

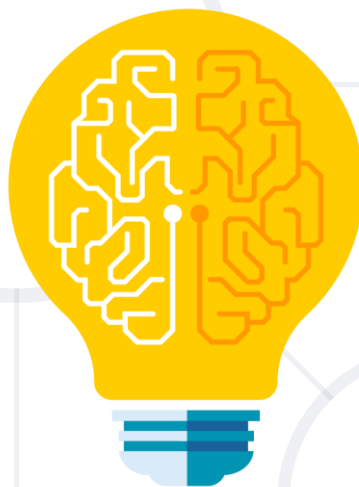
# Прости операции и пресмятания

Работа с конзола, аритметични операции с числа



СофтУни

Преподавателски екип



SoftUni  
Foundation



Софтуерен университет  
<http://softuni.bg>

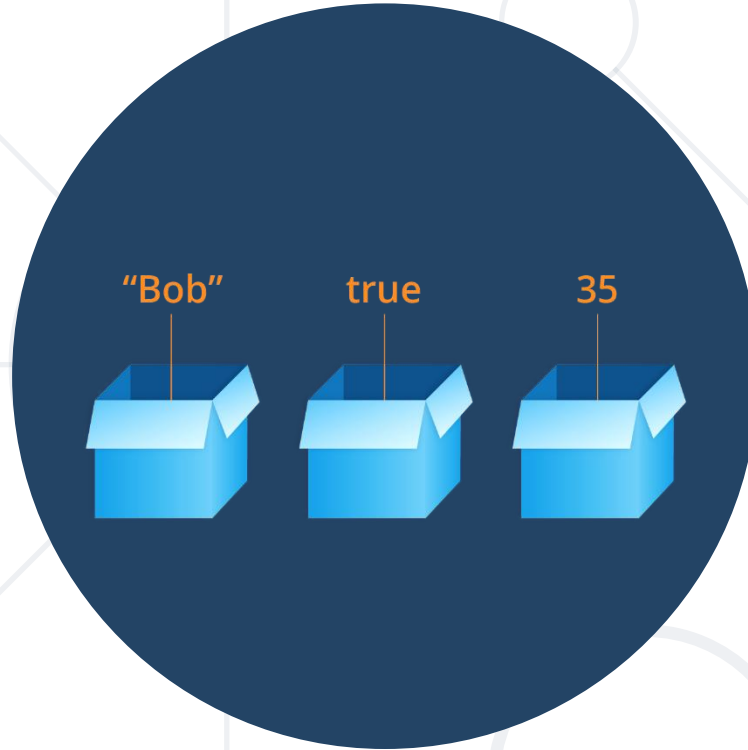
Имате въпроси?

[sli.do](https://sli.do)

**#pb-march**

1. Променливи и типове данни
2. Четене на потребителски вход
3. Прости операции
  - Работа с текст
  - Работа с числа
4. Печатане на екрана
  - Форматиране на изход





# Променливи и типове данни

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

Инициализация

Име на променлива

```
let count = 5;
```

Стойност (от тип число)

- Променливите съхраняват стойност от даден тип
  - Число, буква, текст (низ), дата, списък, ...
- Типове данни – примери:
  - **Number** – число: **7**, **3.14**, **-1.5**, ...
  - **String** – текст: **"Hello"**, **"Здрасти"**, **"p@r0La"** ...
  - **Undefined** – променлива, на която не е присвоена стойност





# **Получаване на потребителски вход**

## **Работа с конзола**

# Четене на текст

- Всичко, което **получаваме** като вход, идва под формата на **аргумент**
  - Всичко, което **печатаме** на конзолата, се **преобразува в текст**
- Получаване на текст:

```
function readText(input) {  
  let str = input.shift();  
}
```





# Четене на текст (2)

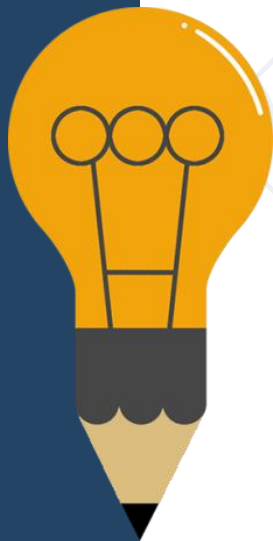
- Пример:

```
function readName(input) {  
  let name = input.shift();  
  console.log(name);  
}  
  
readName([ 'SoftUni' ]);
```



