

Name: Solanki Pooja

Enrollment: 240823101

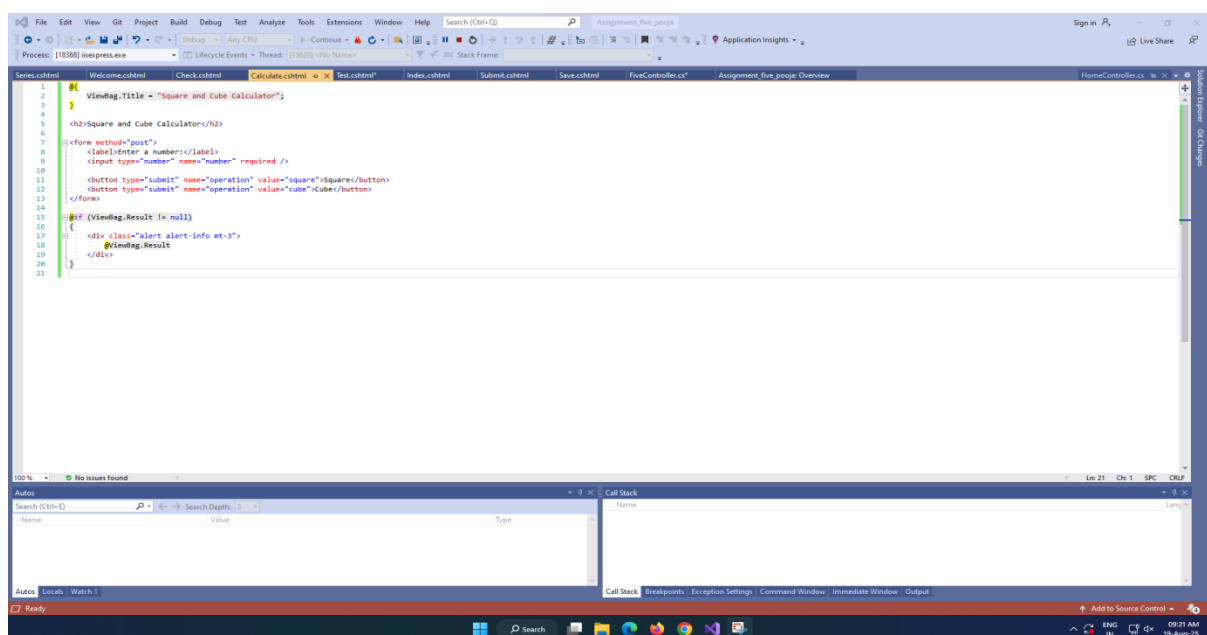
Assignment-5

Q-1

```
[HttpGet]
0 references
public ActionResult Calculate()
{
    return View();
}

[HttpPost]
0 references
public ActionResult Calculate(int number, string operation)
{
    if (operation == "square")
    {
        ViewBag.Result = $"Square of {number} is {number * number}";
    }
    else if (operation == "cube")
    {
        ViewBag.Result = $"Cube of {number} is {number * number * number}";
    }
    else
    {
        ViewBag.Result = "Invalid operation.";
    }

    return View();
}
```

