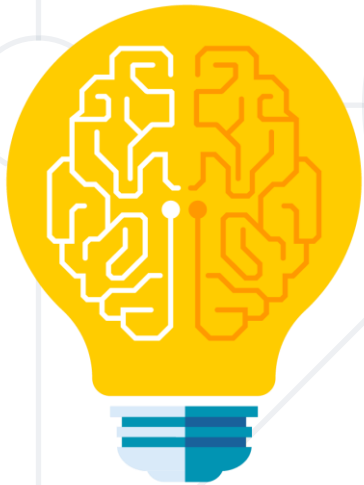


Първи стъпки в програмирането

Работа с конзола, аритметични операции с числа



СофтУни

Преподавателски екип



SoftUni

Софтуерен университет

<https://softuni.bg>

1. Какво е програмиране?
2. Първа програма с **Python** и **PyCharm**
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: **Python, C#, JavaScript, Java, PHP, C, C++**
- Използва се **среда за програмиране (среда за разработка)**
- Пример: **PyCharm, IntelliJ IDEA, Visual Studio, 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**



Интересно за Python

- Един от **топ 3** на най-популярните езици за програмиране
- Един от **най-подходящите** за начинаещи
- Синтаксисът е близък до обикновения **английски език**
- Създаден в **началото 90-те години**
- Поддържа се от **голяма общност** от хора



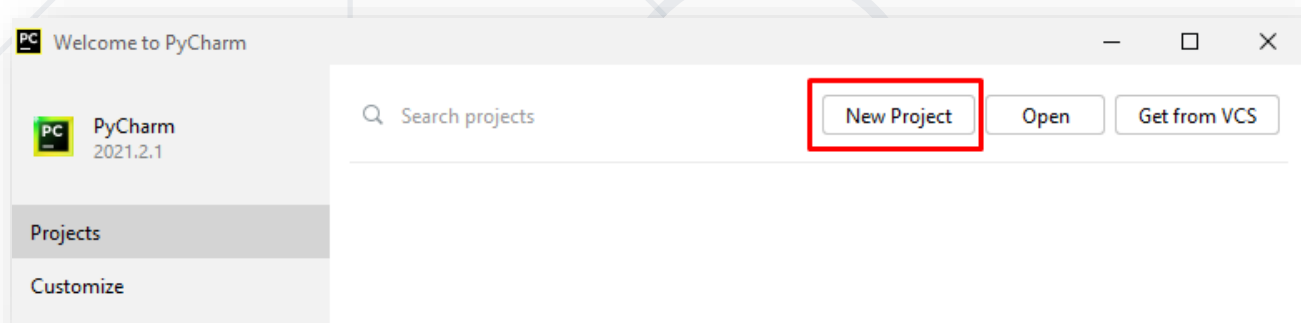


**Да направим първите си конзолни
програми**

- Среда за разработка е нужна, за да програмирате
 - **Integrated Development Environment (IDE)**
 - **PyCharm** е среда за разработка на езика **Python**
- Инсталирайте си **PyCharm Community**
 - Инструкции за инсталация
 - Инструкции за инсталация на по-стара версия
- Приложението е **мултиплатформено** (Linux, Mac OS, Windows)

