/5

        public ActionResult Edit(int id)

        {

            return View();

        }

        // POST: Product/Edit/5

        [HttpPost]

        public ActionResult Edit(int id, FormCollection collection)

        {

            try

            {

                // TODO: Add update logic here

                return RedirectToAction("Index");

            }

            catch

            {

                return View();

            }

        }

        // GET: Product/Delete/5

        public ActionResult Delete(int id)

        {

            return View();

        }

        // POST: Product/Delete/5

        [HttpPost]

        public ActionResult Delete(int id, FormCollection collection)

        {

            try

            {

                // TODO: Add delete logic here

                return RedirectToAction("Index");

            }

            catch

            {

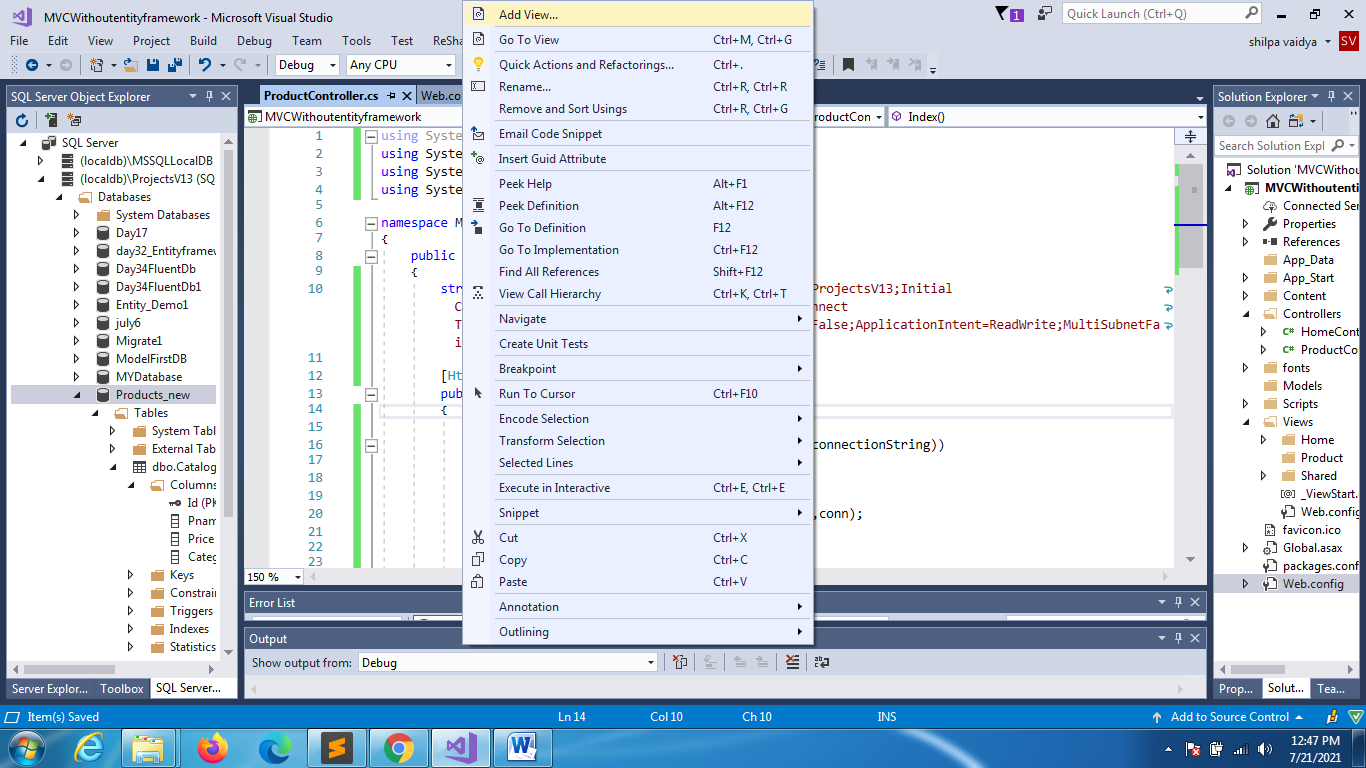
                return View();

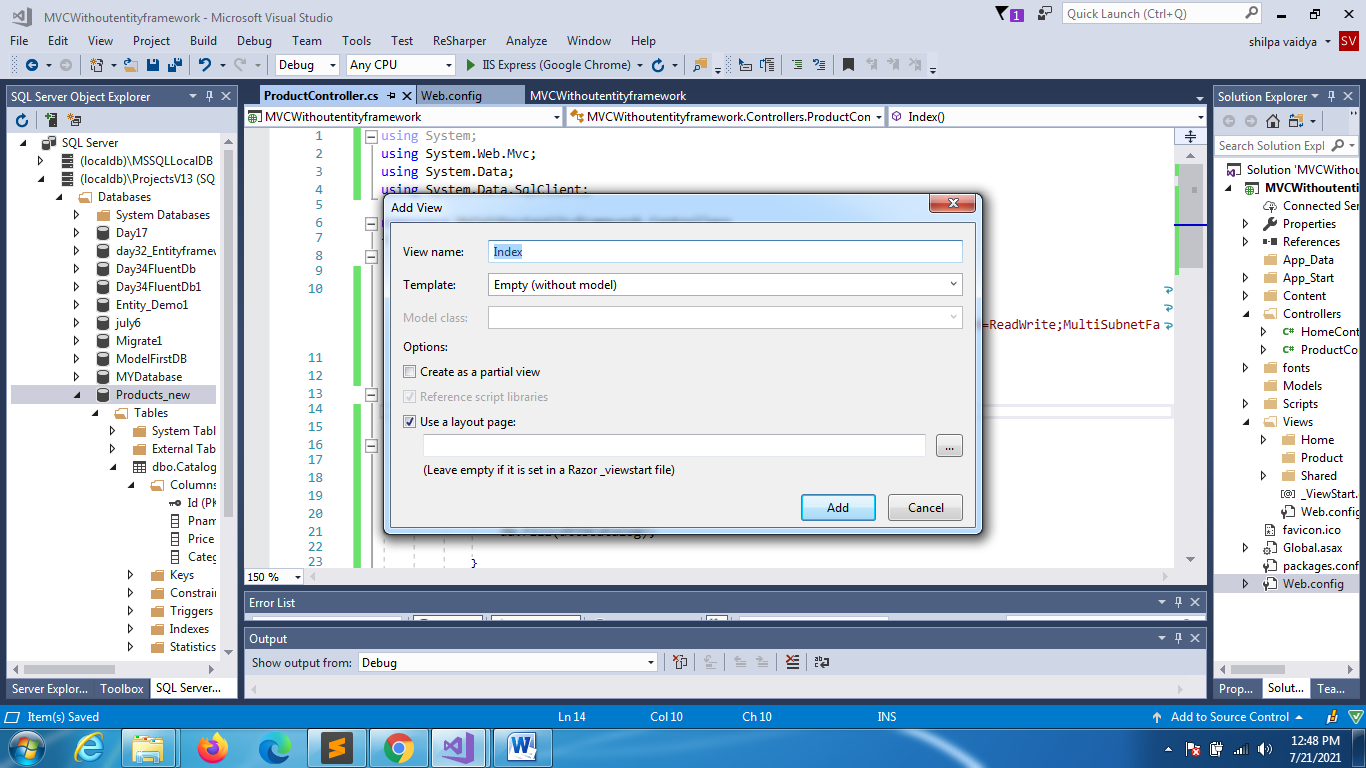
            }

        }

    }

}





@model System.Data.DataTable

@{

    ViewBag.Title = "Index";

}

<h2>Index</h2>

<table class="table table-bordered table-striped table-hover"   >

    <tr>

        <th>PRODUCT ID</th>

        <th>PRODUCT NAME</th>

        <th>PRODUCT PRICE</th>

        <th>PRODUCT CATEGORY</th>

    </tr>

    @for(int i=0;i<Model.Rows.Count; i++)

