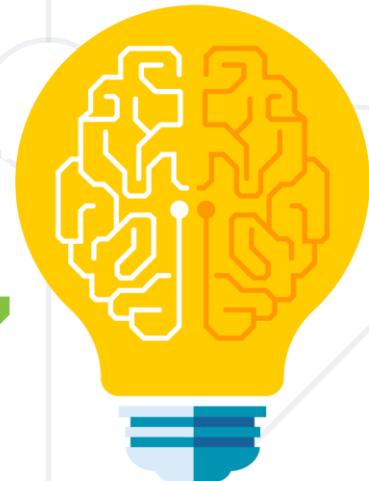


# Първи стъпки в програмирането

Работа с конзола, аритметични операции с числа



Софтуни

Преподавателски екип



SoftUni



Софтуерен университет

<https://softuni.bg>

# Съдържание

1. Какво е програмиране?
2. Първа програма със **C#** и **Visual Studio**
3. Променливи и типове данни
4. Четене на потребителски вход
5. Прости операции
  - работа с текст и числа
  - аритметични операции
6. Печатане на конзолата





**Какво е програмиране?**

# Какво е програмиране?

- Компютърна наука
- Използва **команди**, за да **комуникираме** с компютъра
- Командите се подреждат и изпълняват **една след друга**
- Поредицата от команди образува **компютърна програма**



# Какво е език за програмиране?

- Компютърните програми се пишат на език за програмиране
- Пример: **C#, Java, JavaScript, Python, PHP, C, C++**
- Използва се **среда за програмиране**  
**(среда за разработка)**
- Пример: **Visual Studio, IntelliJ IDEA, PyCharm, Visual Studio Code, Code Blocks**



# Какво е компютърна програма?

- Програмата е **последователност от команди**
- Може да съдържа пресмятания, проверки, повторения
- Програмите се пишат в текстов формат
  - Текстът на програмата се нарича **сурс код**
- Сурс кодът се **компилира** до изпълним файл
  - Например **Program.cs** се компилира до **Program.exe**



# Интересно за C#

- В **топ 5** на най-популярните езици за програмиране
- Около **31%** от всички програмисти го използват редовно
- Третата по големина общност в **StackOverflow** с повече от **1.1 миллиона** теми
- Глобално, **всеки месец** се предлагат повече от **17 000 C# позиции**



