

Първи стъпки в кодирането

Да напишем първата си програма със C# и Visual Studio

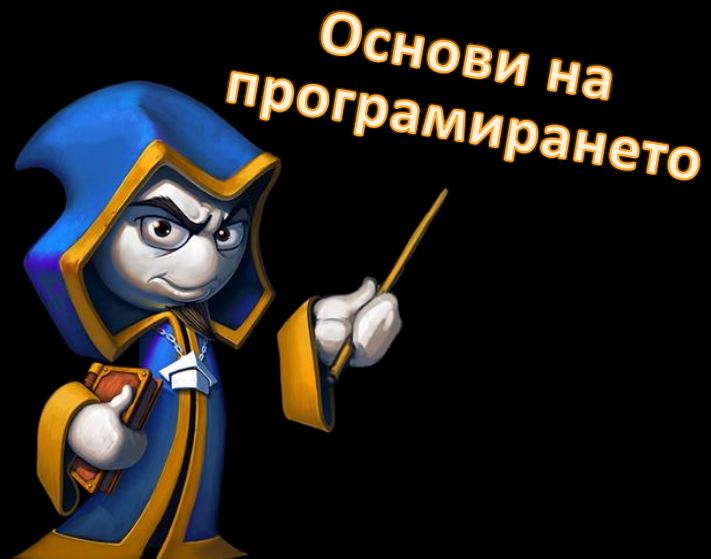


СофтУни

трейнерски екип

Софтуерен университет

<http://softuni.bg>



Съдържание

1. Какво означава да програмираме?
2. Първа програмка със C# и Visual Studio
3. Да направим конзолна програма
 - Създаване на конзолна C# програма
 - Стартиране на програмата
 - Тестване в judge системата
4. Да напишем графична програма
5. Да напишем уеб приложение





**Какво означава
"да програмираме"?**

Какво означава "програмиране"?

- Да "програмираме" означава да даваме команди на компютъра какво да прави
 - Командите се подреждат една след друга
 - Така те образуват "компютърна програма"
 - Компютърната програма е поредица от команди
 - Програмите се пишат на език за програмиране
 - Например C#, Java, JavaScript, Python, PHP, C, C++, ...
 - Използва се среда за програмиране (например Visual Studio)

Компютърна програма – примери

- Програма, която свири музикалната нота "ла" (за 0.5 секунди)

```
Console.Beep(432, 500);
```

- Програма, която свири поредица от музикални ноти:

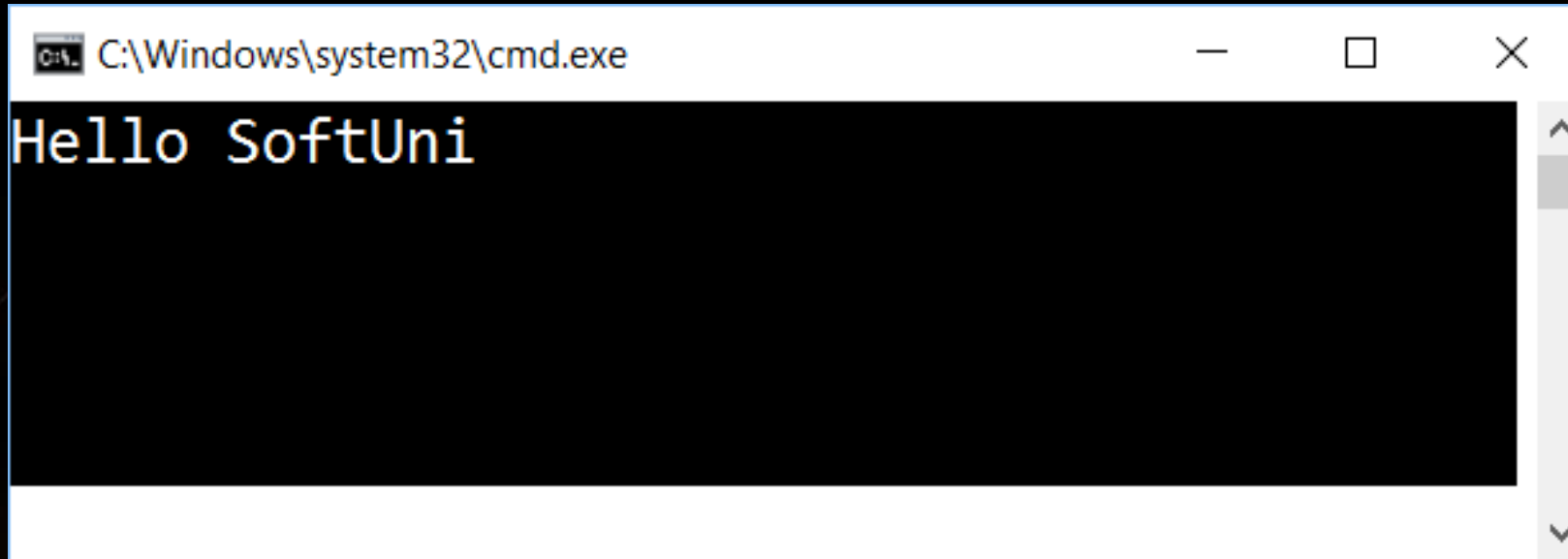
```
for (int i = 200; i <= 4000; i += 200)  
    Console.Beep(i, 100);
```

- Програма, която конвертира от левове в евро:

```
var leva = int.Parse(Console.ReadLine());  
var euro = leva / 1.95583;  
Console.WriteLine(euro);
```


Компютърни програми

- Програма == последователност от команди
 - Съдържа команди, пресмятания, проверки, повторения, ...
- Програмите се пишат в текстов формат
 - Текстът на програмата се нарича **сорс код**
- Сорс кодът се компилира до **изпълним файл**
 - Например **Program.cs** се компилира до **Program.exe**
 - Или се изпълнява директно
 - Например JavaScript сорс кодът се изпълнява от уеб браузъра

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt area is black with white text that reads 'Hello SoftUni'. There is a vertical scrollbar on the right side of the window.

```
C:\Windows\system32\cmd.exe  
Hello SoftUni
```