    {

    <tr>

        <td>@Model.Rows[i][0]</td>

        <td>@Model.Rows[i][1]</td>

        <td>@Model.Rows[i][2]</td>

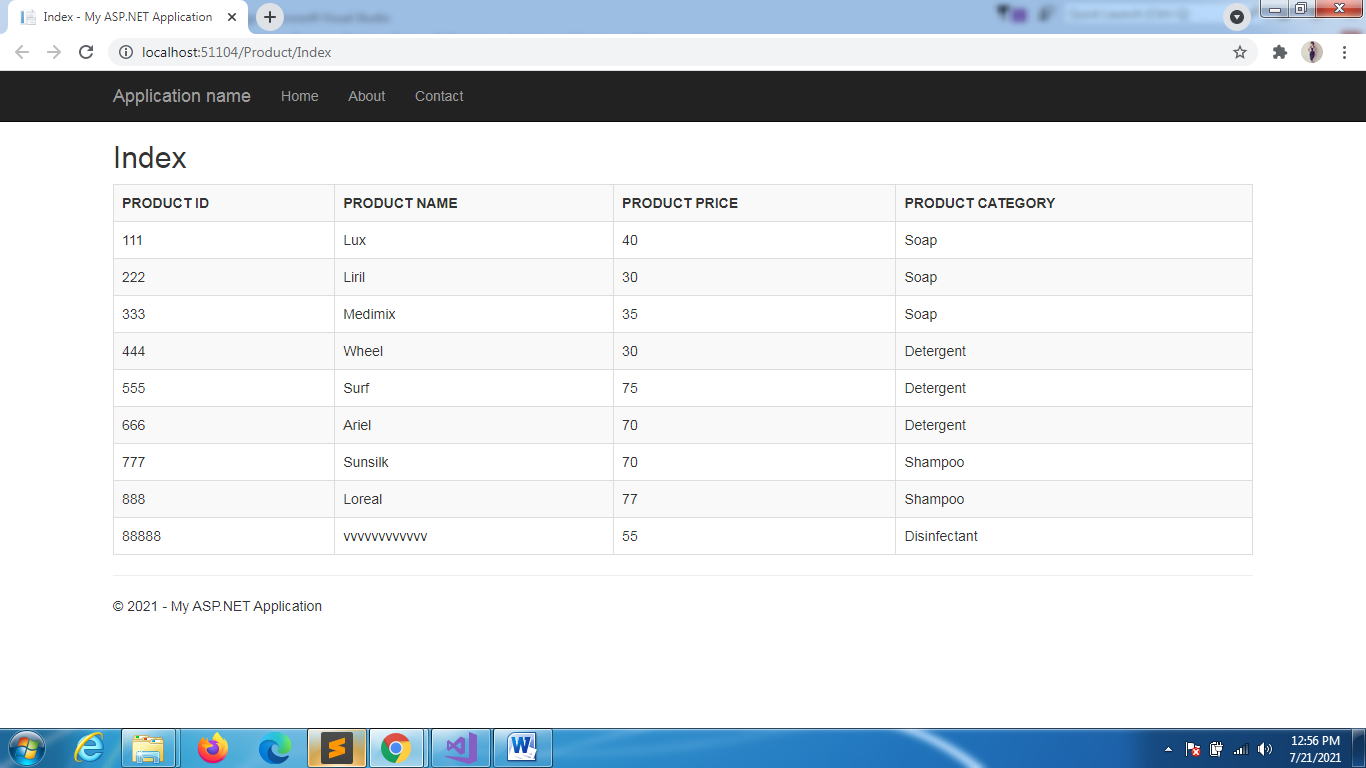
        <td>@Model.Rows[i][3]</td>

    </tr>

    }

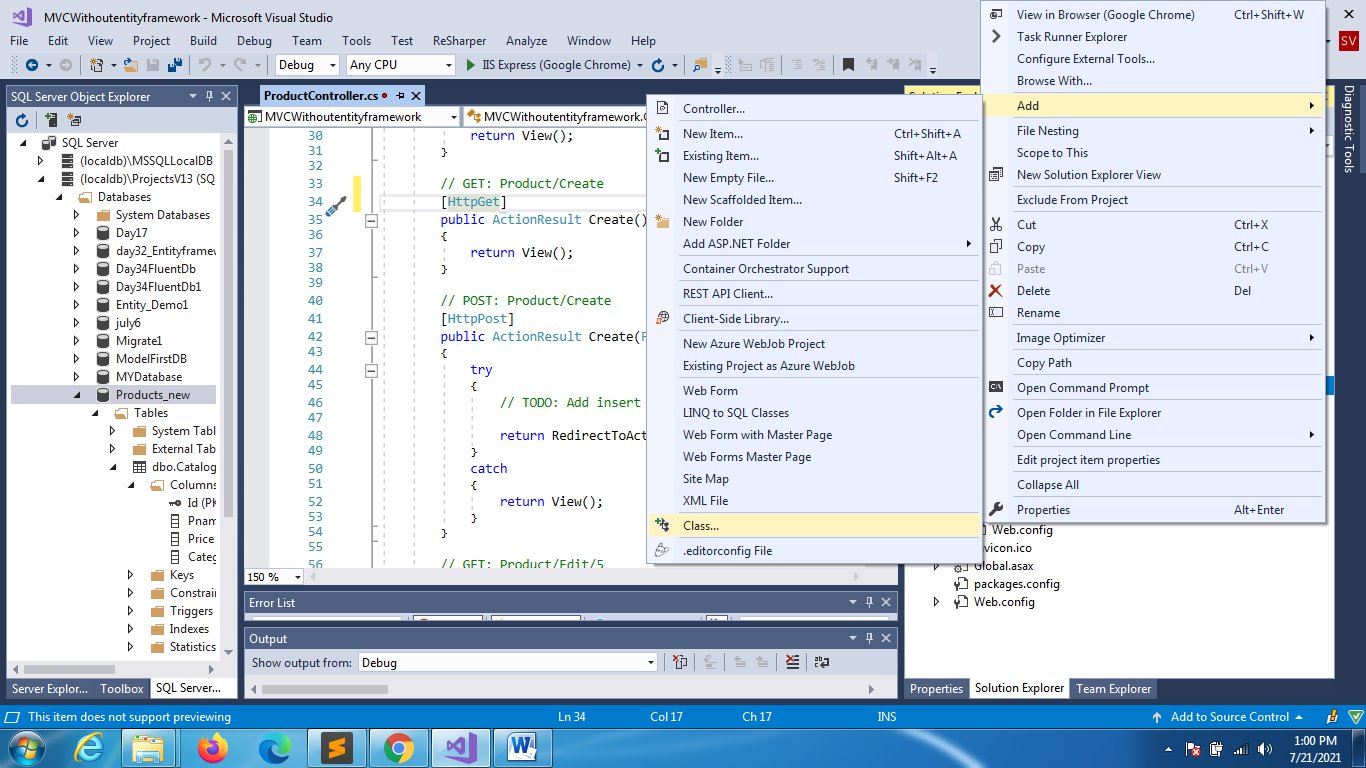
</table>

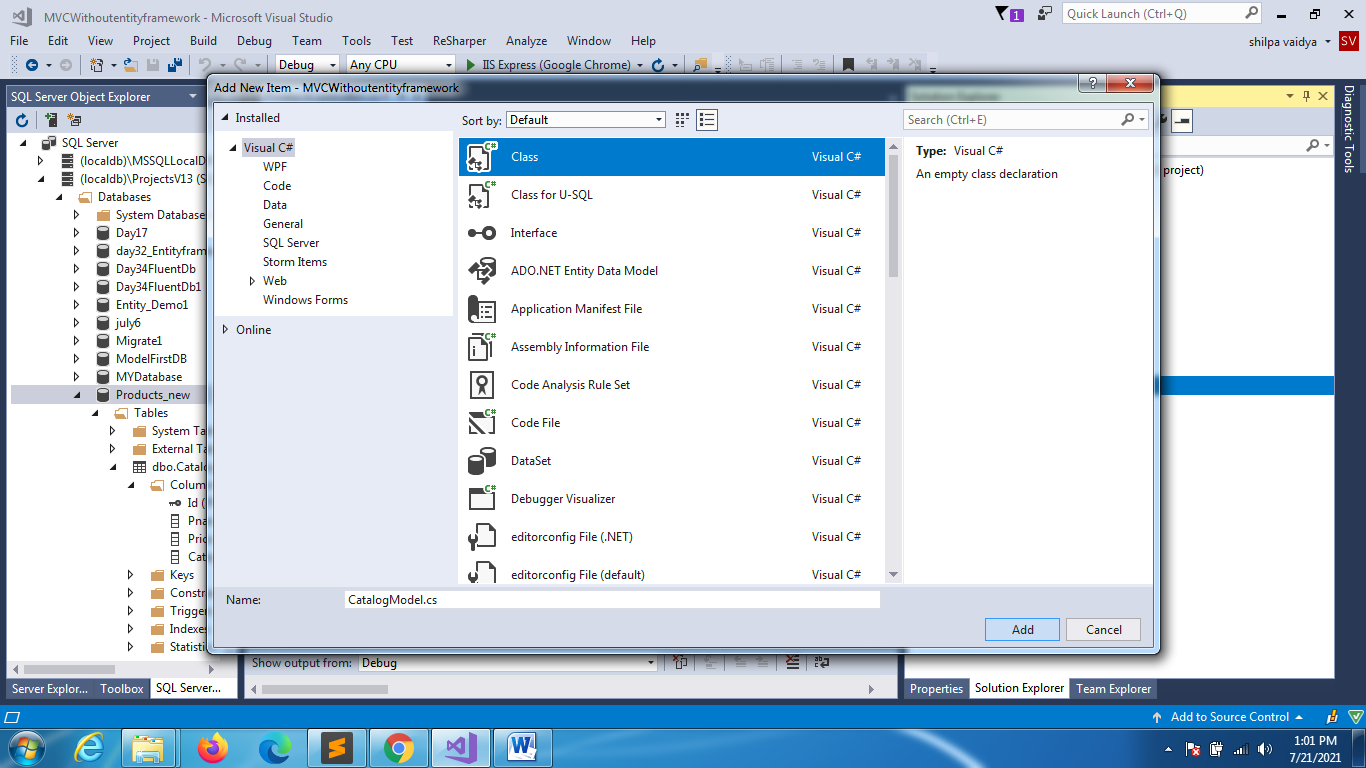
OUTPUT LOOKS LIKE THIS



CREATE METHOD(ACTION METHOD)

ADD A MODEL CLASS(right click model)





CATALOGMODEL