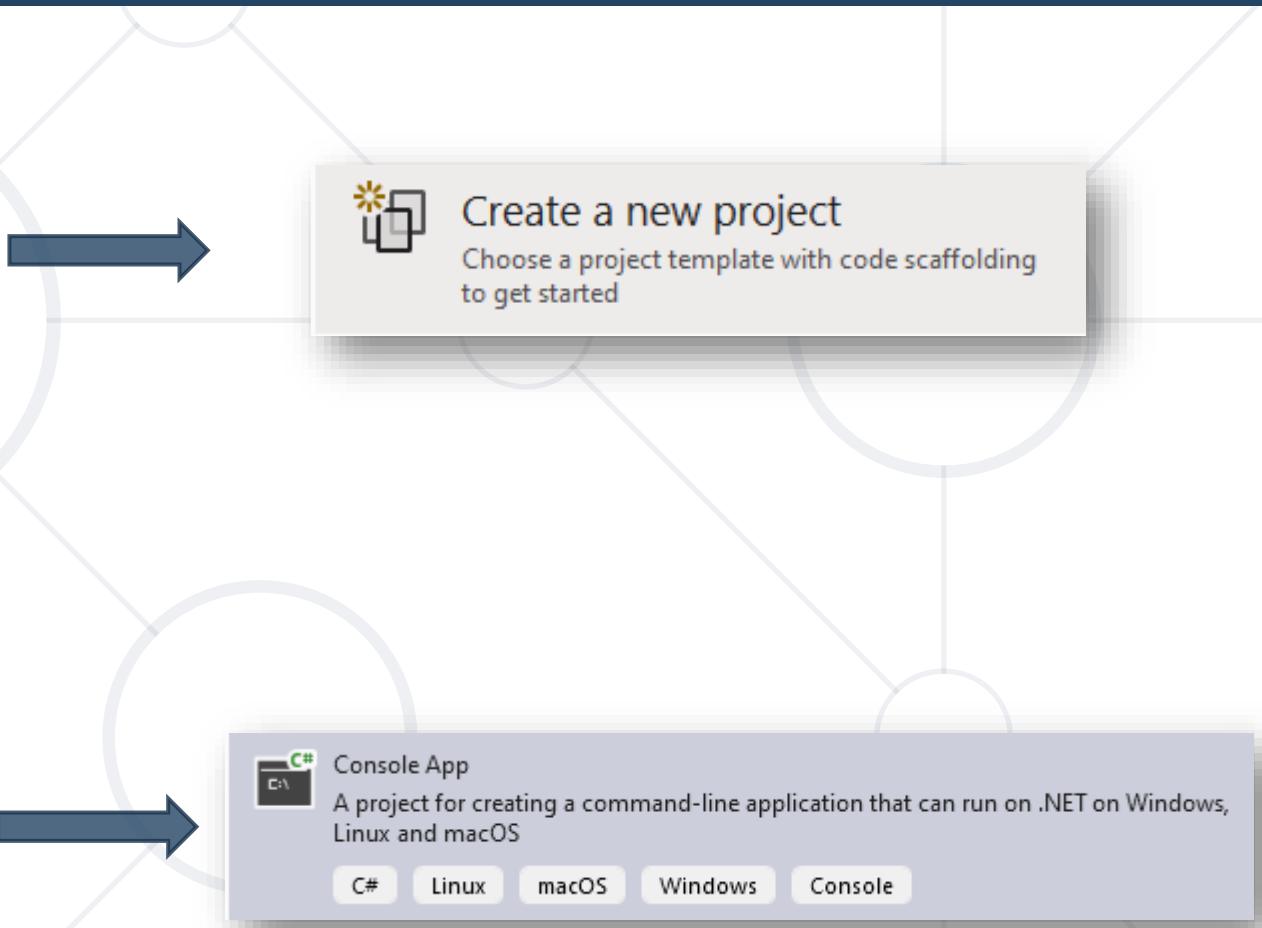


**Да направим първите си конзолни  
програми**

- Средата за разработка е нужна, за да програмирате
  - **Integrated Development Environment (IDE)**
  - **Visual Studio** е среда за разработка на езика C#
- Инсталирайте си **Visual Studio**
  - [Инструкции за инсталация на най-новата версия](#)
  - [Инструкции за инсталация на по-стара версия](#)
- Приложението е **мултиплатформено** (Linux, Mac OS, Windows)