Да направим конзолна програмка

Среда за разработка

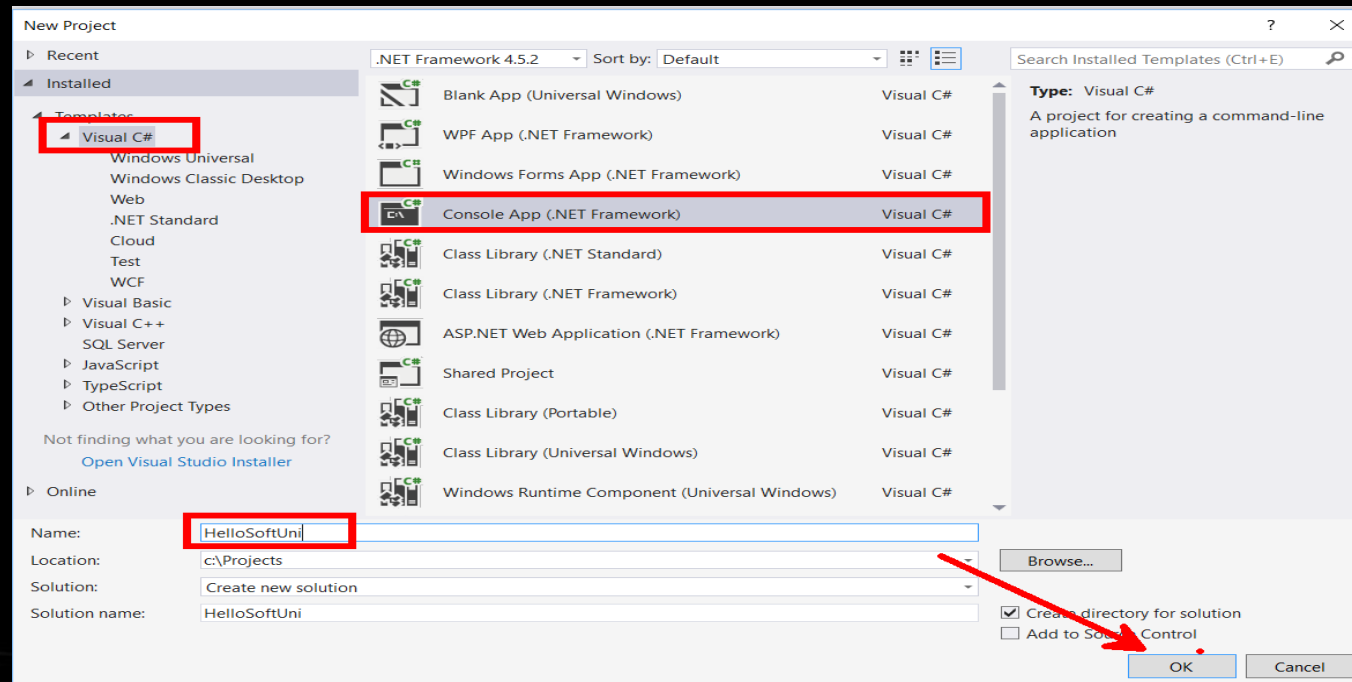
- За да програмирате, ви трябва среда за разработка
 - Integrated Development Environment (**IDE**)
 - За C# →
 - Visual Studio за Windows
 - MonoDevelop за Linux / Mac OS X
 - за Java → IntelliJ IDEA
 - за PHP → PHP Storm

Среда за разработка

- Инсталирайте си Microsoft **Visual Studio Community 2017**
 - <https://visualstudio.com/products/visual-studio-community-vs>
 - Може и по-стара версия на Visual Studio, но не се препоръчва
- Алтернативна среда за разработка (online)
 - C# – .NET Fiddle - <https://dotnetfiddle.net/>
 - Java – <https://www.compilejava.net/>
 - JavaScript – може директно в конзолата на браузър

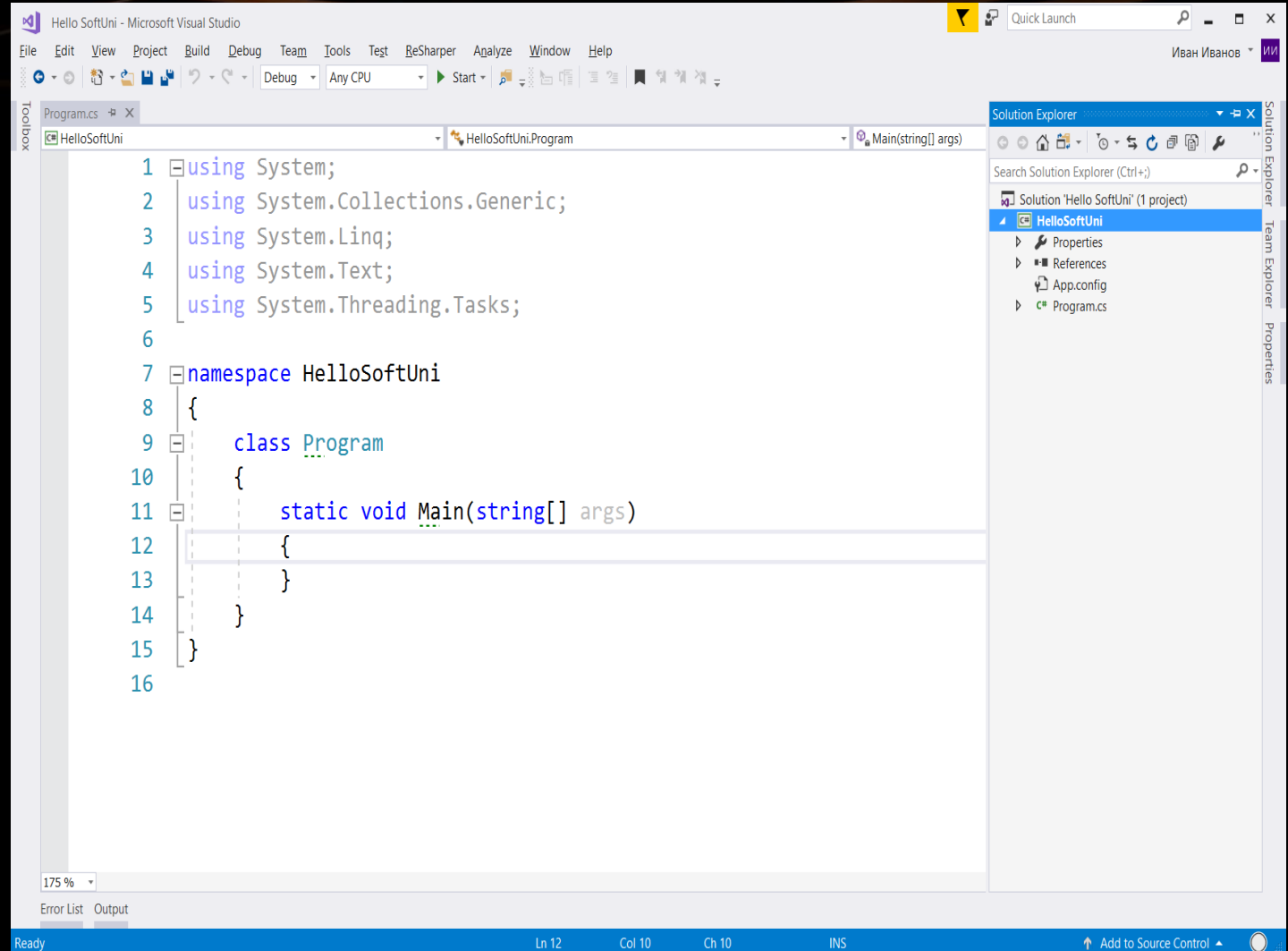
Създаване на конзолна програма

- Стартирайте Visual Studio
- Нов конзолен проект – [File] → [New] → [Project] → [Visual C#] → [Console Application]