using System;

namespace MVCWithoutentityframework.Models

{

    public class CatalogModel

    {

        public int Id { get; set; }

        public string Pname { get; set; }

        public float Price { get; set; }

        public string Category { get; set; }

    }

}

ProductsController

// GET: Product/Create

       [HttpGet]

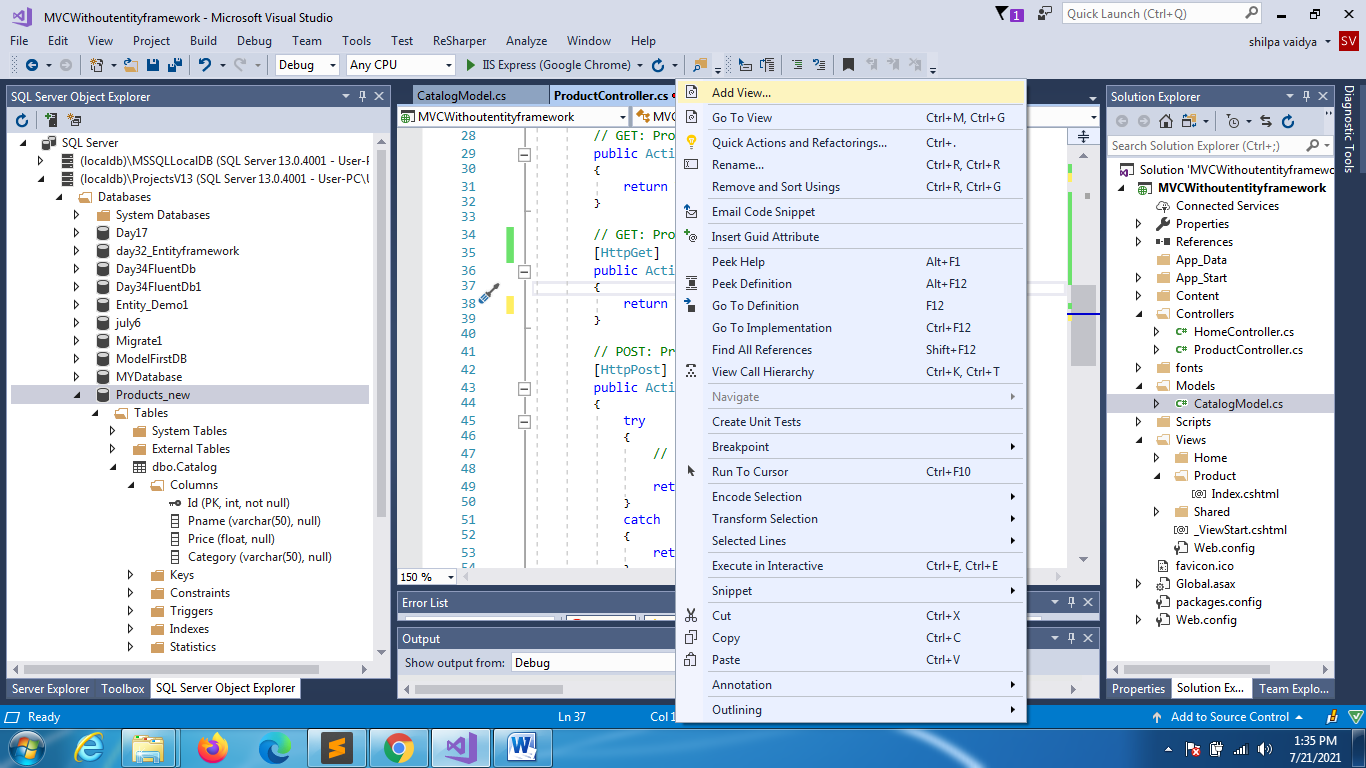
       public ActionResult Create()