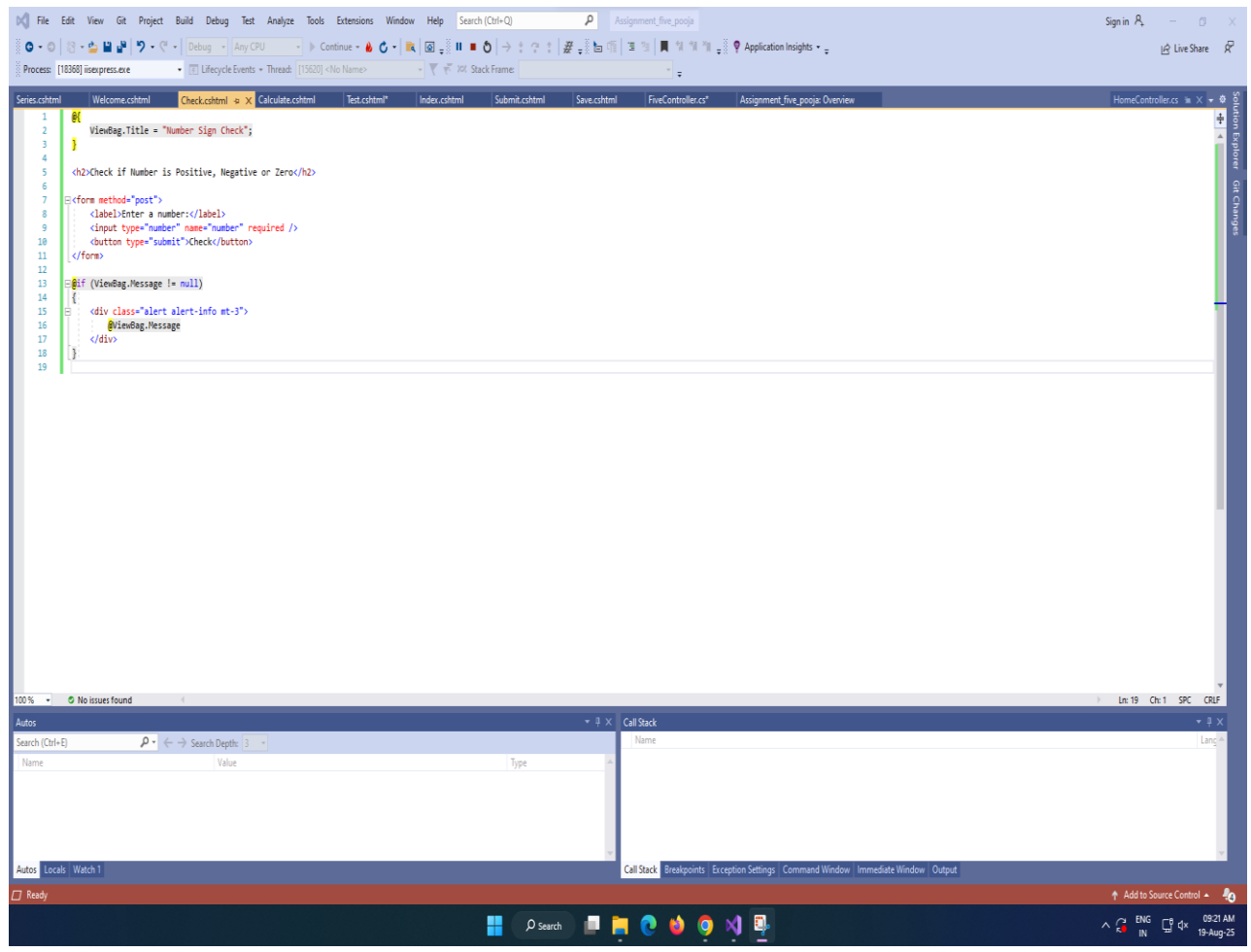


Square and Cube Calculator

Enter a number:

Square of 4 is 16

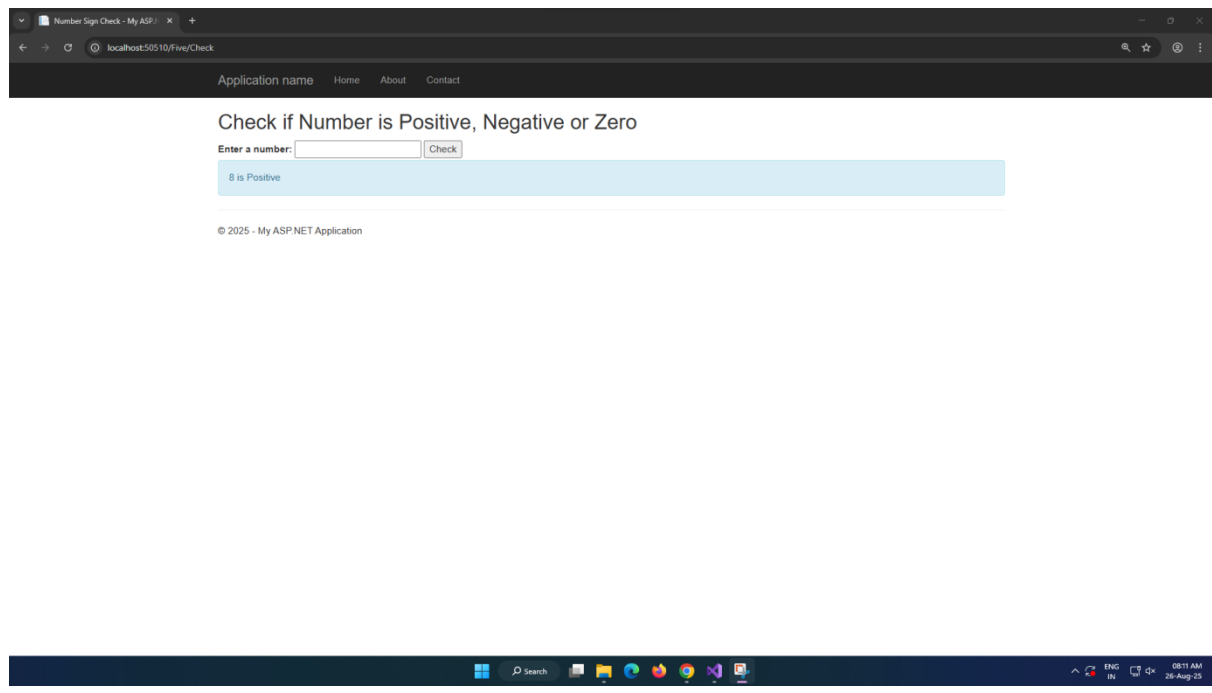
Q-2



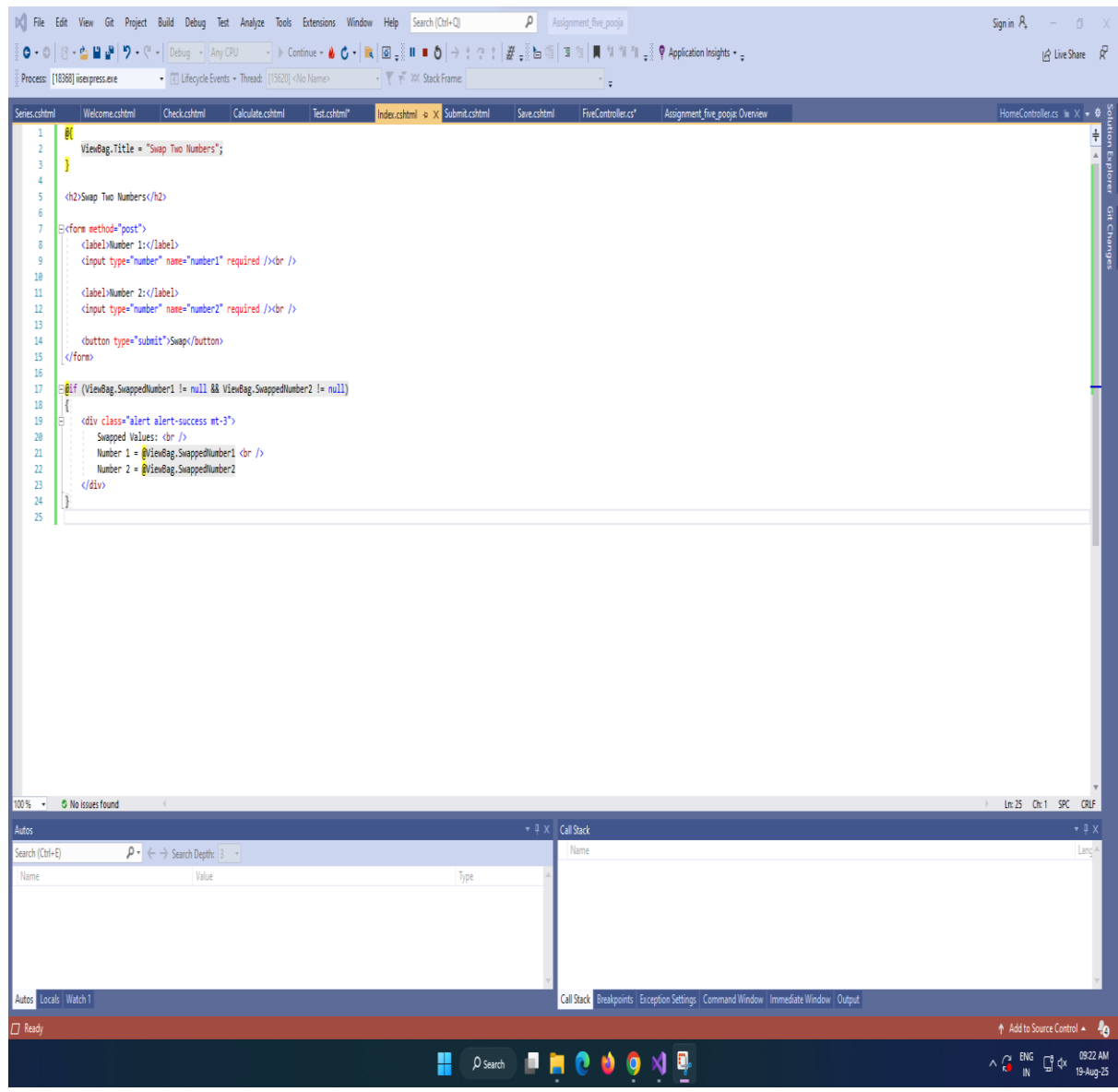
```
[HttpGet]
0 references
public ActionResult Check()
{
    return View();
}

[HttpPost]
0 references
public ActionResult Check(int number)
{
    if (number > 0)