Писане на програмен код

- Сорс кодът на програма се пише в секцията **Main(string[] args)**
 - Между отварящата и затварящата скоба { }
- Натиснете [Enter] след отварящата скоба {
- Кодът на програмата се пише отместен навътре



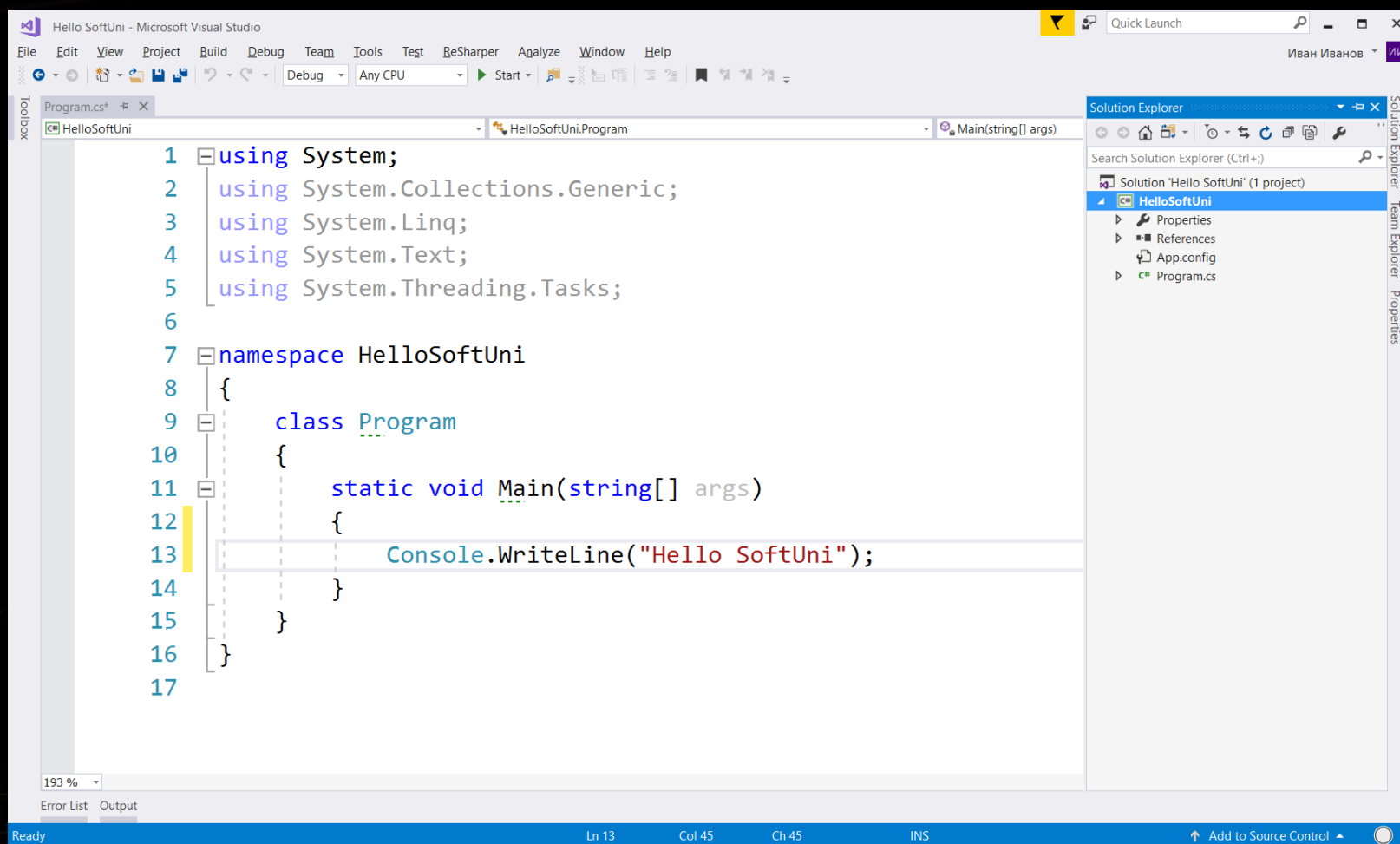
The screenshot shows the Microsoft Visual Studio IDE with a project named 'HelloSoftUni'. The main code file, 'Program.cs', is open and displays the following C# code:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloSoftUni
8 {
9     class Program
10    {
11        static void Main(string[] args)
12        {
13        }
14    }
15 }
16
```

The Solution Explorer on the right shows the project structure with 'HelloSoftUni' as the main project, containing 'Properties', 'References', 'App.config', and 'Program.cs'. The status bar at the bottom indicates 'Ready' and 'Ln 12 Col 10 Ch 10 INS'.

Писане на програмен код (2)

- Напишете следния код: `Console.WriteLine("Hello SoftUni");`

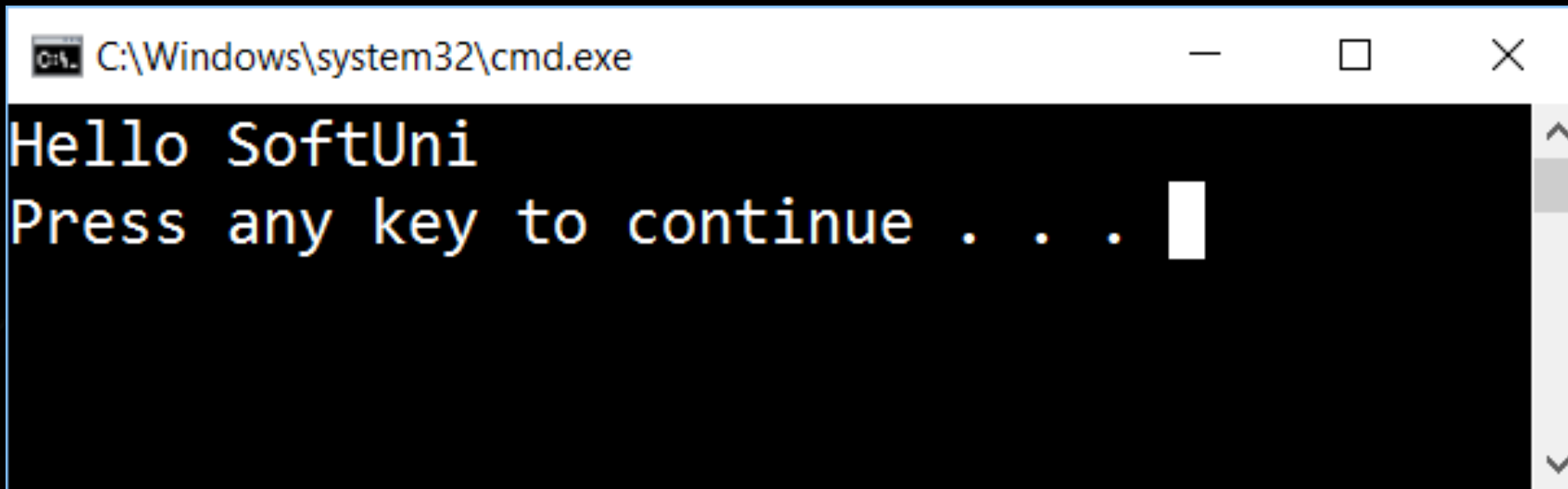


The screenshot shows the Microsoft Visual Studio IDE with a project named 'Hello SoftUni'. The main editor window displays the code for 'Program.cs'. The code includes several using statements for System, System.Collections.Generic, System.Linq, System.Text, and System.Threading.Tasks. It defines a namespace 'HelloSoftUni' and a class 'Program'. Inside the 'Program' class, there is a static method 'Main' that takes a string array 'args' and calls 'Console.WriteLine' to output 'Hello SoftUni'. The Solution Explorer on the right shows the project structure with 'Program.cs' selected. The status bar at the bottom indicates the current line is 13, column 45.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloSoftUni
8 {
9     class Program
10    {
11        static void Main(string[] args)
12        {
13            Console.WriteLine("Hello SoftUni");
14        }
15    }
16 }
17
```