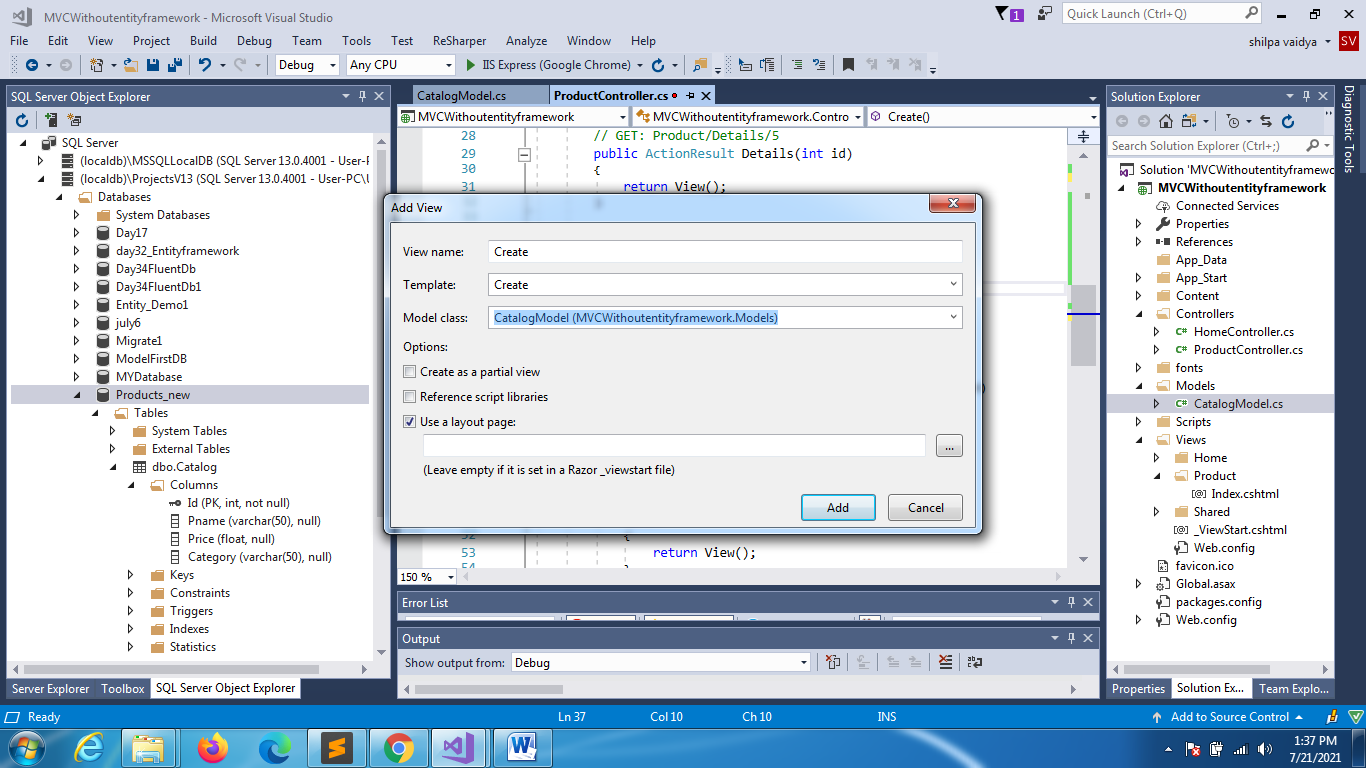
       {

           return View(new CatalogModel());