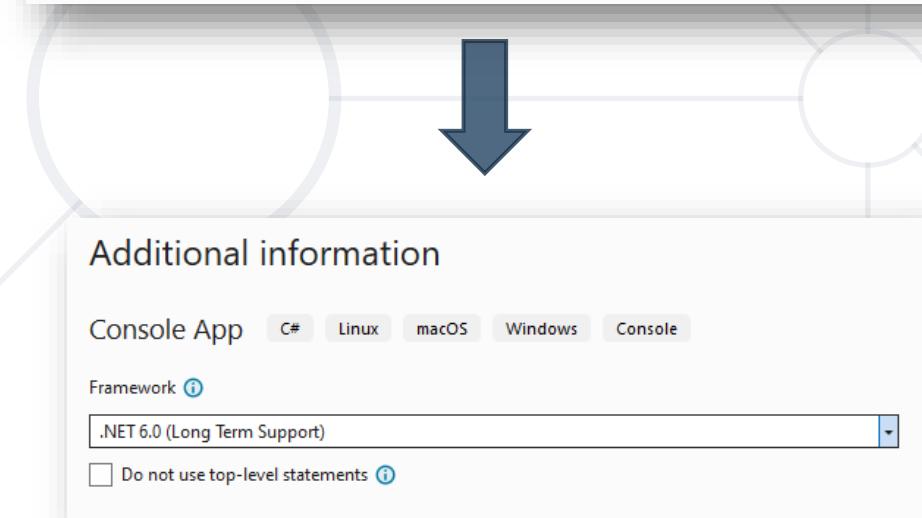
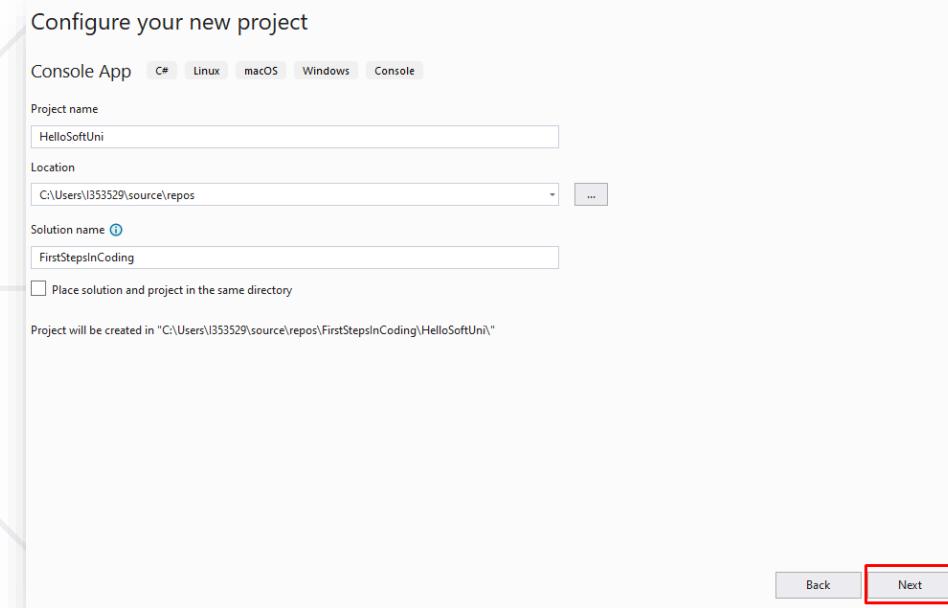
# Създаване на конзолна програма

- Стартрайте **Visual Studio**
- Изберете **Create a new project**
- Изберете **Console Application**
- Изберете **Next**



# Създаване на конзолна програма

- Въведете подходящо име за проекта и директория, в която да се създаде
- Изберете Next
- Изберете Create



- Напишете следния код:

```
Console.WriteLine("Hello SoftUni");
```

- За стартиране на програмата натиснете: **Ctrl + F5**
- Ако няма грешки, програмата ще се изпълни
- Резултатът ще се изпише на конзолата

# Тестване на програмата в Judge

- Тествайте кода си в онлайн judge системата:

Tasks	Points
01. Hello SoftUni	100/100
02. Nums 1...10	0/100
03. Rectangle Area	0/100
04. Inches to Centimeters	0/100
05. Greeting by Name	0/100
06. Concatenate Data	0/100
07. Projects Creation	0/100
08. Pet Shop	0/100
09. Yard Greening	0/100

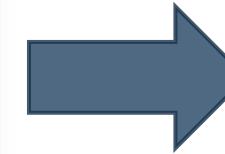
**01. Hello SoftUni**

Remaining time: **620 d, 10 h, 8 m**

**PROBLEM**   **TESTS**

 Условия

```
1 Console.WriteLine("Hello SoftUni");
```



✓✓ **100 /100**

# Типични грешки в C# програмите

- Бъркане на малки и главни букви:

```
Console.WriteLine("Hello SoftUni!");
```

```
console.WriteLine("Hello SoftUni!");
```



# Типични грешки в C# програмите (2)

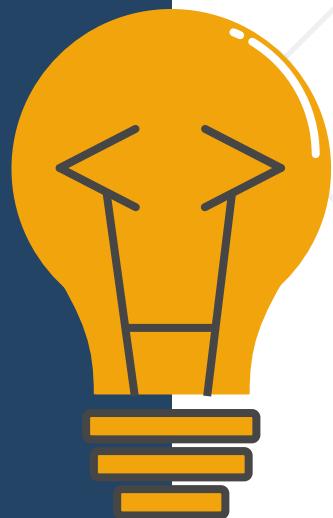
- Липса на ; в края на всяка команда

```
Console.WriteLine("Hello SoftUni!")~
```