        ViewBag.Message = $"{number} is Positive";
    else if (number < 0)
        ViewBag.Message = $"{number} is Negative";
    else
        ViewBag.Message = "The number is Zero";

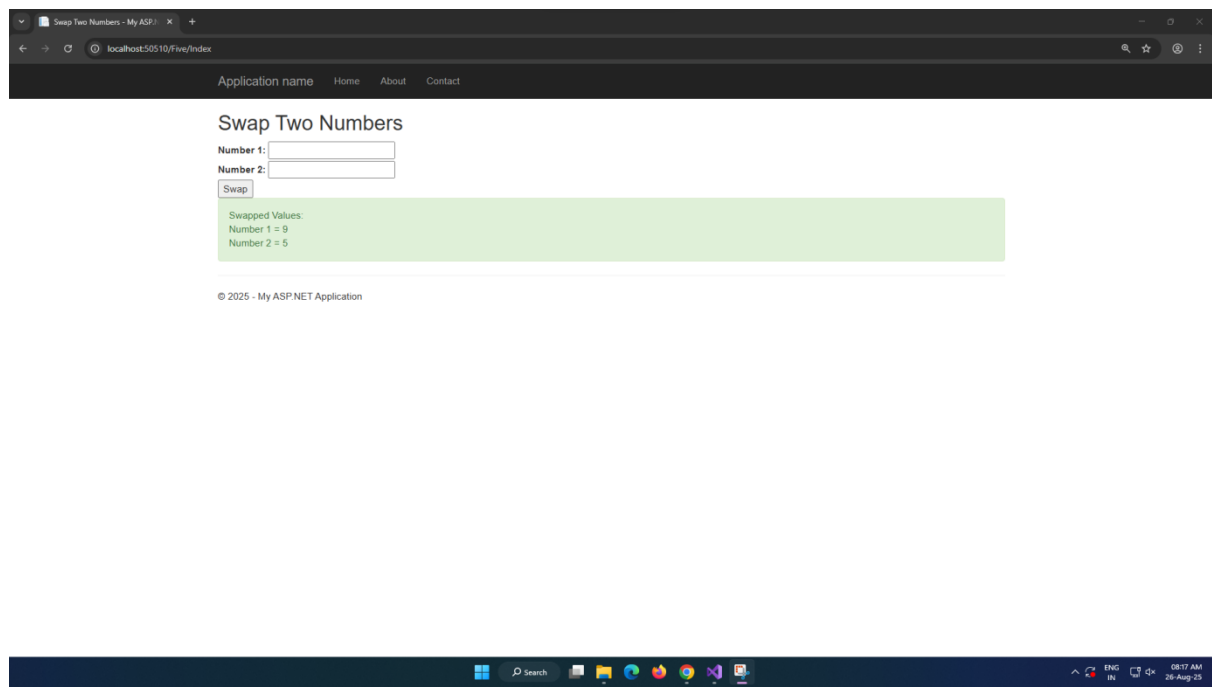
    return View();
}
```