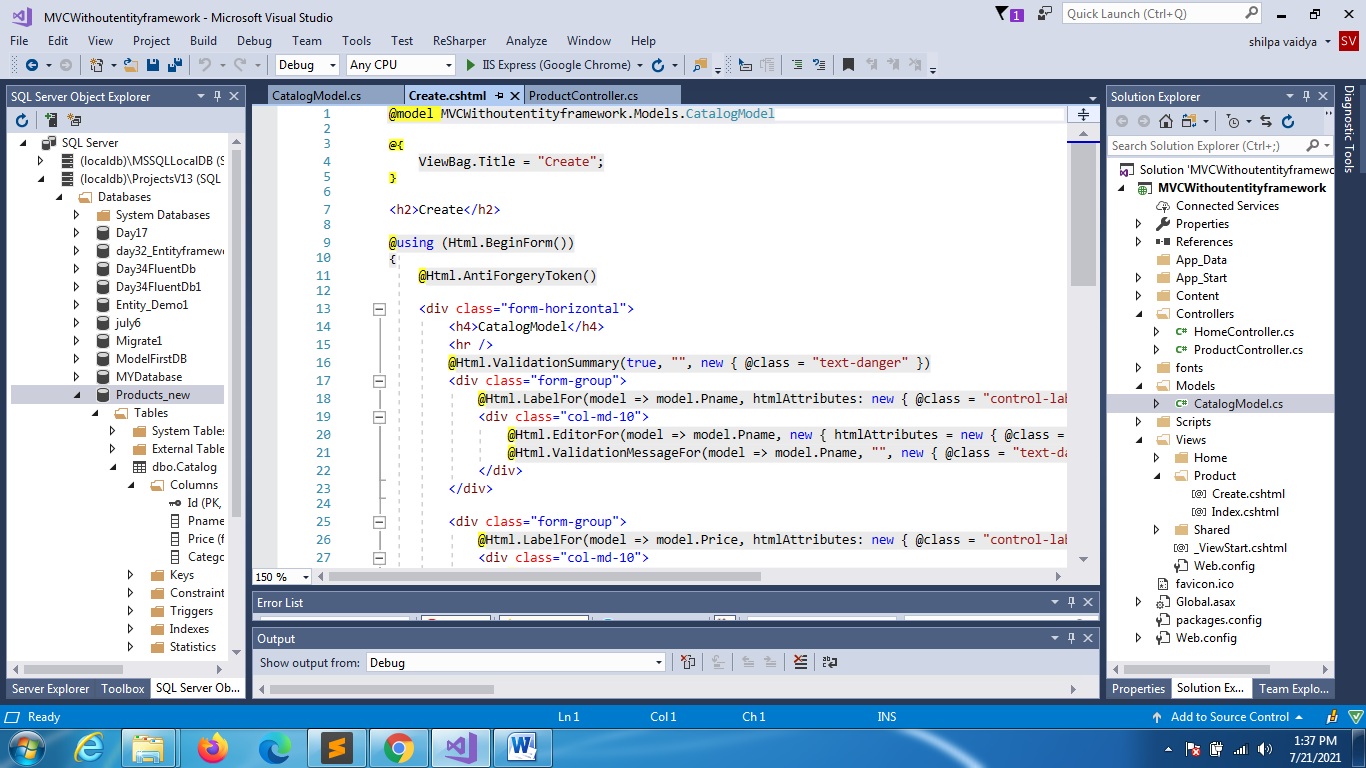
       }

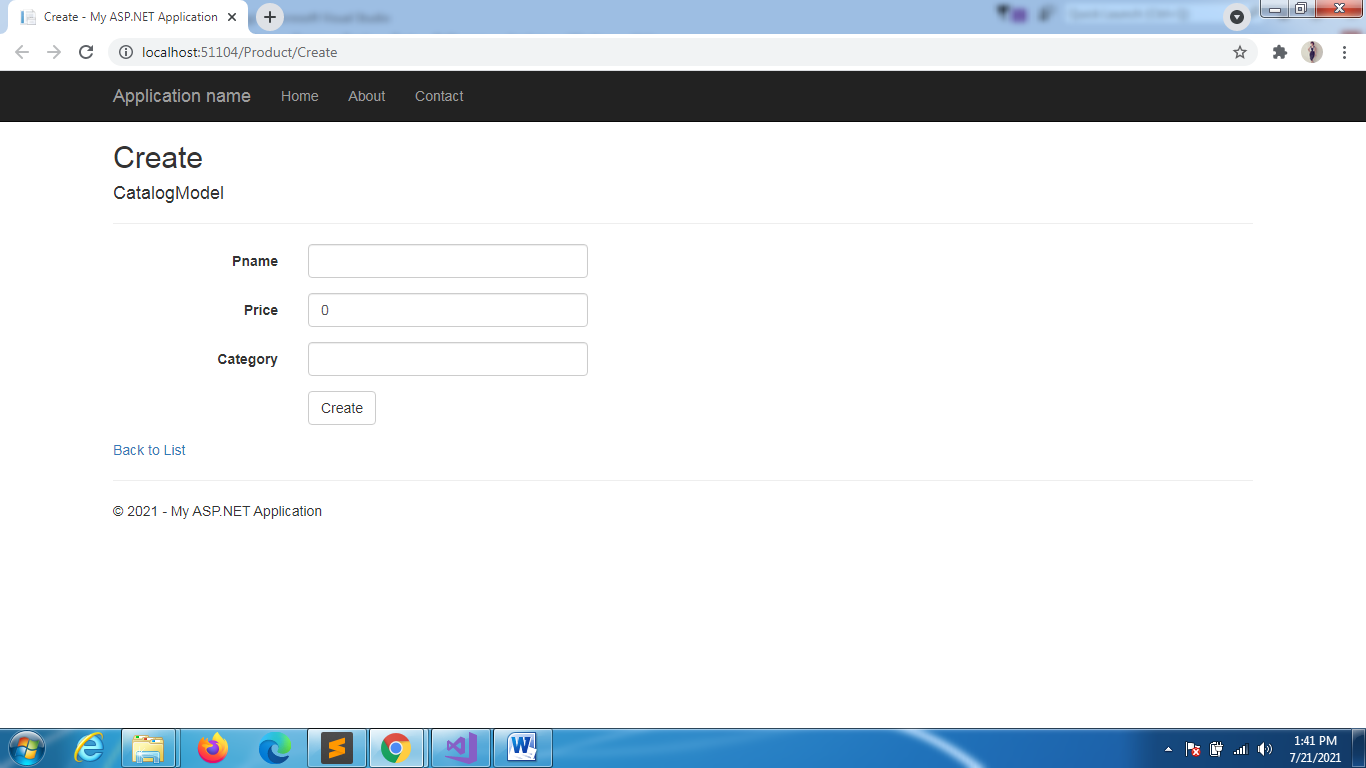
Rightclick inside create and add view



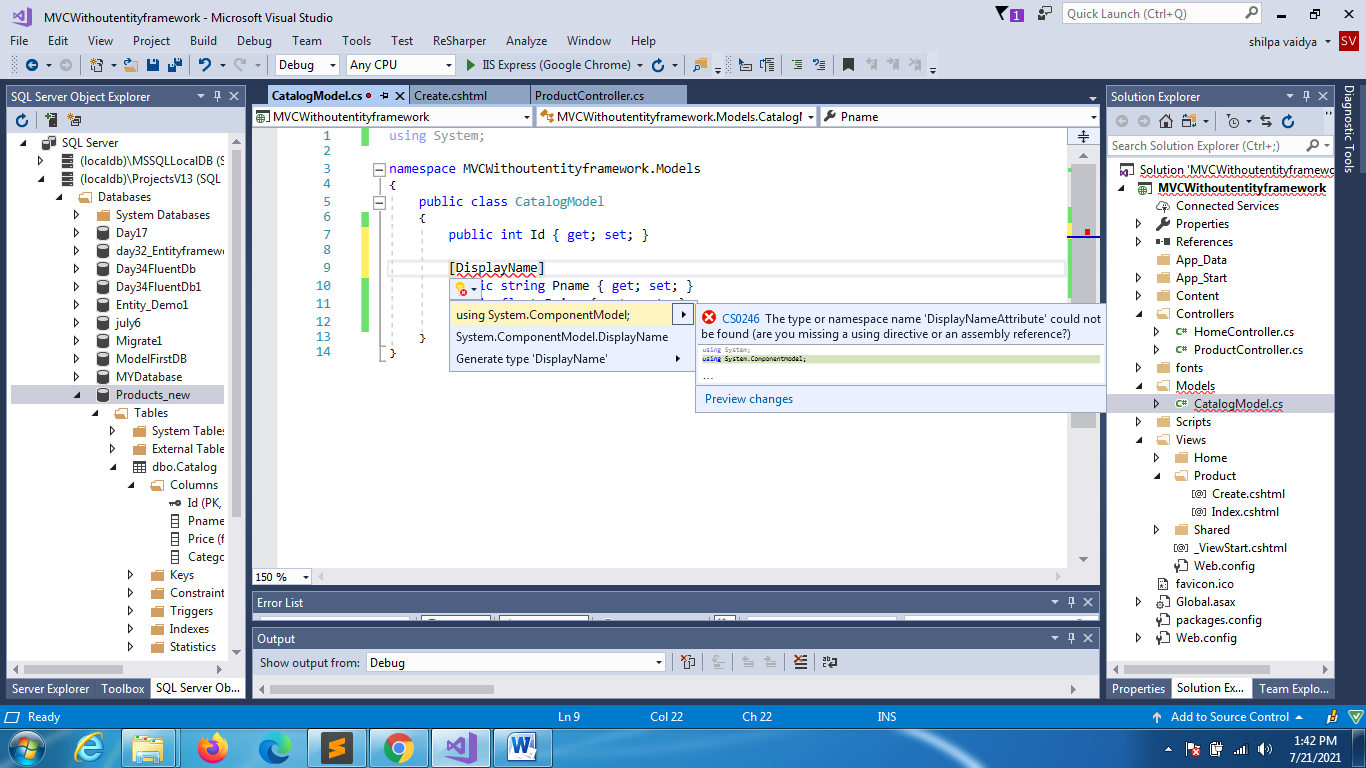


New view will be created





CatalogModel



using System;

using System.ComponentModel;

namespace MVCWithoutentityframework.Models

{

    public class CatalogModel

    {

        public int Id { get; set; }

        [DisplayName("Product Name")]

        public string Pname { get; set; }

        public float Price { get; set; }

        public string Category { get; set; }

    }

}

Link Index page to create

@model System.Data.DataTable

@{

    ViewBag.Title = "Index";

}

<h2>Index</h2>

<table class="table table-bordered table-striped table-hover"   >

    <tr>

        <th>PRODUCT ID</th>

        <th>PRODUCT NAME</th>

        <th>PRODUCT PRICE</th>

        <th>PRODUCT CATEGORY</th>

    </tr>

    @for(int i=0;i<Model.Rows.Count; i++)

    {

    <tr>

        <td>@Model.Rows[i][0]</td>

        <td>@Model.Rows[i][1]</td>

        <td>@Model.Rows[i][2]</td>

        <td>@Model.Rows[i][3]</td>

    </tr>

    }

</table>

<a href="@Url.Action("Create","Product")">ADD PRODUCTS</a>

The next create method is by post method(that has HttpPost) modify that

// POST: Product/Create

       [HttpPost]

       public ActionResult Create(CatalogModel catalogModel)

       {

           using (SqlConnection conn = new SqlConnection(connectionString))

           {

               conn.Open();

               string sql = "insert into Catalog Values(@Id,@Pname,@Price,@Category)";

               SqlCommand cmd = new SqlCommand(sql, conn);

               cmd.Parameters.AddWithValue("@Id", catalogModel.Id);

               cmd.Parameters.AddWithValue("@Pname", catalogModel.Pname);

               cmd.Parameters.AddWithValue("@Price", catalogModel.Price);