## DEBUG CONSOLE

```
Debugging with inspector protocol  
node simpleCalculations.js  
SoftUni
```



# Четене на числа

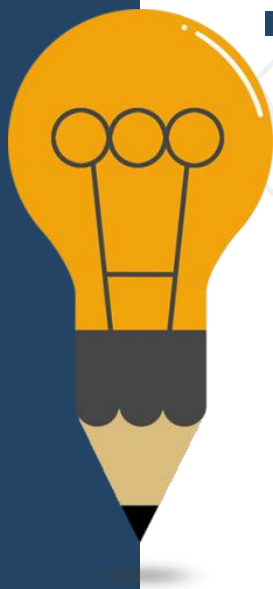
- Получаване на число:

```
function readNumber(input)
  let data = input.shift();
  let num = Number(data);
```

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

```
function squareArea(input) {
  let a = Number(input.shift());
  let area = a * a;
  console.log(area);
}
```

Получаване на  
число на един ред





**Прости операции**  
**Работа с текст и числа**

# Поздрав по име - пример

- Да се напише функция, която:
  - Получава като аргумент **име** на човек
  - Отпечатва "Hello, <name>", където <name> е полученото преди това **име**
- Примерен вход и изход:

Petar



Hello, Petar!

Viktor



Hello, Viktor!



HELLO

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1011#0>

# Поздрав по име - решение

```
function greetingByName(input) {  
  let name = input.shift();  
  let greeting = "Hello, " + name;  
  console.log(greeting);  
}  
greetingByName(["Svetlin Nakov"]);
```



DEBUG CONSOLE ...

Debugging with inspector protocol  
node simpleCalculations.js  
Hello, Svetlin Nakov

Операцията "+" долепя  
текстовата стойност и  
променливата

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1011#0>

# Съединяване на текст и число

- Съединяване на текст и число (оператор +):



```
let firstName = "Maria";  
let lastName = "Ivanova";  
let age = 19;  
let str = firstName + " " + lastName + " @ " + age;  
console.log(str);           // Maria Ivanova @ 19
```

Резултатът е  
долепяне/конкатенация

```
let a = 1.5;  
let b = 2.5;  
let sum = "The sum is: " + a + b;  
console.log(sum);           // The sum is 1.52.5
```

# Аритметични операции: + и -

- Събиране на числа (оператор +):

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

- Изваждане на числа (оператор -):

```
function subtract(input)  
  let a = Number(input.shift());  
  let b = Number(input.shift());  
  let result = a - b;  
  console.log(result);
```



# Аритметични операции: \* и /

- Умножение на числа (оператор \*):

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



- Деление на числа (оператор /):

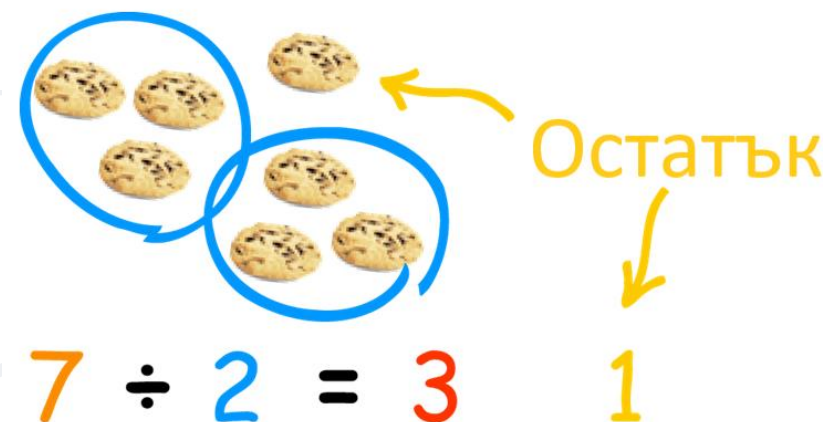
```
let a = 25;  
let i = a / 4; // 6.25 (дробно делене)  
let f = parseInt(a / 4.0); // 6 (дробната част се отрязва)  
let infinity = a / 0; // Infinity (безкрайност)  
let sqrt = Math.sqrt(-1); // получава се NaN
```