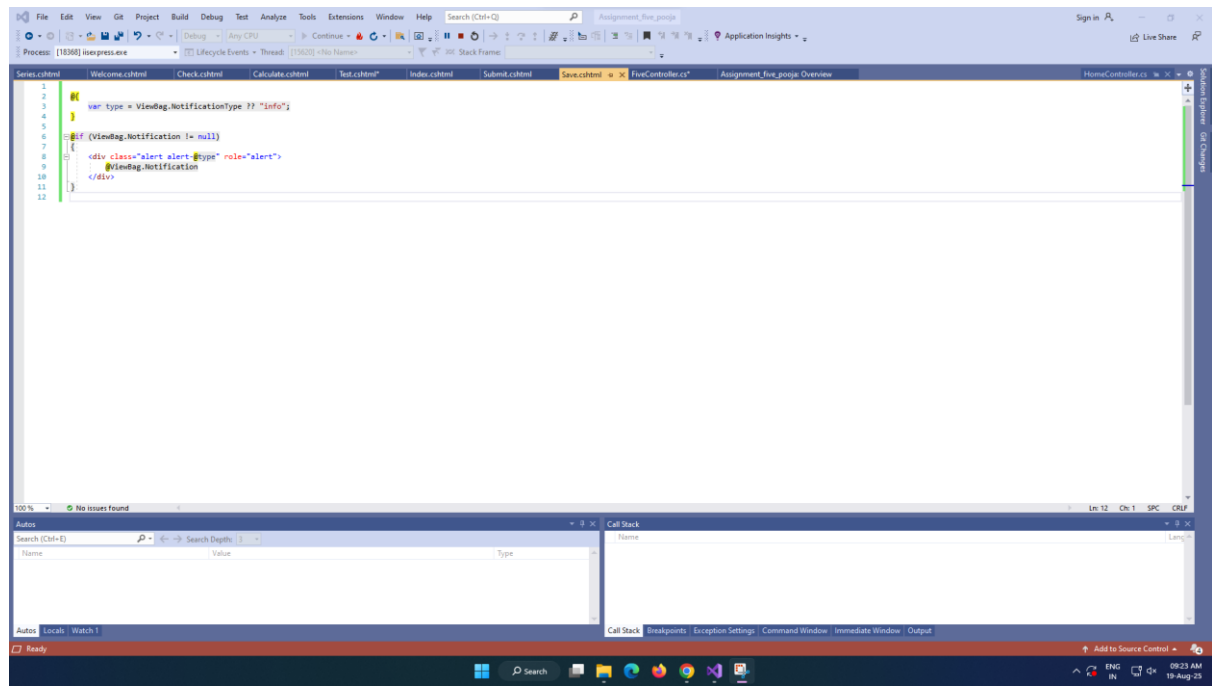
Q-3



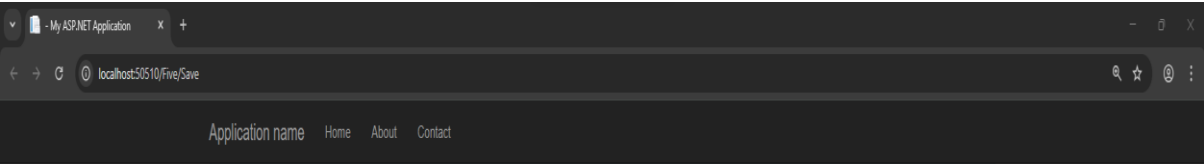
```
Assignment_five_pooja
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6
7  namespace Assignment_five_pooja.Controllers
8  {
9      0 references
10     public class FiveController : Controller
11     {
12         // GET: Five
13
14         [HttpGet]
15         0 references
16         public ActionResult Index()
17         {
18             return View();
19         }
20
21         [HttpPost]
22         0 references
23         public ActionResult Index(int number1, int number2)
24         {
25             int temp = number1;
26             number1 = number2;
27             number2 = temp;
28
29             ViewBag.SwappedNumber1 = number1;
30             ViewBag.SwappedNumber2 = number2;
31
32             return View();
33         }
34
35
36
37
```



Q-4



```
References
public ActionResult Save()
{
    ViewBag.Notification = "Data saved successfully!";
    return View();
}
```



Q-5

```
[HttpGet]
0 references
public ActionResult Series()
{
    return View();
}

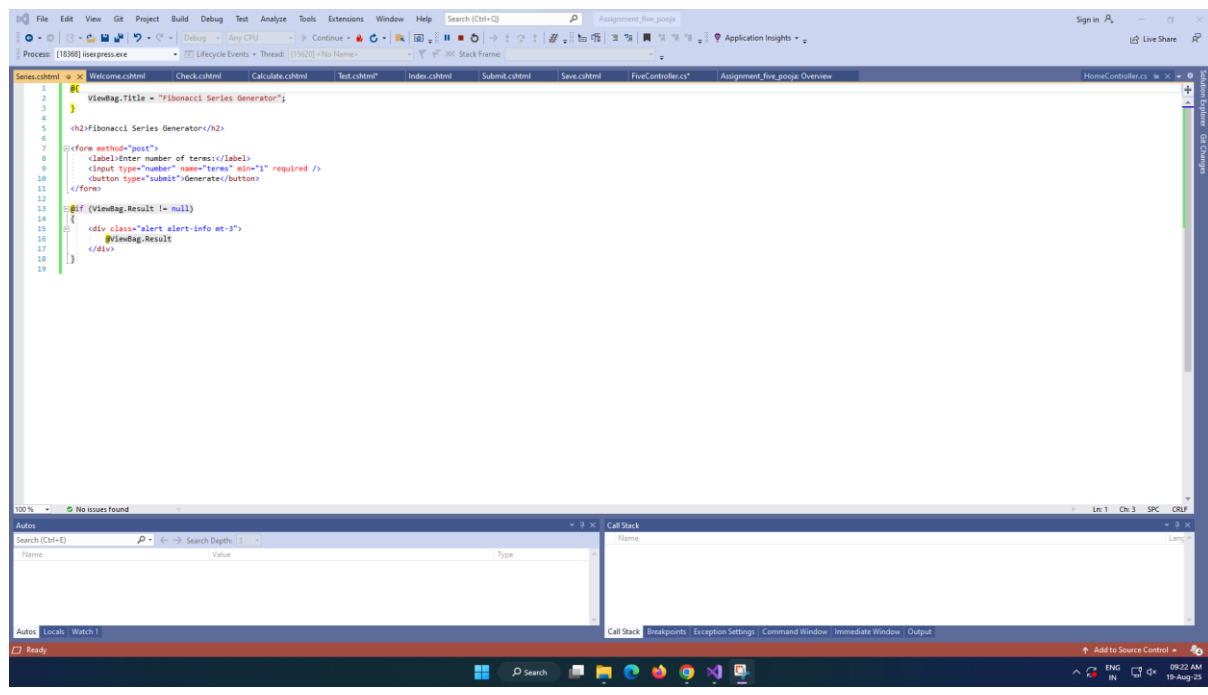
[HttpPost]
0 references
public ActionResult Series(int terms)
{
    var series = new List<int>();

    if (terms > 0)
    {
        int a = 0, b = 1;

        for (int i = 1; i <= terms; i++)
        {
            series.Add(a);
            int next = a + b;
            a = b;
            b = next;
        }

        ViewBag.Result = string.Join(", ", series);
    }
    else
    {
        ViewBag.Result = "Please enter a positive number.";
    }

    return View();
}
}
```