- Липсваща кавичка " или липсваща скоба ( или )

```
Console.WriteLine("Hello SoftUni!");~
```

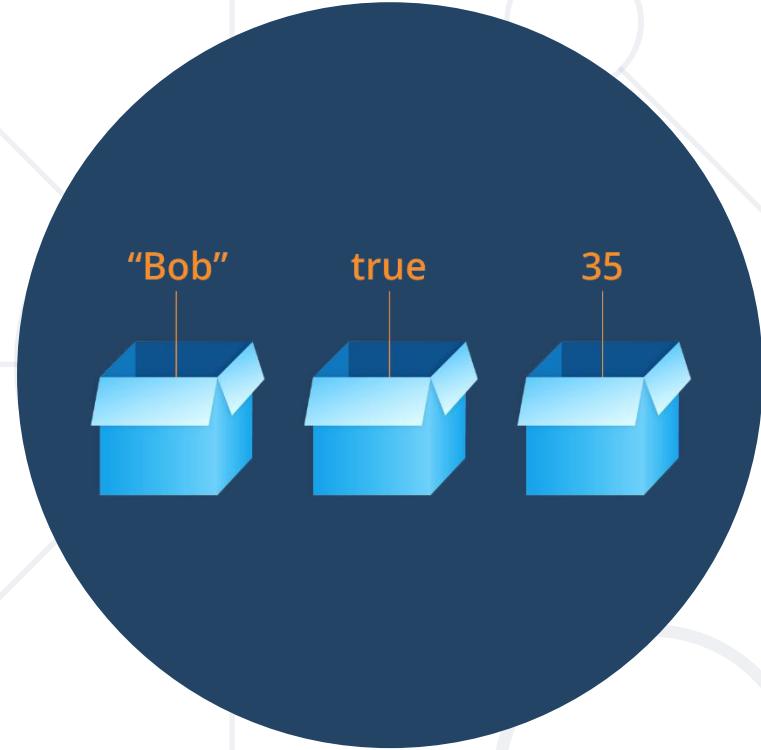
```
Console.WriteLine("Hello SoftUni!";~
```



- Напишете програма, която принтира числата от 1 до 10, всяко на нов ред
- Решение:

```
Console.WriteLine(1);
Console.WriteLine(2);
Console.WriteLine(3);
...
Console.WriteLine(10);
```

# Променливи и типове данни



# Променливи

- Компютрите са машини, които обработват **данни**
  - Данните се записват в компютърната памет в **променливи**
  - Променливите имат **име, тип и стойност**
- **Дефиниране на променлива и присвояване на стойност:**

Тип

Име на променлива

```
int count = 5;
```

Стойност

# Типове данни

- Променливите съхраняват **стойност от даден тип**
  - Число, буква, текст (низ), дата, цвят, картичка, списък, ...
- Типове данни - примери:
  - **int** - цяло число: **1, 2, 3, 4, 5, ...**
  - **double** - дробно число: **0.5, 3.14, -1.5, ...**
  - **string** - текст (низ): **"Здрави", "Hi", "Car", ...**
  - **char** - символ: **'A', '#', '@', '+', ...**





# Работа с конзола

# Прочитане на текст

- Всичко, което **получаваме** от конзолата, идва под формата на **текст**
- Всичко, което **печатаме** на конзолата, се **преобразува в текст**
- Команда за четене от конзолата:  

```
string name = Console.ReadLine();
```