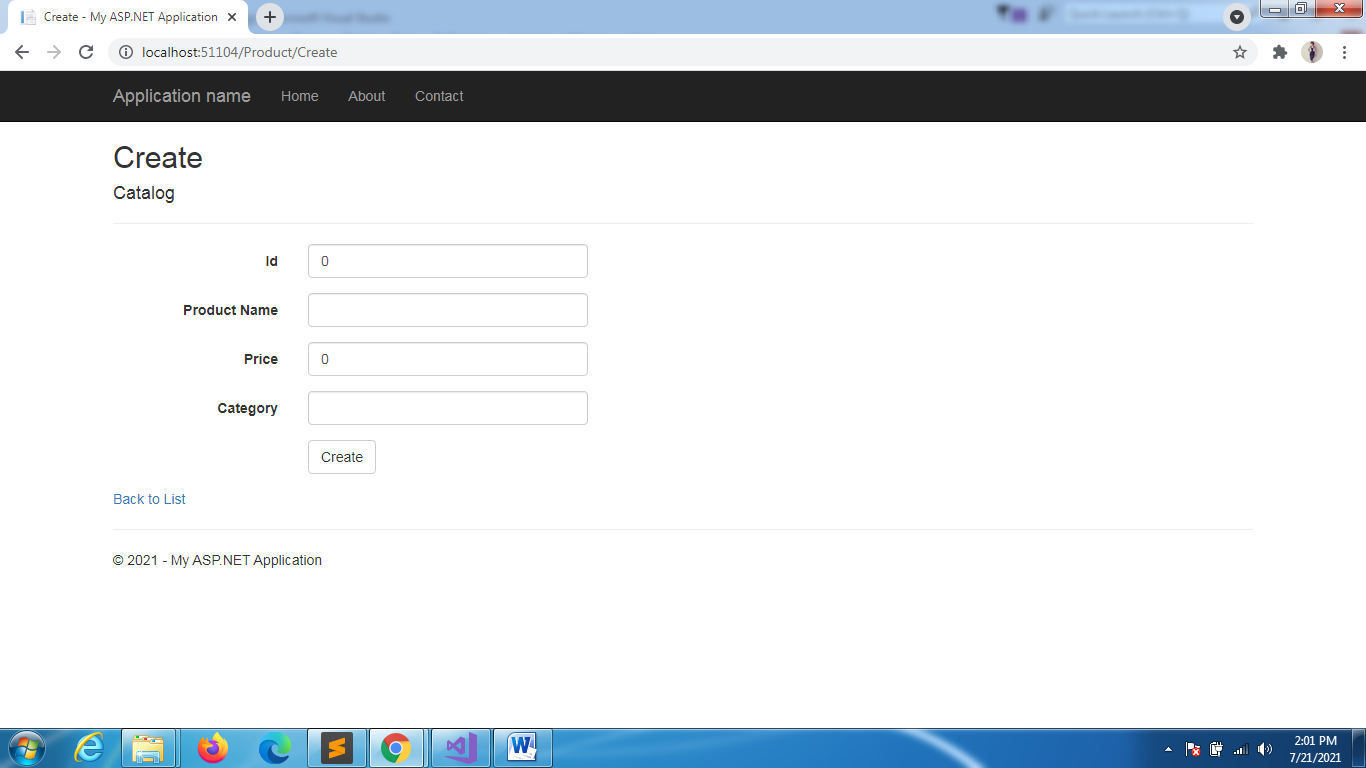
               cmd.Parameters.AddWithValue("@Category", catalogModel.Category);

               cmd.ExecuteNonQuery();

           }

           return RedirectToAction("Index");

       }



//GET EDIT(int id)

// GET: Product/Edit/5

public ActionResult Edit(int id)

{

    CatalogModel catalogModel = new CatalogModel();

    DataTable dt = new DataTable();

    using (SqlConnection conn = new SqlConnection(connectionString))

    {

        conn.Open();

         string sql = "select \* from Catalog where Id=@Id";

         SqlDataAdapter da = new SqlDataAdapter(sql,conn);

         da.SelectCommand.Parameters.AddWithValue("@Id", id);

         da.Fill(dt);

        if(dt.Rows.Count==1)

        {

            catalogModel.Id = Convert.ToInt32(dt.Rows[0][0].ToString());

            catalogModel.Pname = dt.Rows[0][1].ToString();

            catalogModel.Price  = float.Parse(dt.Rows[0][2].ToString());

            catalogModel.Category = dt.Rows[0][3].ToString();

Right click and add new view

            return View(catalogModel);