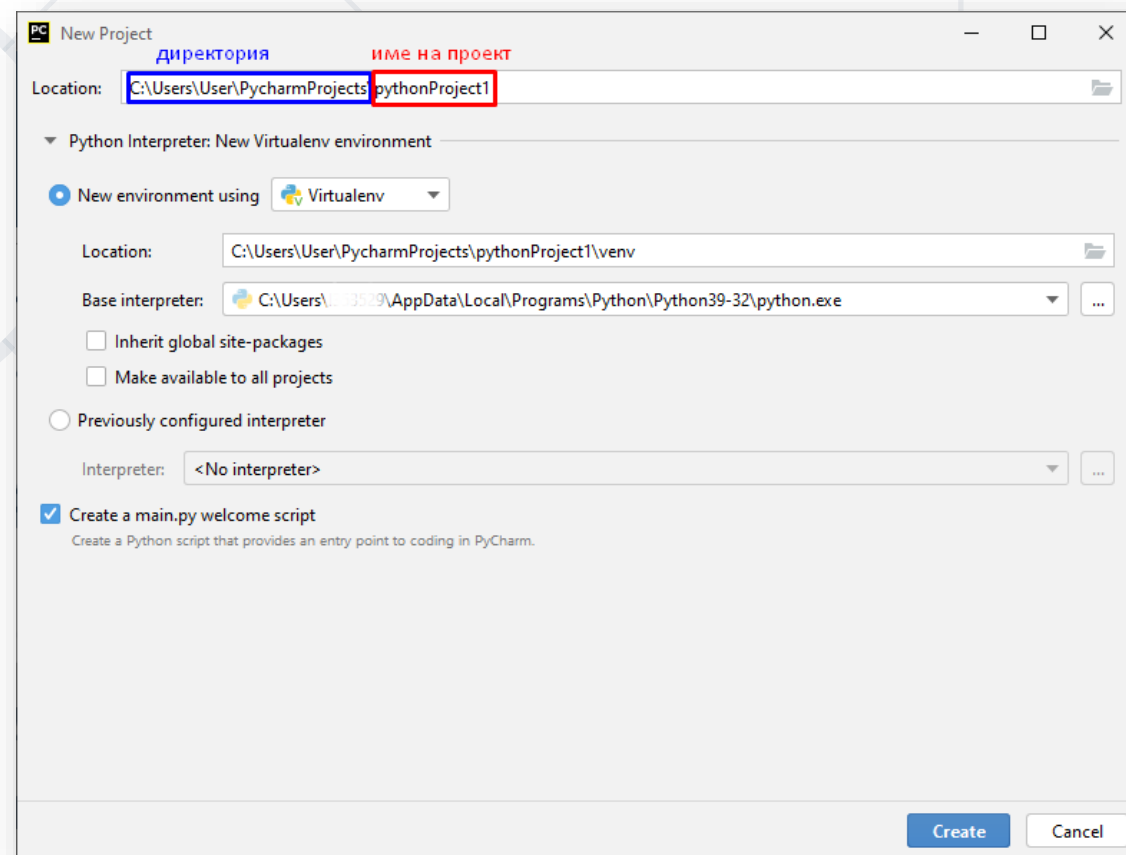
Създаване на конзолна програма

- Стартирайте PyCharm
- Изберете **New Project**



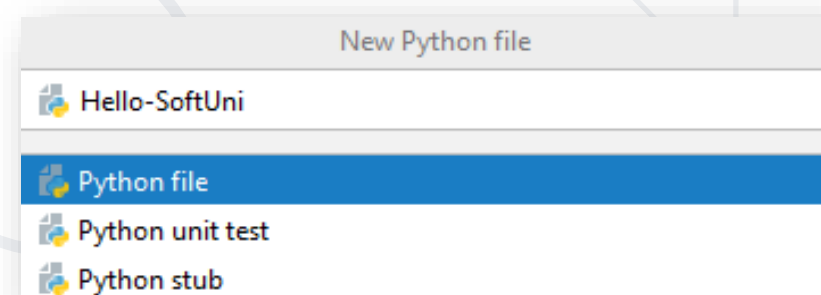
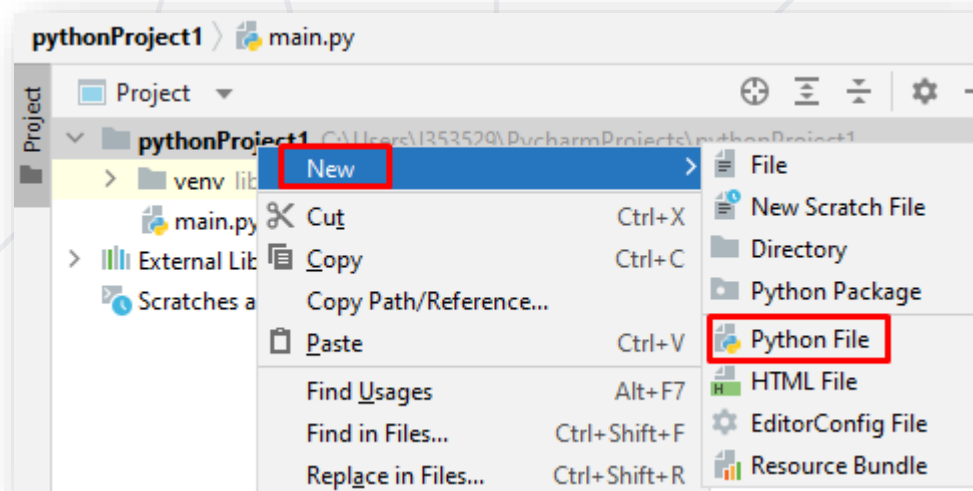
Създаване на конзолна програма (2)