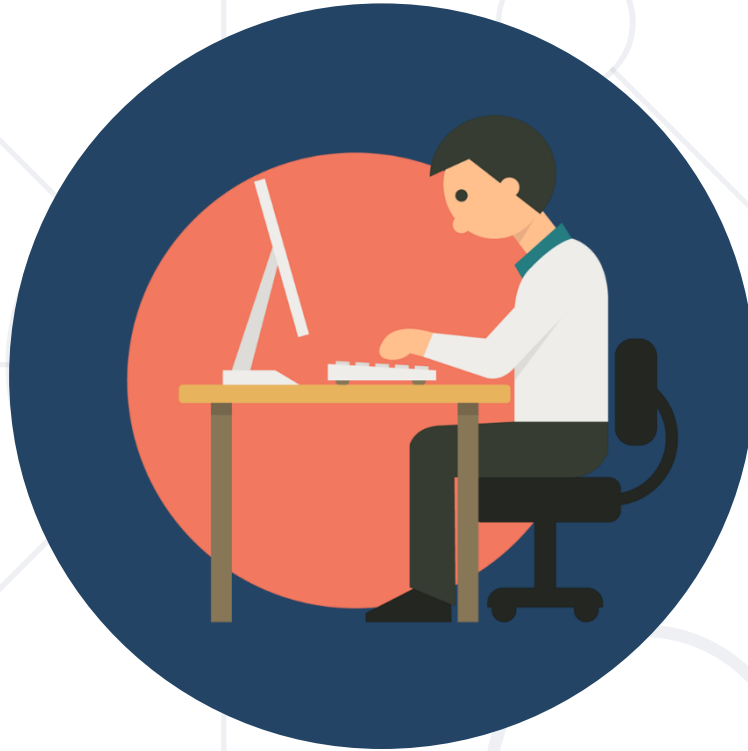


- Модул/остатък от целочислено деление на числа (оператор %):

```
let a = 7  
let b = 2  
let product = a % b // 1
```



```
let odd = 3 % 2 // 1 – числото 3 е нечетно  
let even = 4 % 2 // 0 – числото 4 е четно  
let error = 3 % 0 // получава се NaN
```



# **Задачи с прости изчисления**

## **Решаване на задачи в клас (лаб)**



**Печатане на конзолата**

- При печат на текст, числа и други данни, можем да ги съединим, използвайки интерполация:

``${arg1}, ${arg2}, ${arg3}``

- Пример:

```
function greet(input) {  
  let name = input.shift();  
  console.log(`Hello, ${name}!`);  
}
```

Интерполация на стринг  
се извършва с back-tick(`)

Изразът `${name}` се замества с  
със стойността, която стои зад  
променливата `name`

- Да се напише програма, която:
  - Получава като аргумент име, фамилия, възраст на човек и град, въведени от потребителя
  - Отпечатва "You are <firstName> <lastName>, a <age>-years old person from <town>."
  - Примерен вход и изход:

Вход: `['Petar', 'Petrov', '24', 'Sofia']`

Изход: `You are Petar Petrov, a 24-years old person from Sofia.`

```
function personalInfo(arguments) {  
    let firstName = arguments.shift();  
    let lastName = arguments.shift();  
    let age = Number(arguments.shift());  
    let town = arguments.shift();  
    console.log(`You are ${firstName} ${lastName}, a  
    ${age}-years old person from ${town}.`);  
}
```



**Преобразуване на типове**

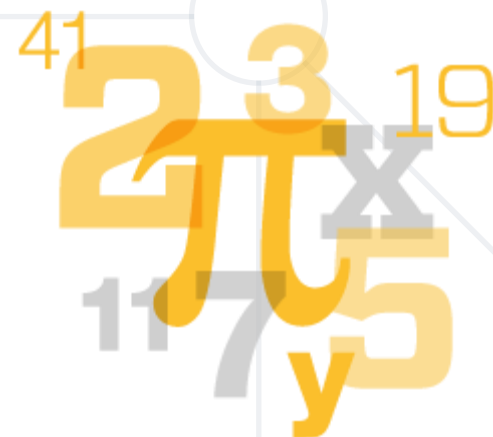
- В програмирането можем да закръгляме дробни числа

- Закръгляне до следващо (по-голямо) цяло число:

```
let up = Math.ceil(23.45); // up = 24
```

- Закръгляне до предишно (по-малко) цяло число:

```
let down = Math.floor(45.67); // down = 45
```





- Отрязване на знаците след десетичната запетая:

```
let trunc = Math.trunc(45.67); // trunc = 45
```

- Форматиране до 2 знака след десетичната запетая:

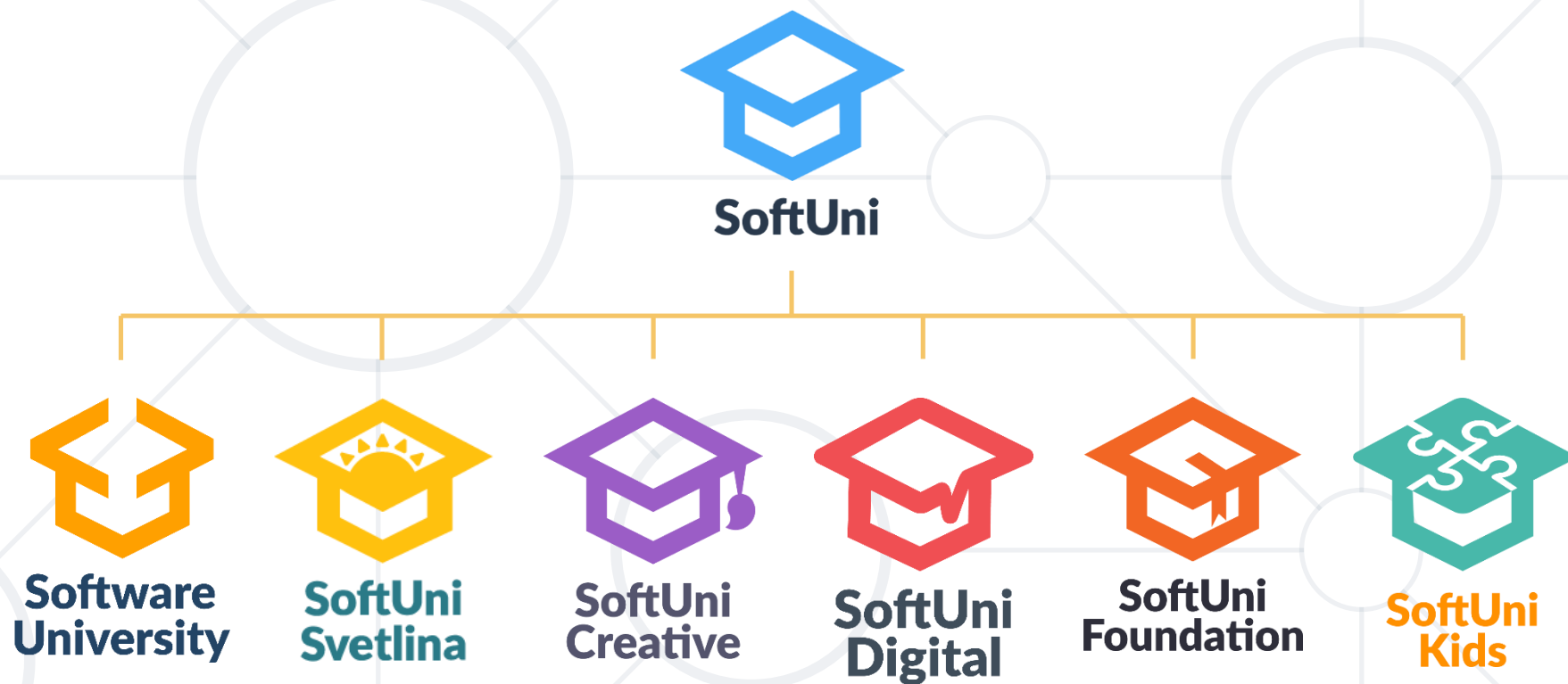
```
(123.456).toFixed(2); // 123.46
```

Брой символи след  
десетичната запетая

- Въвеждане на текст
- Четене на число
- Пресмятания с числа:  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $()$
- Извеждане на текст по шаблон



# Въпроси?



# SoftUni Diamond Partners



**XS**software



**SBTech**  
*we know sports*



telenor



**SoftwareGroup**  
*doing it right*

**NETPEAK**



**SmartIT**



**Postbank**

*Решения за твоето утре*



**INDEAVR**

*Serving the high achievers*



**INFRAGISTICS®**



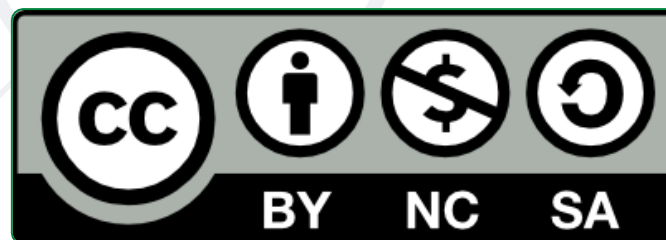
**STEMO®**  
*Computer Systems & Software*

**SUPERHOSTING.BG**

# SoftUni Organizational Partners



- 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 с JavaScript" от Светлин Наков и колектив с лиценз CC-BY-NC-SA

# Обучения в СофтУни

- Software University – High-Quality Education, Profession and Job for Software Developers
  - [softuni.bg](http://softuni.bg)
- Software University Foundation
  - <http://softuni.foundation/>
- Software University @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)
- Software University Forums
  - [forum.softuni.bg](http://forum.softuni.bg)



**Software  
University**