Стартиране на програмата

- За стартиране на програмата натиснете [Ctrl + F5]
- Ако няма грешки, програмата ще се изпълни
- Резултатът ще се изпише на конзолата (в черния прозорец):



```
C:\Windows\system32\cmd.exe  
Hello SoftUni  
Press any key to continue . . .
```

Тестване на програмата в Judge

- Тествайте кода си в онлайн judge системата:
 - <https://judge.softuni.bg/Contests/150/First-Steps-in-Coding>

01. Hello SoftUni

Participants Tests Change Delete

Administration

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloSoftUni
8 {
9     class Program
10     {
11         static void Main(string[] args)
12         {
13             Console.WriteLine("Hello SoftUni");
14         }
15     }
16 }
17
```

Allowed working time: 0.100 sec.
Allowed memory: 16.00 MB
Size limit: 16.00 KB
Checker: Accept Everything

C# code Submit

Submissions

1 2

Points	Time and memory used	Submission date
✓ 100 / 100	Memory: 1.09 MB Time: 0.001 s	16:48:58 04.03.2017

Details

Типични грешки в C# програмите

- Писане извън тялото на **Main()** метода:

```
Console.WriteLine("Hello C#");
```

- Бъркане на малки и главни букви:

```
Console.writeLine("Hello C#");
```

```
console.WriteLine("Hello C#");
```

- Липса на ; в края на всяка команда

```
Console.WriteLine("Hello C#")
```

- Липсваща кавичка " или липсваща скоба (или)

```
Console.WriteLine("Hello C#);
```

```
Console.WriteLine("Hello C#" ;
```

```
Hello SoftUni - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test ReSharper Analyze Window Help
Program.cs* x
HelloSoftUni
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloSoftUni
8 {
9     class Program
10     {
11         static void Main(string[] args)
12         {
13             Console.WriteLine("Hello SoftUni");
14         }
15     }
16 }
17
```