        }

        else

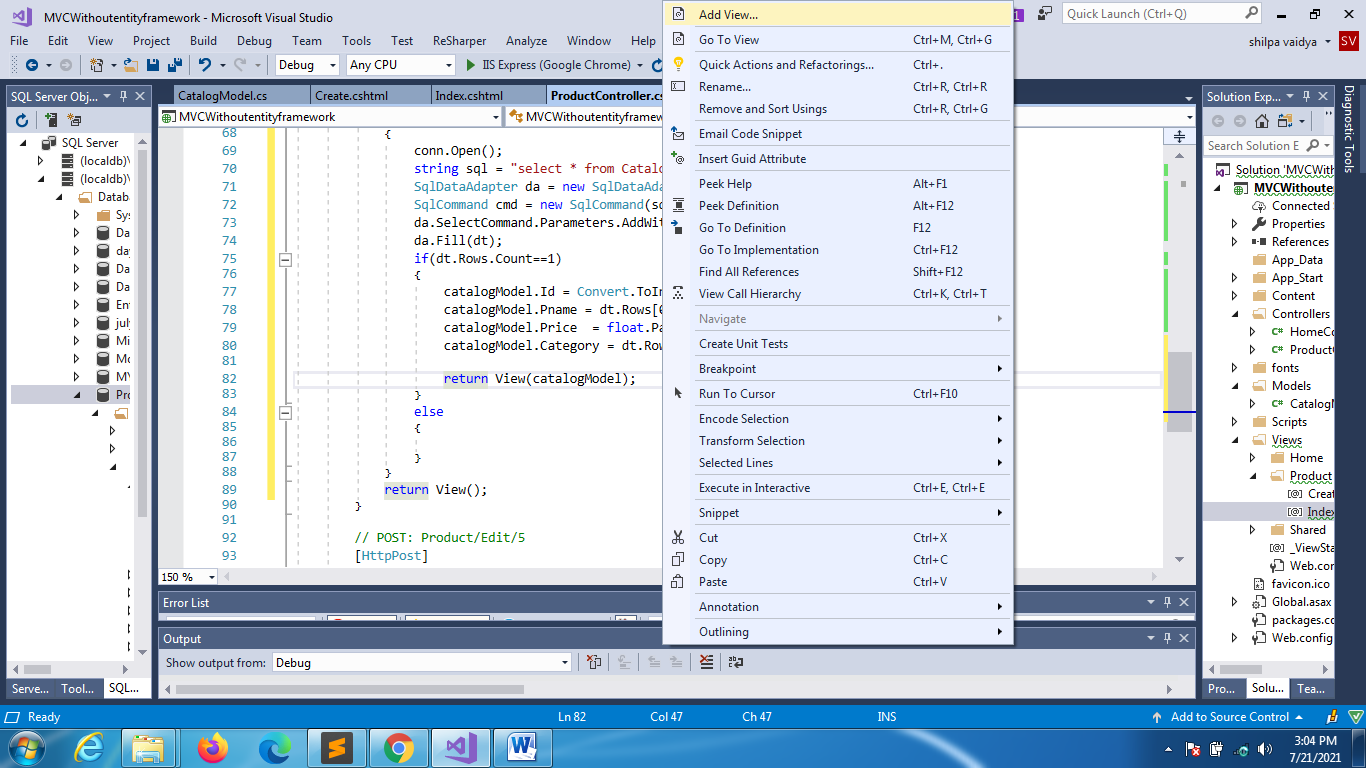
        {

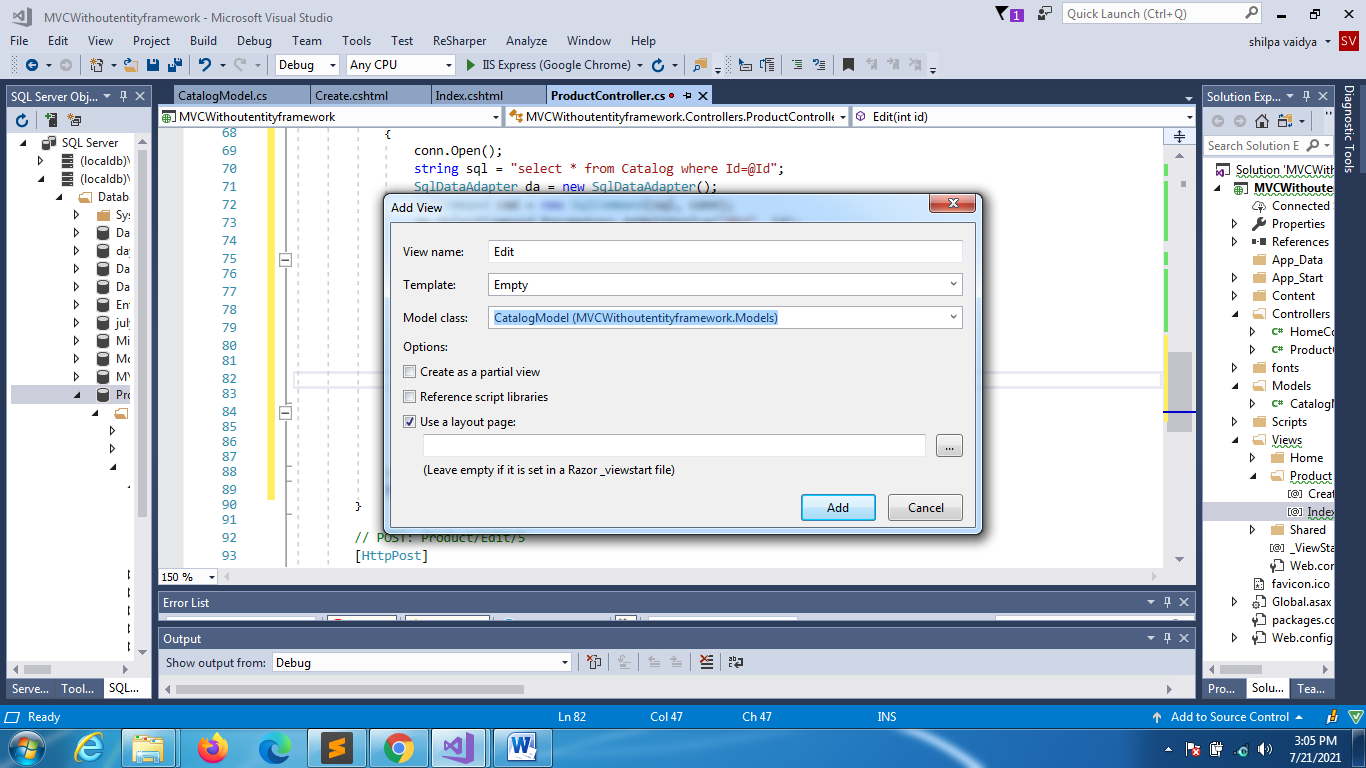
        }

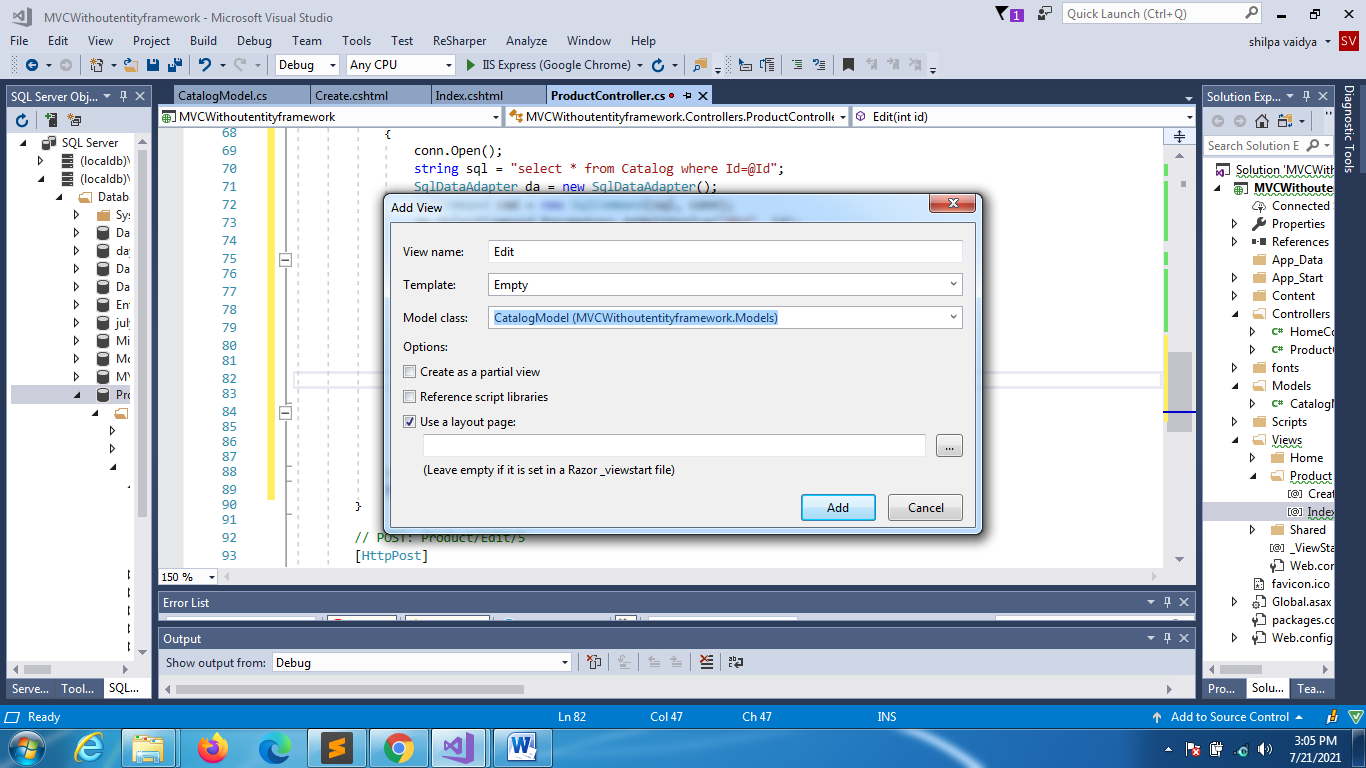
    }

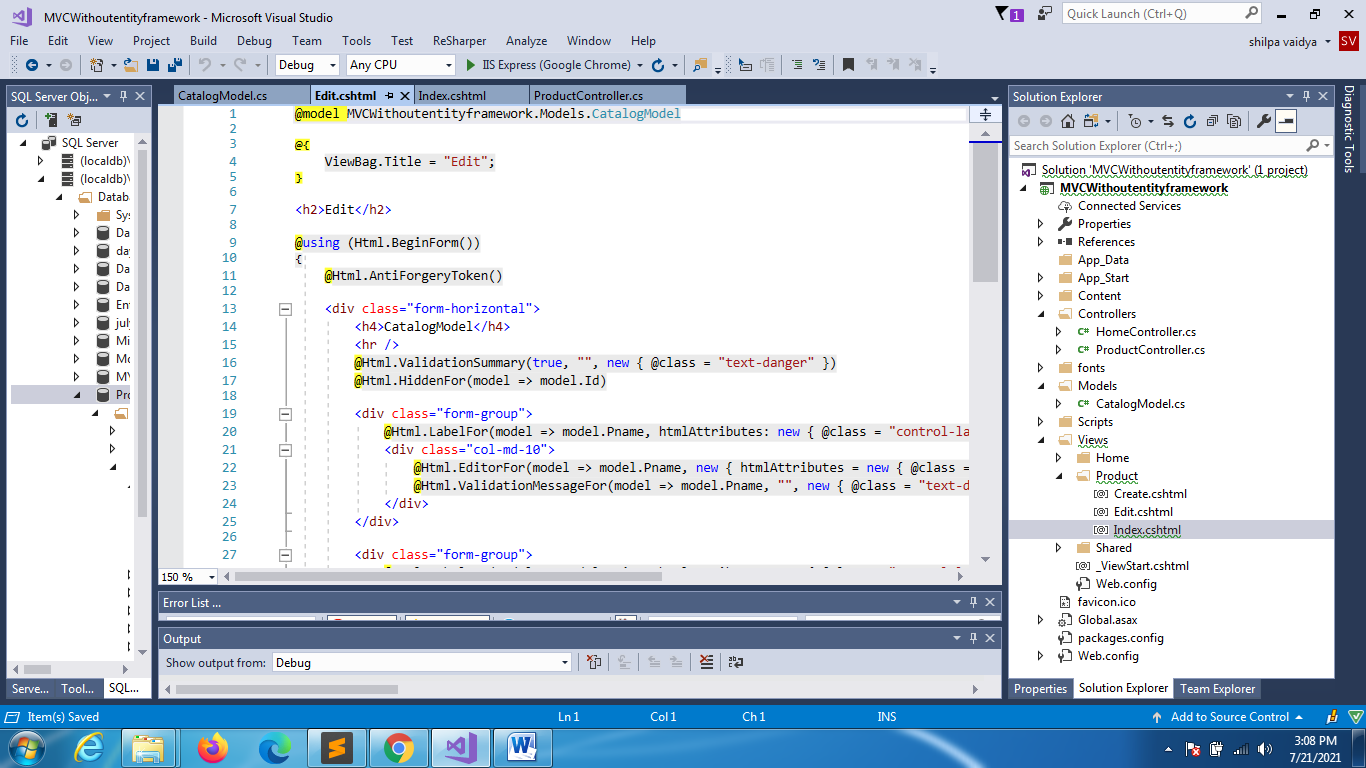
    return View();