- Въведете подходящо име за проекта и директория, в която да се създаде
- Проверете, че имате конфигуриран Base Interpreter
- Изберете **Create**



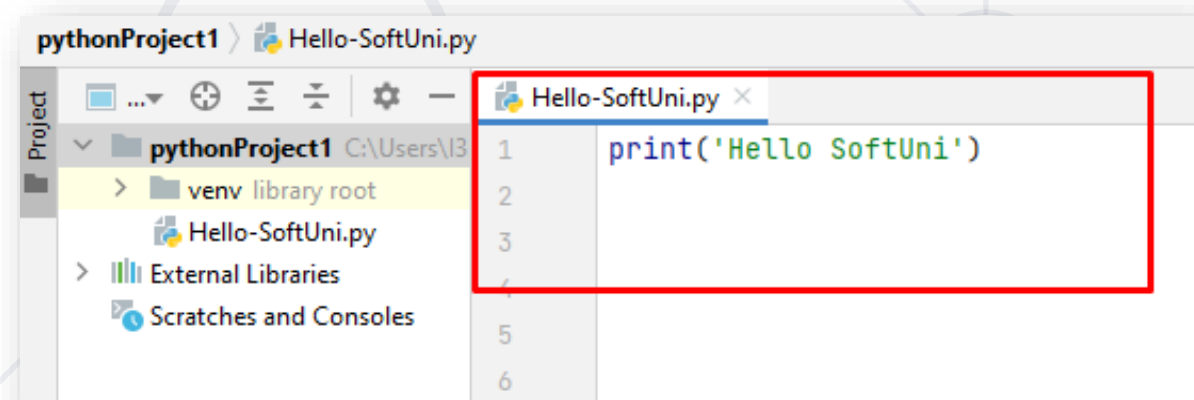
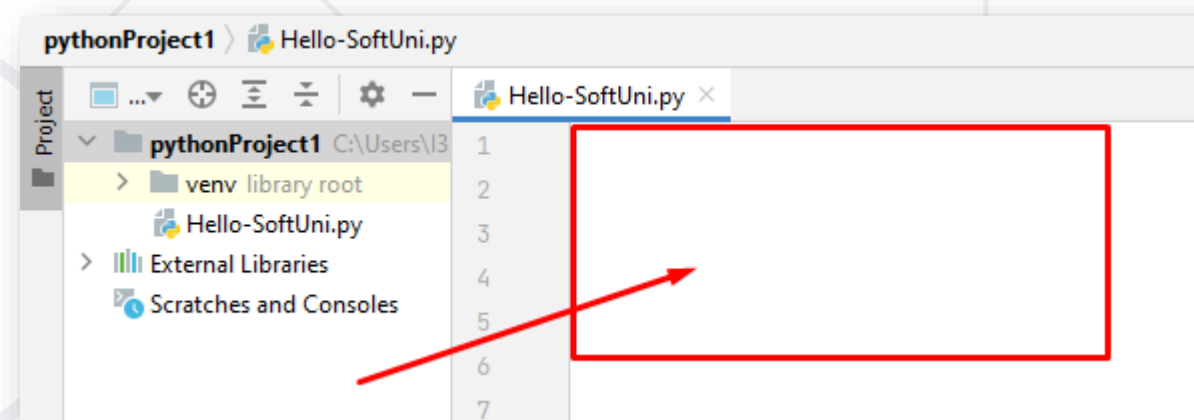
Създаване на конзолна програма (3)