```
C:\Windows\system32\cmd.exe
Hello SoftUni
Press any key to continue . . .
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloSoftUni
8 {
9     class Program
10     {
11         static void Main(string[] args)
12         {
13             Console.WriteLine("Hello SoftUni");
14         }
15     }
16 }
17
```

Allowed working time: 0.100 sec.
Allowed memory: 16.00 MB
Size limit: 16.00 KB
Checker: Accept Everything

C# code Submit

Submissions

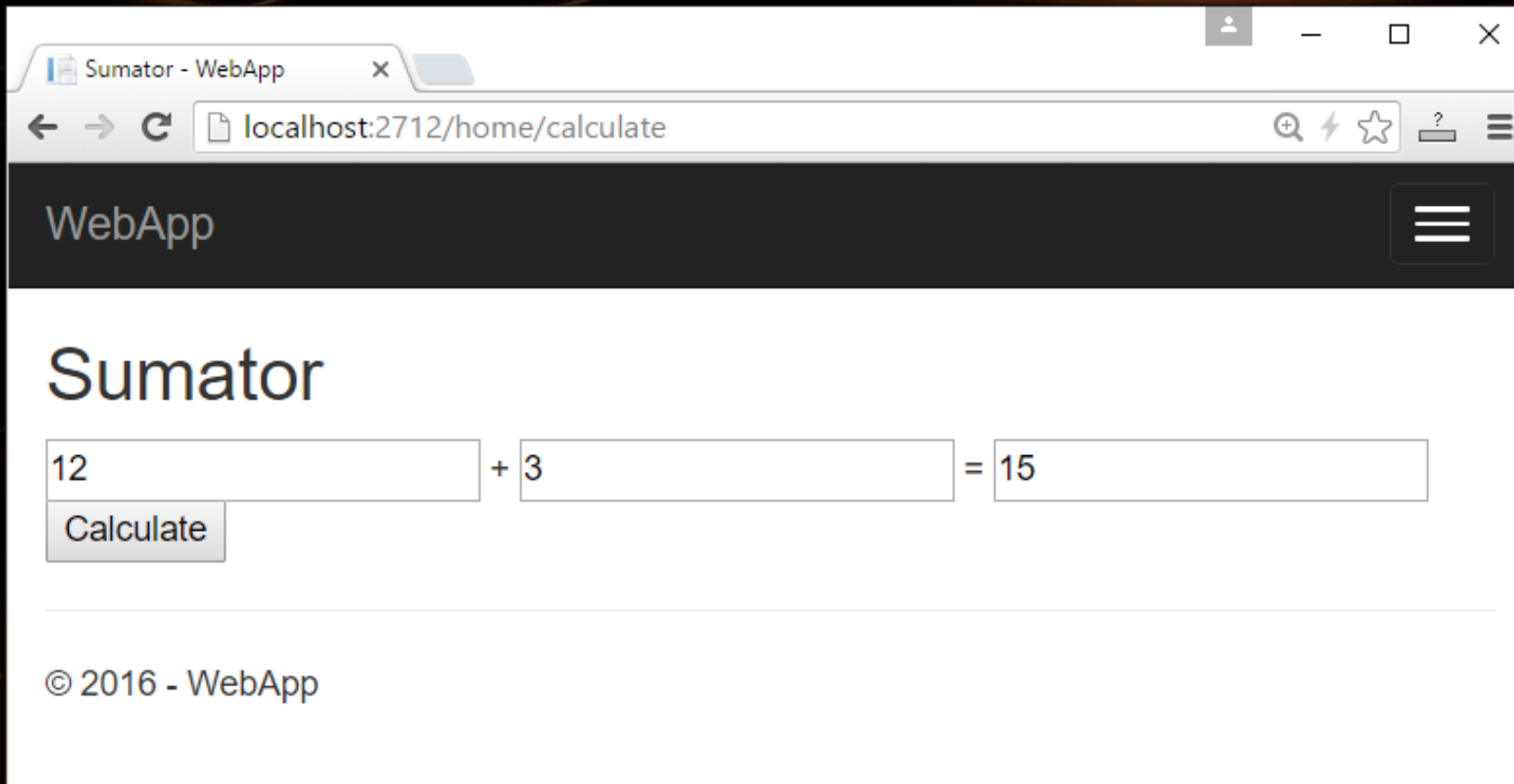
1 2

Points	Time and memory used	Submission date
✓ 100 / 100	Memory: 1.09 MB Time: 0.001 s	16:48:58 04.03.2017

Details

Конзолни програмки със C#

Упражнения в клас (лаб)



The screenshot shows a web browser window with the title "Sumator - WebApp". The address bar displays "localhost:2712/home/calculate". The page has a dark header with the text "WebApp" and a hamburger menu icon. The main content area is titled "Sumator" and contains a calculation form. The form consists of two input fields: the first contains "12" and the second contains "3". Between them is a "+" sign, and after the second field is an "=" sign followed by a third input field containing "15". Below the first input field is a "Calculate" button. At the bottom of the page, there is a copyright notice: "© 2016 - WebApp".

Sumator - WebApp

localhost:2712/home/calculate

WebApp

Sumator

12 + 3 = 15

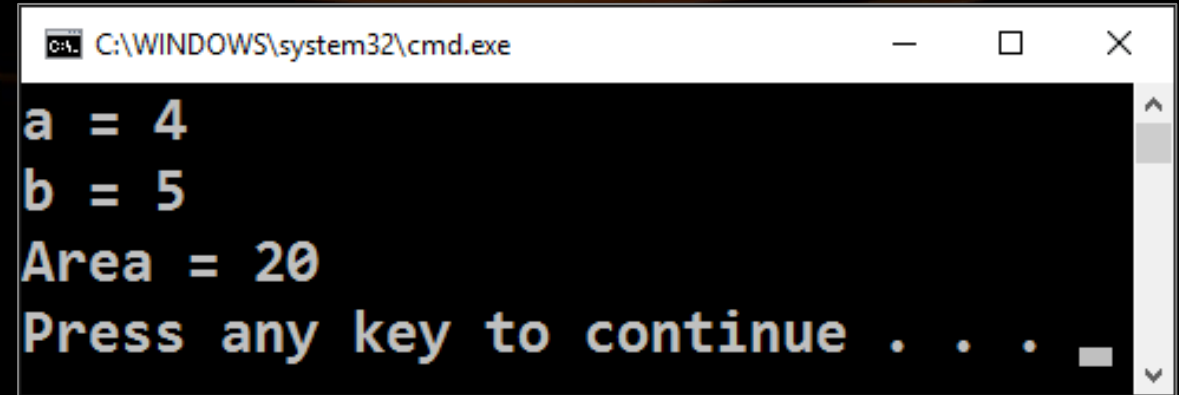
Calculate

© 2016 - WebApp

Да направим уеб приложение

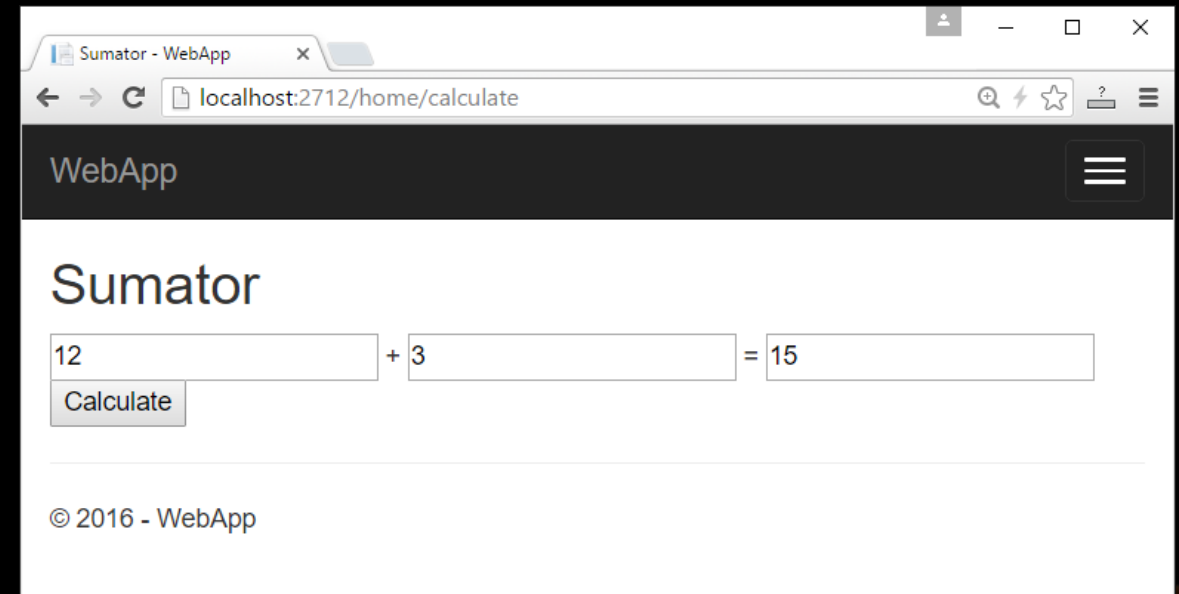
Конзолни и уеб приложения

- Конзолните приложения четат входните си данни и отпечатват изхода си на текстова **конзола**
- Уеб приложенията ползват уеб-базиран потребителски интерфейс
 - Работят през **уеб** **браузър** и **уеб сървър**