}









Change index.cshtml

@model System.Data.DataTable

@{

    ViewBag.Title = "Index";

}

<h2>Index</h2>

<table class="table table-bordered table-striped table-hover"   >

    <tr>

        <th>PRODUCT ID</th>

        <th>PRODUCT NAME</th>

        <th>PRODUCT PRICE</th>

        <th>PRODUCT CATEGORY</th>

        <th></th>

    </tr>

    @for (int i = 0; i < Model.Rows.Count; i++)

    {

    <tr>

        <td>@Model.Rows[i][0]</td>

        <td>@Model.Rows[i][1]</td>

        <td>@Model.Rows[i][2]</td>

        <td>@Model.Rows[i][3]</td>

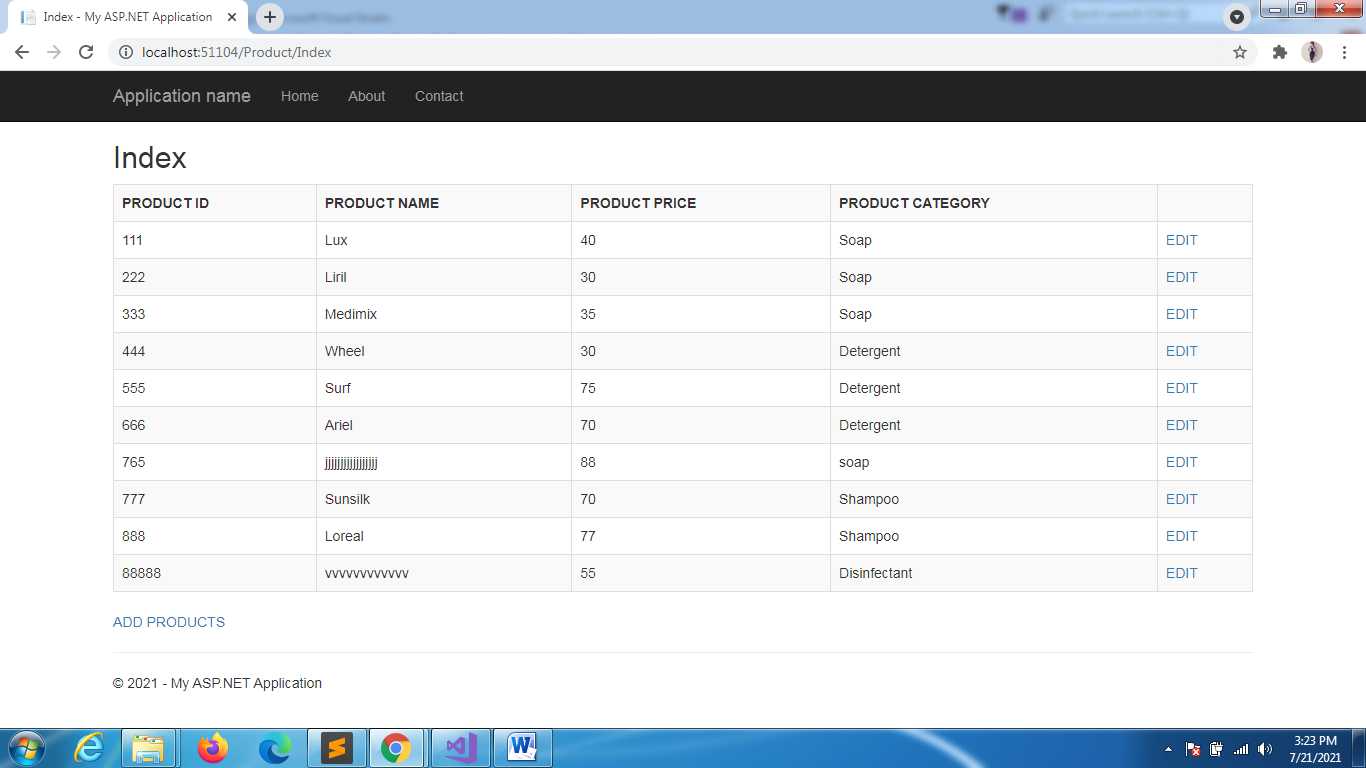
        <td><a href="@Url.Action("Edit", "Product", new {@id=@Model.Rows[i][0]})"> EDIT </a></td>

    </tr>

    }

</table>

<a href="@Url.Action("Create","Product")">ADD PRODUCTS</a>



// POST: Product/Edit/5

      [HttpPost]

      public ActionResult Edit(int id, CatalogModel catalogModel)

      {

          using (SqlConnection conn = new SqlConnection(connectionString))

          {

              conn.Open();

              string sql = "update Catalog set Pname=@Pname,Price=@Price,Category=@Category where Id=@Id";

              SqlCommand cmd = new SqlCommand(sql, conn);

              cmd.Parameters.AddWithValue("@Id", catalogModel.Id);

              cmd.Parameters.AddWithValue("@Pname", catalogModel.Pname);

              cmd.Parameters.AddWithValue("@Price", catalogModel.Price);

              cmd.Parameters.AddWithValue("@Category", catalogModel.Category);

              cmd.ExecuteNonQuery();

          }

          return RedirectToAction("Index");

      }

Delete

// GET: Product/Delete/5

      public ActionResult Delete(int id)

      {

          using (SqlConnection conn = new SqlConnection(connectionString))

          {

              conn.Open();

              string sql = "delete from Catalog  where Id=@Id";

              SqlCommand cmd = new SqlCommand(sql, conn);

              cmd.Parameters.AddWithValue("@Id", id);

              cmd.ExecuteNonQuery();

          }

          return RedirectToAction("Index");

      }

     // we dont need post action delete

Index changes

@model System.Data.DataTable

@{

    ViewBag.Title = "Index";

}

<h2>Index</h2>

<table class="table table-bordered table-striped table-hover"   >

    <tr>

        <th>PRODUCT ID</th>

        <th>PRODUCT NAME</th>

        <th>PRODUCT PRICE</th>

        <th>PRODUCT CATEGORY</th>

        <th></th>

    </tr>

    @for (int i = 0; i < Model.Rows.Count; i++)

    {

    <tr>

        <td>@Model.Rows[i][0]</td>

        <td>@Model.Rows[i][1]</td>

        <td>@Model.Rows[i][2]</td>

        <td>@Model.Rows[i][3]</td>

        <td>

        <a href="@Url.Action("Edit", "Product", new {@id=@Model.Rows[i][0]})"> EDIT </a> |

        <a href="@Url.Action("Delete", "Product", new {@id=@Model.Rows[i][0]})"> Delete </a>

        </td>

    </tr>

    }

</table>

<a href="@Url.Action("Create","Product")">ADD PRODUCTS</a>