- Десен бутон върху папката на проекта
- Изберете опцията **New** от падащото меню
- Изберете опцията **Python File** от падащото меню
- Задайте подходящо име на проекта си



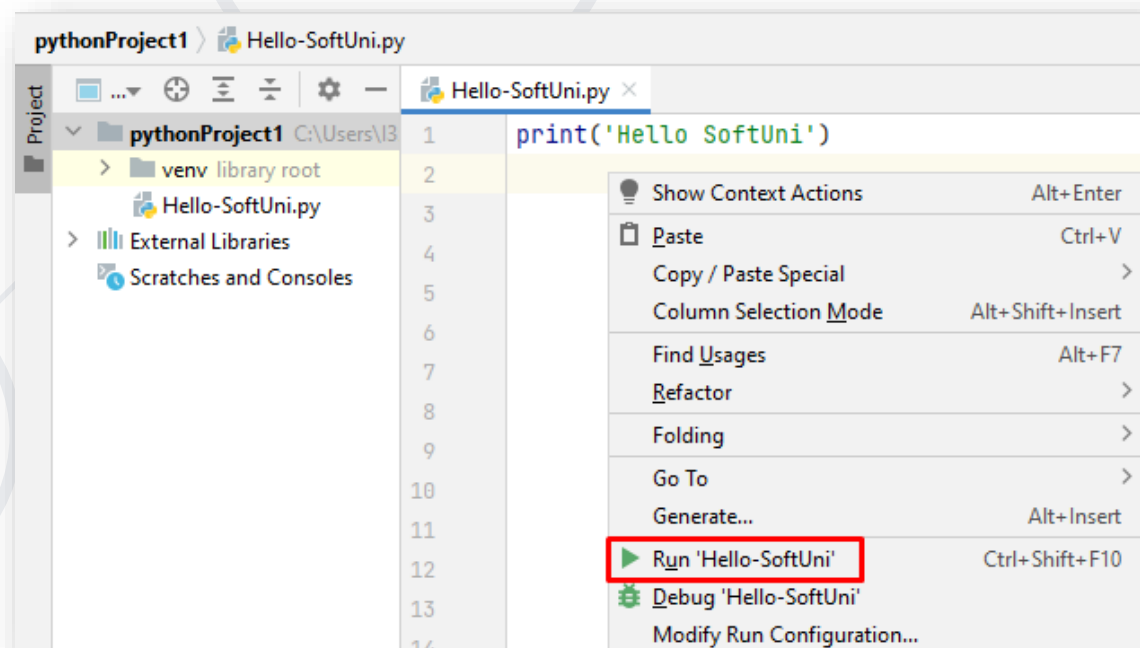
- Кодът на програмата ще напишем в файла "Hello-SoftUni.py", който вече създадохме
- Напишете следния код

```
print('Hello SoftUni')
```



Стартиране на програмата

- Има два начина за стартиране на програмата:
 - Чрез клавишната комбинация: **Ctrl + Shift + F10**
 - Чрез десен бутон -> **Run** (от падащото меню)



Резултат от стартиране на програмата

- Ако няма грешки, програмата ще се изпълни
- Резултатът ще се изпише на конзолата (отдолу):



The screenshot shows the PyCharm IDE interface. The top pane displays a Python file named `Hello-SoftUni.py` with the following code:

```
1 print('Hello SoftUni')
```

The bottom pane shows the output of the program execution. The command executed is:

```
C:\Users\I353529\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:/Users/I353529/PycharmProjects/pythonProject1/Hello-SoftUni.py
```