```
C:\WINDOWS\system32\cmd.exe

a = 4
b = 5
Area = 20
Press any key to continue . . .
```



Създаване на уеб приложение

New Project

Recent .NET Framework 4.6 Sort by: Default Search Installed Templates

Installed

Templates

- Visual C#
 - Windows
 - Web**
 - Android
 - Cloud
 - Extensibility
 - iOS
 - LightSwitch
 - Mobile Apps

ASP.NET Web Application Visual C#

Class Library (Package) Visual C#

Console Application (Package) Visual C#

Type: Visual C#

A project template for creating ASP.NET applications. You can create ASP.NET Web Forms, MVC, or Web API applications and add many other features in ASP.NET.

Application Insights

☐ Add Application Insights to Project

Help you understand and optimize your application.

[Learn more](#)

[Privacy statement](#)

[Click here to go online and find templates.](#)

Name: WebApp

Location: C:\Projects

Solution name: WebApp

☒ Create directory for solution

☐ Add to source control

Създаване на уеб приложение (2)

New ASP.NET Project - WebApp

Select a template:

ASP.NET 4.6 Templates

Empty Web Forms **MVC** Web API Single Page Application

Azure API App (Preview) Azure Mobile App (Preview) Azure Mobile Service

ASP.NET 5 Preview Templates

Empty Web API Web Application

Add folders and core references for:

☐ Web Forms ☒ MVC ☐ Web API

☐ Add unit tests

Test project name: WebApp.Tests

A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

[Learn more](#)

Change Authentication

Authentication: Individual User Accounts

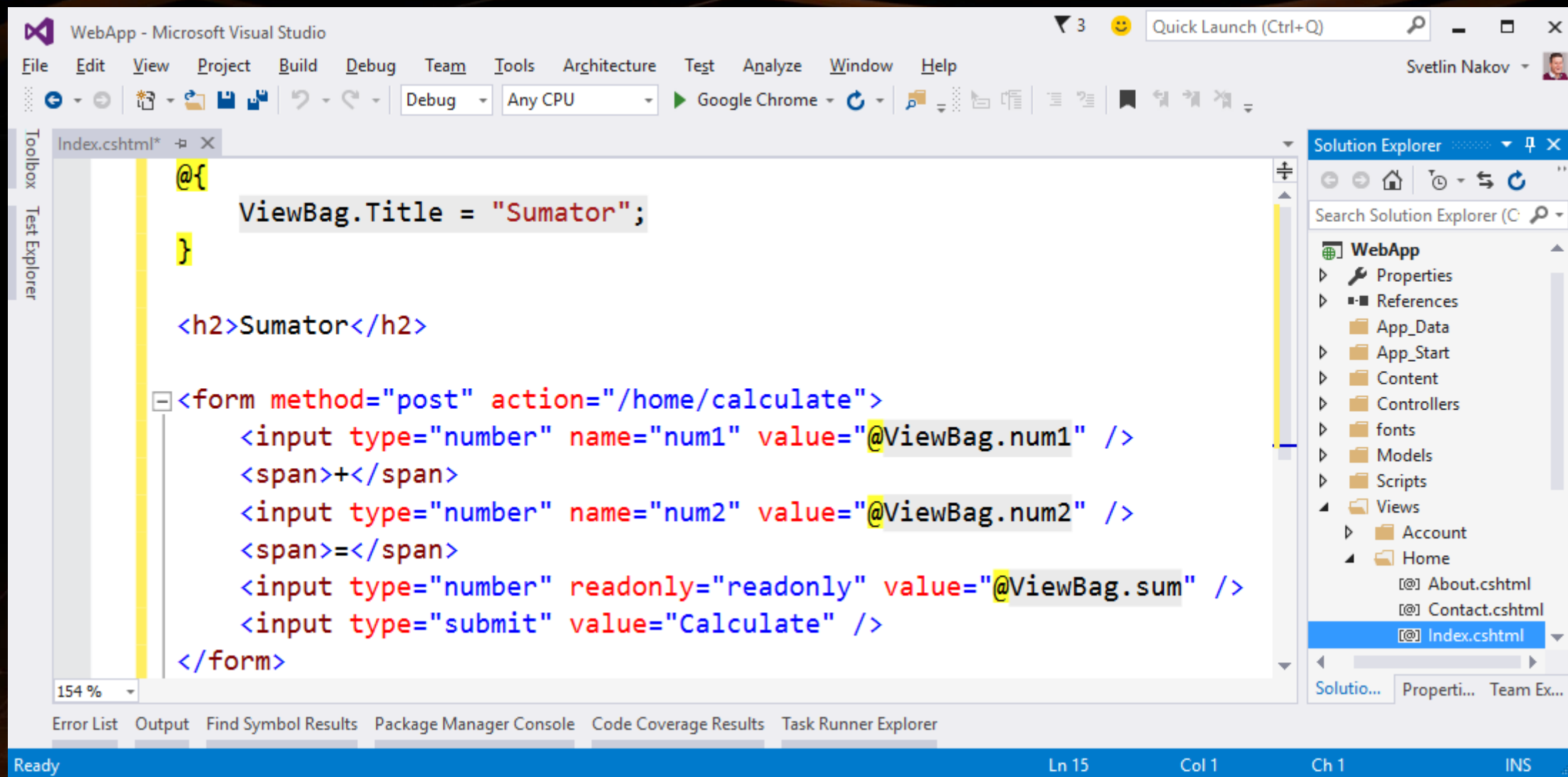
Microsoft Azure

☐ Host in the cloud

Web App

OK Cancel

Създаване на изглед: Index.cshtml

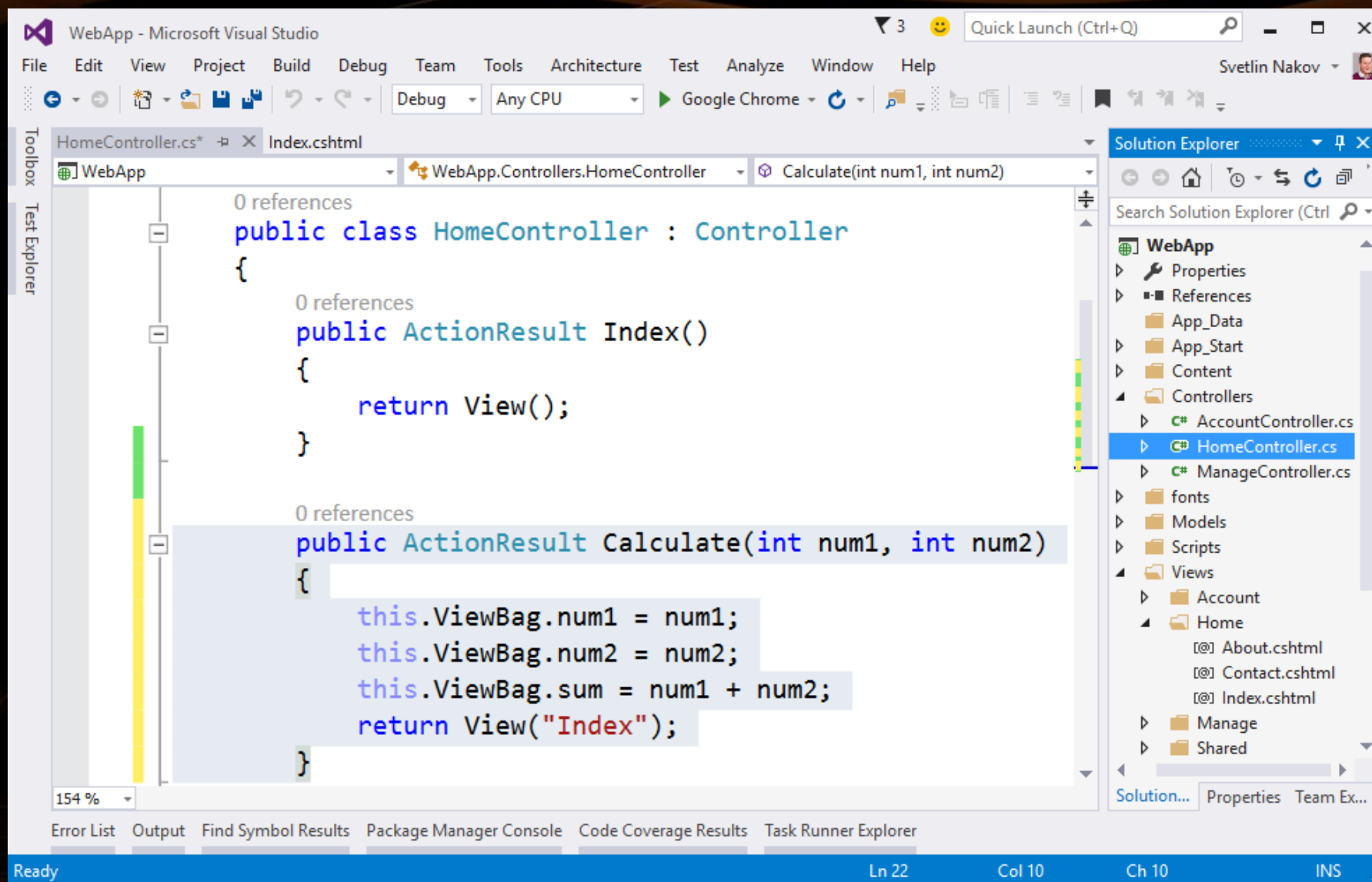


The screenshot displays the Microsoft Visual Studio IDE with a web application project named 'WebApp'. The main editor window shows the 'Index.cshtml' file, which contains the following code:

```
@{  
    ViewBag.Title = "Sumator";  
}  
  
<h2>Sumator</h2>  
  
<form method="post" action="/home/calculate">  
    <input type="number" name="num1" value="@ViewBag.num1" />  
    <span>+</span>  
    <input type="number" name="num2" value="@ViewBag.num2" />  
    <span>=</span>  
    <input type="number" readonly="readonly" value="@ViewBag.sum" />  
    <input type="submit" value="Calculate" />  
</form>
```

The Solution Explorer on the right side of the IDE shows the project structure, with the 'Views' folder expanded and 'Index.cshtml' selected. The status bar at the bottom indicates the current position is Line 15, Column 1, Character 1.

Създаване на действие: HomeController.cs



The screenshot displays the Microsoft Visual Studio IDE with a web application project named 'WebApp'. The 'HomeController.cs' file is open, showing the following code:

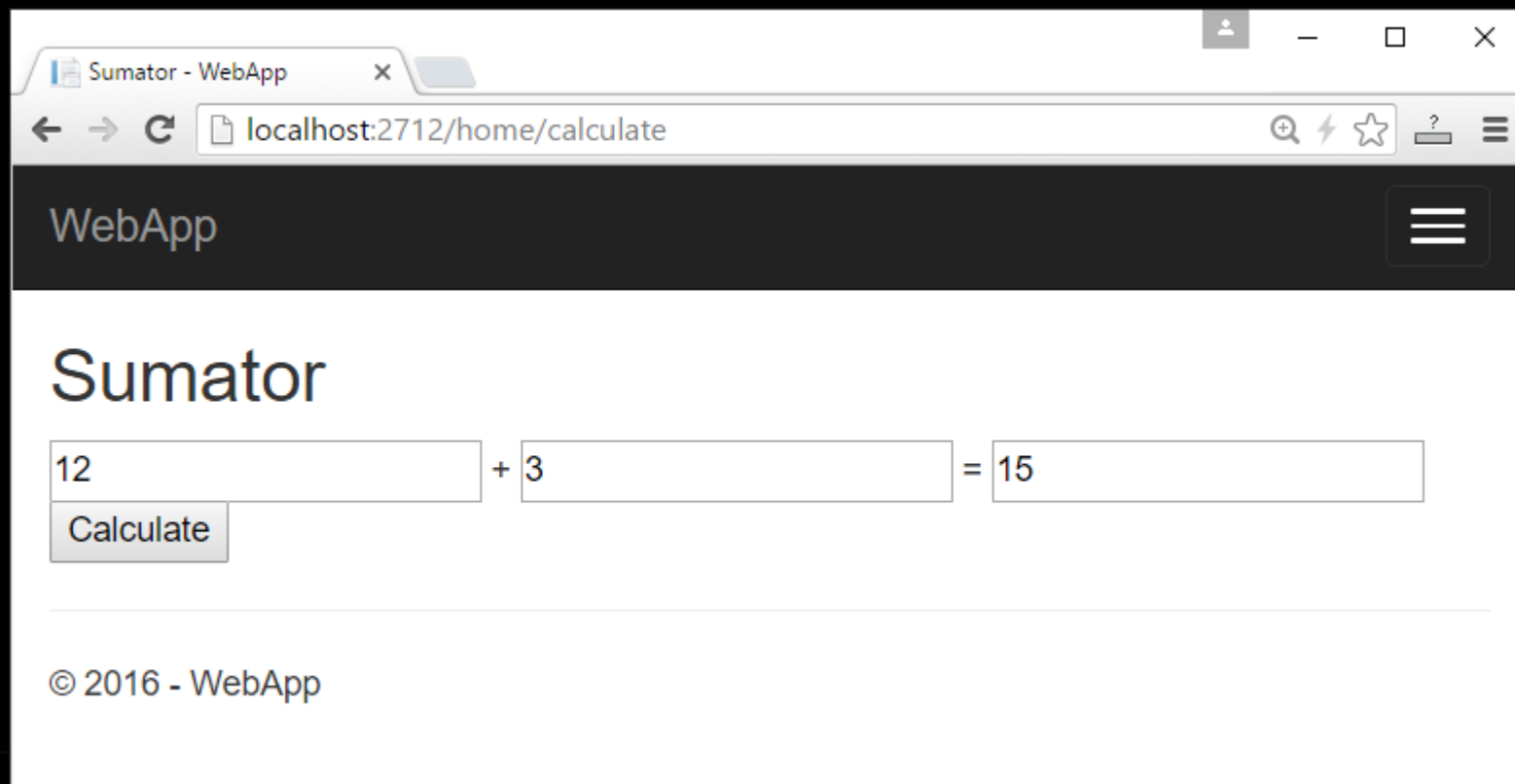
```
0 references
public class HomeController : Controller
{
    0 references
    public ActionResult Index()
    {
        return View();
    }