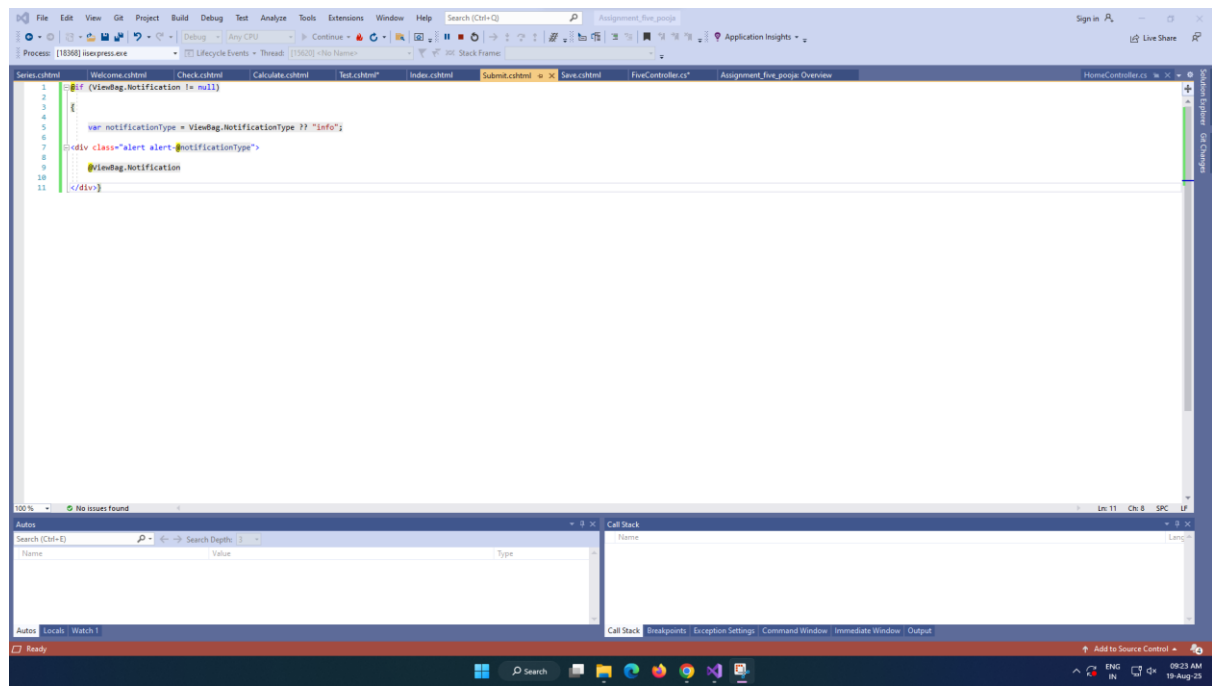
Fibonacci Series Generator

Enter number of terms:

Generate

0, 1, 1

Q-6

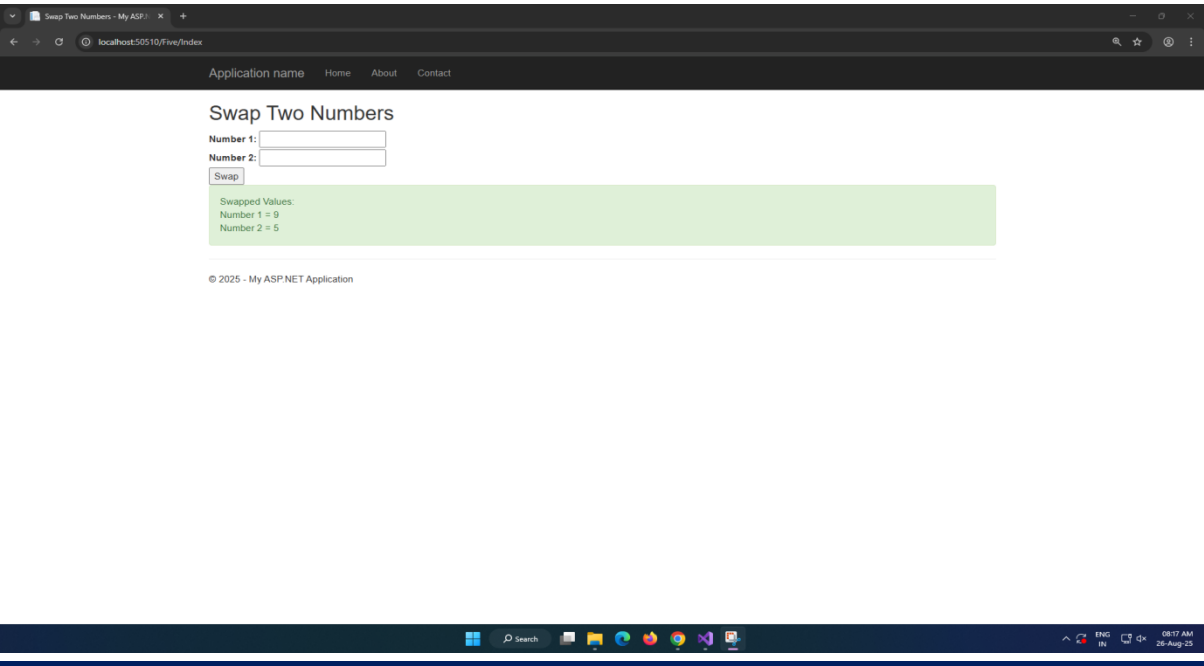


References

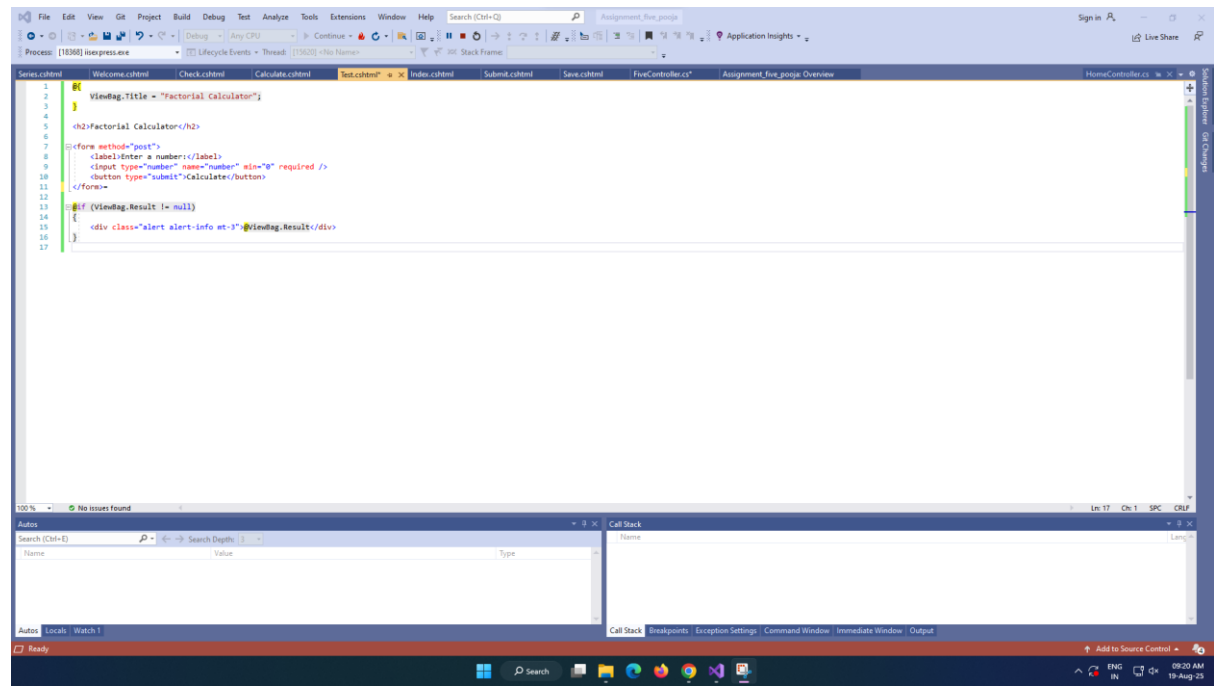
```
public ActionResult Submit()
{
    ViewBag.Notification = "Your form has been submitted successfully!";

    ViewBag.NotificationType = "success";

    return View();
}
```