  - Връща ни текста, въведен от потребителя

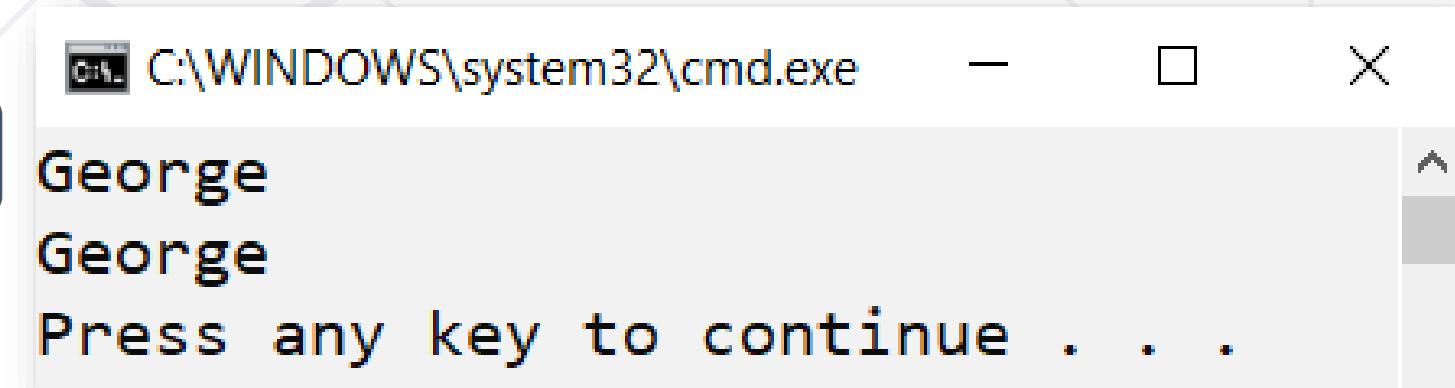


Програма, която **чете** име от конзолата и го **принтира**:

```
string name = Console.ReadLine();
Console.WriteLine(name);
```

Примерен вход

Изход



# Четене на числа

- Четене на цяло число:

```
string input = Console.ReadLine();
int num = int.Parse(input);
```

- Пример: пресмятане на лице на квадрат със страна a:

```
int a = int.Parse(Console.ReadLine());
int area = a * a;
Console.WriteLine(area);
```

Прочитане на цяло  
число на един ред



# Четене на дробно число

- Четене на дробно число от конзолата:

```
string input = Console.ReadLine();
double num = double.Parse(input);
```

- Пример: конвертиране от инчове в сантиметри

```
double inches = double.Parse(Console.ReadLine());
double centimeters = inches * 2.54;
Console.WriteLine(centimeters);
```

Прочитане на дробно  
число на един ред





**Работа с числа**

# Аритметични операции: + и -

- Събиране на числа (**оператор +**):

```
int a = 5;  
int b = 7;  
int sum = a + b; // 12
```



- Изваждане на числа (**оператор -**):

```
int a = int.Parse(Console.ReadLine());  
int b = int.Parse(Console.ReadLine());  
int result = a - b;  
Console.WriteLine(result);
```



# Аритметични операции: \* и /

- Умножение на числа (**оператор \***):

```
int a = 5;  
int b = 7;  
int product = a * b; // 35
```

- Деление на числа (**оператор /**):

```
int a = 25;  
int b = a / 4; // 6 - дробната част се отрязва  
double c = a / 4.0; // 6.25 - дробно делене  
int error = a / 0; // Грешка: деление на 0
```



# Особености при деление на числа

- При деление на цели числа резултатът е **цяло число**:

```
int a = 25;  
Console.WriteLine(a / 4);    // Целочислен резултат: 6  
Console.WriteLine(a / 0);    // Грешка: деление на 0
```

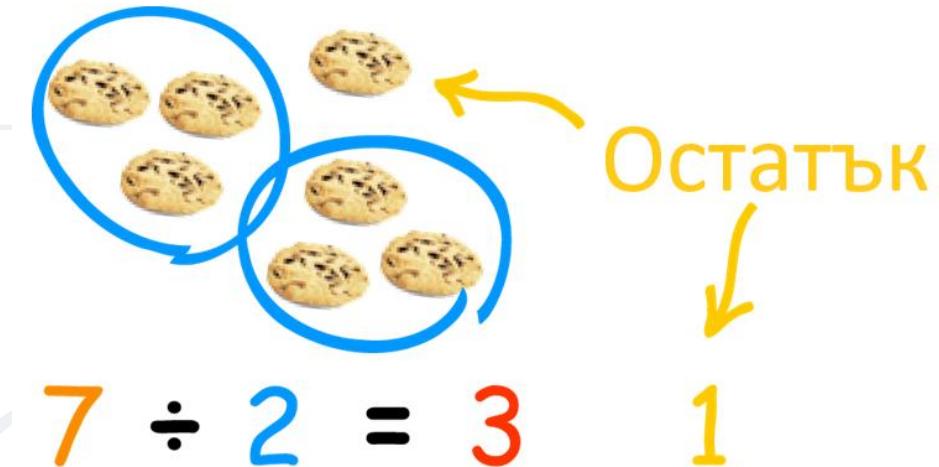
- При деление на дробни числа резултатът е **дробно число**:

```
double a = 15;  
Console.WriteLine(a / 2.0);    // Дробен резултат: 7.5  
Console.WriteLine(a / 0.0);    // Резултат: Infinity  
Console.WriteLine(0.0 / 0.0); // Резултат: NaN
```