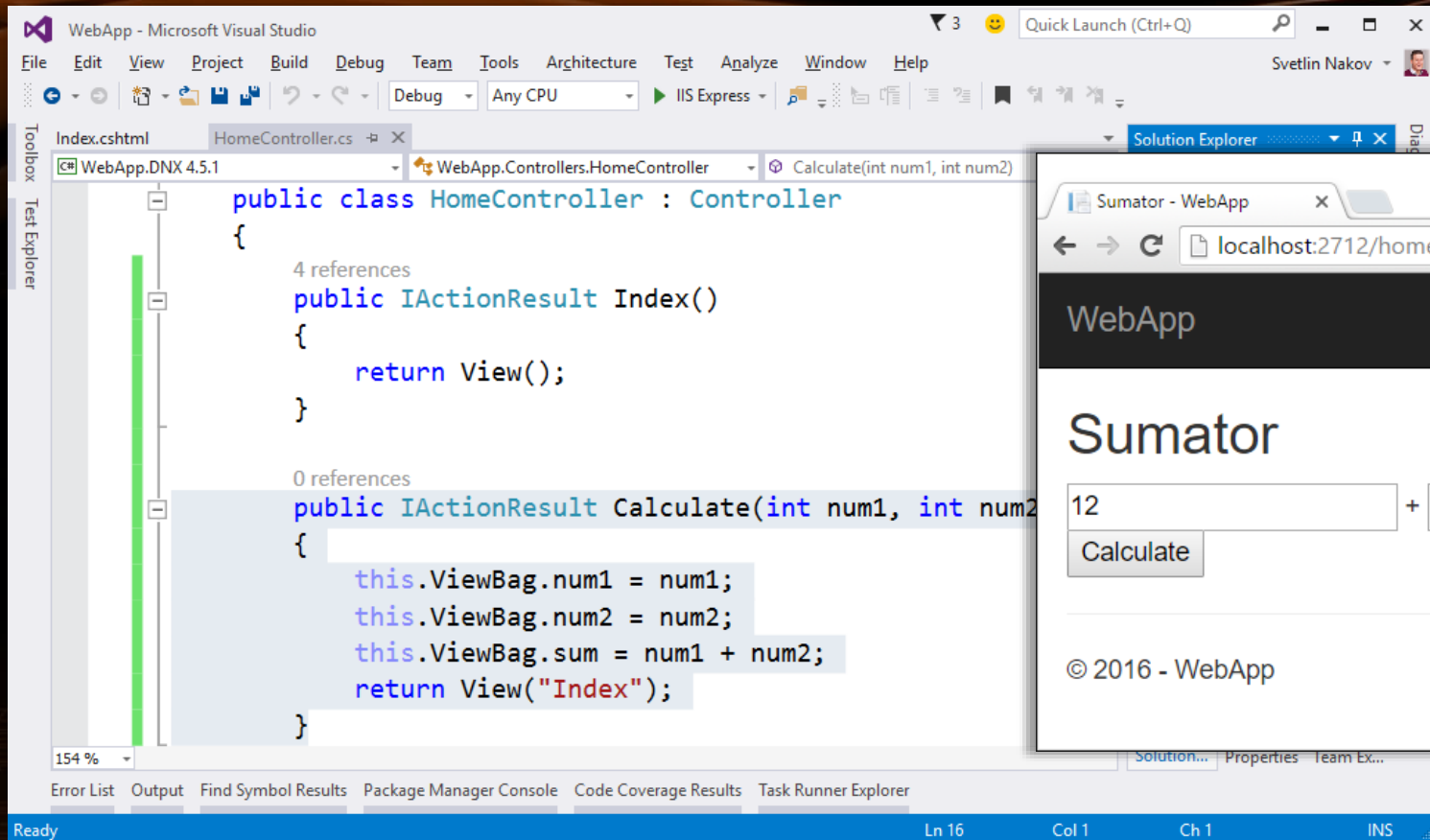
    0 references
    public ActionResult Calculate(int num1, int num2)
    {
        this.ViewBag.num1 = num1;
        this.ViewBag.num2 = num2;
        this.ViewBag.sum = num1 + num2;
        return View("Index");
    }
}
```

The 'Solution Explorer' on the right shows the project structure, with 'HomeController.cs' selected under the 'Controllers' folder. The status bar at the bottom indicates 'Ready' and 'Ln 22 Col 10 Ch 10 INS'.

Стартиране на уеб приложението

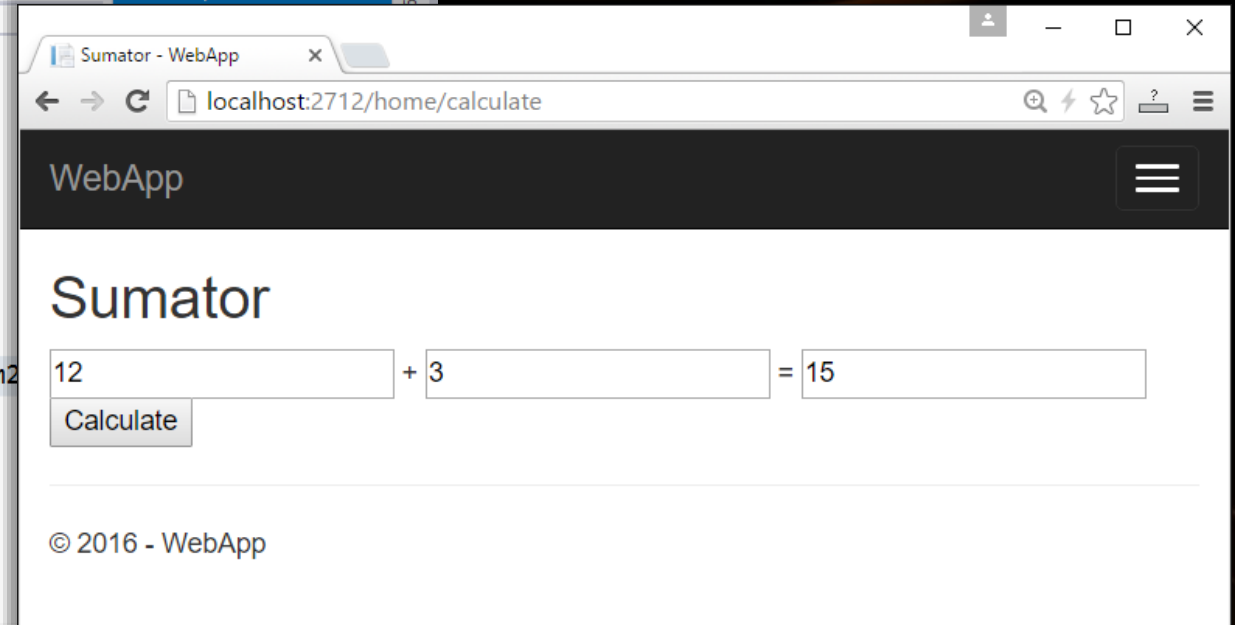
- Стартираме уеб приложението с [Ctrl+F5]
- Тестваме в уеб браузъра с различни числа





```
public class HomeController : Controller
{
    4 references
    public IActionResult Index()
    {
        return View();
    }

    0 references
    public IActionResult Calculate(int num1, int num2)
    {
        this.ViewBag.num1 = num1;
        this.ViewBag.num2 = num2;
        this.ViewBag.sum = num1 + num2;
        return View("Index");
    }
}
```



Изграждане на веб приложение

Упражнение в клас (лаб)

Какво научихме днес?

- Програмиране означава да пишеш команди за компютъра
 - Компютърна програма е **поредица команди**
 - Използва се **език за програмиране** (например C#) + **среда за разработка** (например Visual Studio)
- На C# командите се пишат в частта **Main(...)**

```
static void Main(string[] args)
{
    Console.WriteLine("Hello");
}
```



- Печатаме с **Console.WriteLine(...)**, стартираме с [Ctrl+F5]

Първи стъпки в кодирането



Въпроси?



- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"



- Благодарности: настоящият материал може да съдържа части от следните източници
 - Книга "Основи на програмирането със C#" от Светлин Наков и колектив с лиценз CC-BY-SA

Безплатни обучения в СофтУни



- Фондация "Софтуерен университет" – softuni.org
- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
 - softuni.bg
- СофтУни @ Facebook
 - facebook.com/SoftwareUniversity
- СофтУни форуми – forum.softuni.bg