Q-7

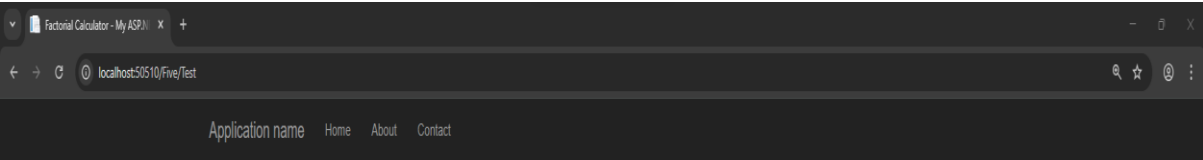


```
[HttpGet]
0 references
public ActionResult Test()
{
    return View();
}

[HttpPost]
0 references
public ActionResult Test(int number)
{
    long factorial = 1;
    for (int i = 1; i <= number; i++)
    {
        factorial *= i;
    }

    ViewBag.Result = $"Factorial of {number} is {factorial}";

    return View();
}
```



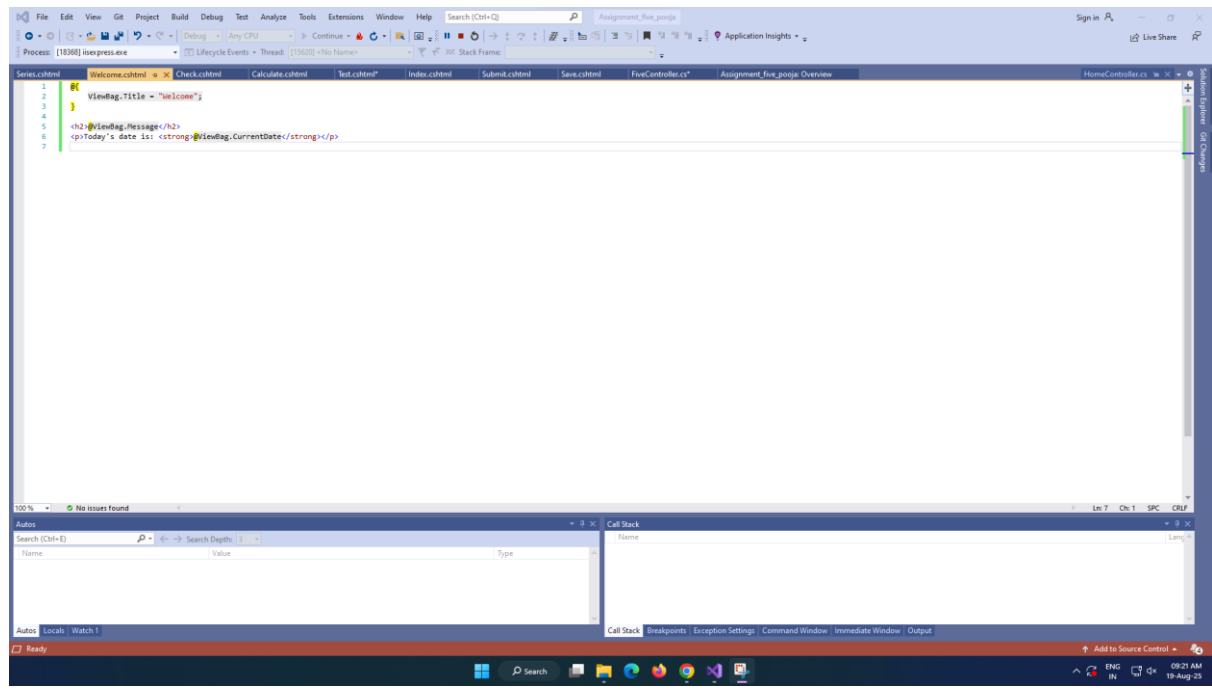
Factorial Calculator

Enter a number:

Factorial of 5 is 120



Q-8



References

```
public ActionResult Welcome()
{
    ViewBag.Message = "Welcome to the site!";
    ViewBag.CurrentDate = DateTime.Now.ToString("dddd, MMMM dd, yyyy");

    return View();
}
```