# Аритметични операции: %

- Модул/остатък от целочислено деление на числа (оператор %):

```
int a = 7;
int b = 2;
int product = a % b; // 1
```



<code>int odd = 3 % 2;</code>	<code>// 1 - числото 3 е нечетно</code>
<code>int even = 4 % 2;</code>	<code>// 0 - числото 4 е четно</code>
<code>int error = 3 % 0;</code>	<code>// Грешка: деление на 0</code>



**Печатане на конзолата**

# Съединяване на текст и число

- Съединяване на текст и число (**оператор +**):

```
string firstName = "Maria";
string lastName = "Ivanova";
int age = 19;
string str = firstName + " " + lastName + " @ " + age;
Console.WriteLine(str); // Maria Ivanova @ 19
```

Резултатът е  
допояне/конкатенация

```
double a = 1.5;
double b = 2.5;
string sum = "The sum is: " + a + b;
Console.WriteLine(sum); // The sum is 1.52.5
```

# Съединяване на текст и числа

- Можем да форматираме изхода чрез **интерполяция**, която се означава със символа '\$':

```
string firstName = Console.ReadLine();
string lastName = Console.ReadLine();
int age = int.Parse(Console.ReadLine());
string town = Console.ReadLine();
Console.WriteLine($"You are {firstName} {lastName},
a {age}-years old person from {town}.");
```

В къдравите скоби поставяме името на променливата

# Поздрав по име – пример

- Да се **напише програма**, която:
  - Чете от конзолата **име** на човек, въведено от **потребителя**
  - Отпечатва "Hello, <name>!", където **<name>** е **въведеното** преди това **име**
- Примерен вход и изход:

Petar	→	Hello, Petar!
Viktor	→	Hello, Viktor!



# Поздрав по име – решение

```
string name = Console.ReadLine();
Console.Write("Hello, ");
Console.WriteLine(name);
Console.WriteLine("!");
```

Курсорът остава  
на същия ред

```
string name = Console.ReadLine();
Console.WriteLine("Hello, " + name + "!");
```

Долепяне

```
string name = Console.ReadLine();
Console.WriteLine($"Hello, {name}!");
```

Интерполяция

# Какво научихме днес?

- Компютърната програма е поредица команди
- В C# командите се пишат в частта **Main(...)**
- Печатаме с команда **Console.WriteLine(...)**
- Въвеждане на текст и числа
- Аритметичните операции с числа:  
**+ , - , \* , / , () , %**
- Отпечатване на текст по шаблон



# Въпроси?



SoftUni



Software  
University



SoftUni  
Creative



SoftUni  
Digital



SoftUni  
Foundation



SoftUni  
Kids



Finance  
Academy

- Този курс (презентации, примери, демонстрационен код, упражнения, домашни, видео и други активи) представлява **зашитено авторско съдържание**
- Нерегламентирано копиране, разпространение или използване е незаконно
- © СофтУни – <https://softuni.org>
- © Софтуерен университет – <https://softuni.bg>



# Обучения в Софтуерен университет (Софтуни)



- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
  - [softuni.bg](http://softuni.bg)
- Фондация "Софтуерен университет"
  - [softuni.foundation](http://softuni.foundation)
- Софтуерен университет @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)



Software  
University