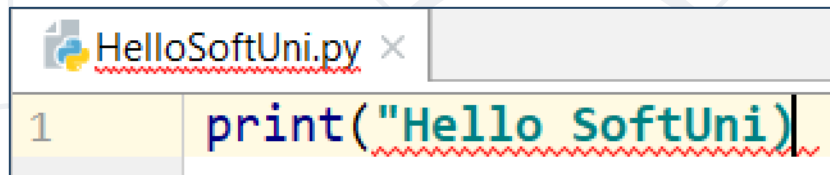
The output is:

```
Hello SoftUni
```

The output text is highlighted with a red rectangle. Below the output, it states: "Process finished with exit code 0".

Типични грешки в Python програмите

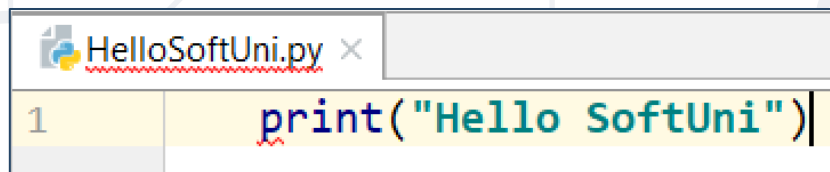
- Грешки на синтаксиса



```
1 print("Hello SoftUni')
```

- Липсват **затварящите кавички** в скобите

- Грешки при индентацията



```
1 print("Hello SoftUni")
```

- Има ненужна табулация пред командата **print**

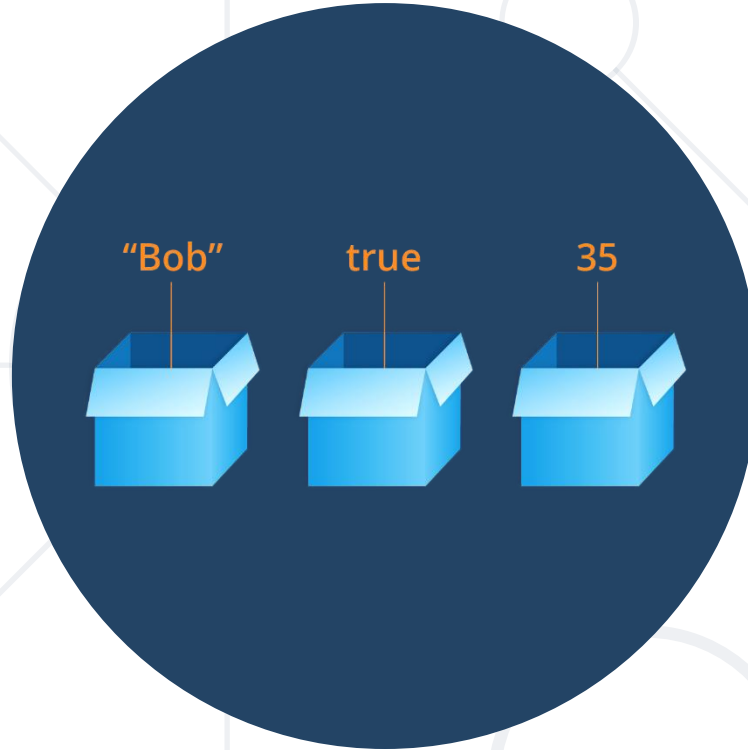


Числата от 1 до 10

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

- Решение:

```
print(1)  
print(2)  
print(3)  
...  
print(10)
```



Променливи и типове данни

- 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:

Име на променлива

`count` = `5`

Стойност (от тип число)

Типове данни

- Променливите съхраняват **стойност от даден тип**
 - Число, текст (низ), дата, цвят, картинка, списък, ...
- Типове данни – примери:
 - `int` – цяло число: 1, 2, 3, 4, 5, ...
 - `float` – дробно число: 0.5, 3.14, -1.5, ...
 - `str` – текст (низ) и символи: 'а', 'Здрасти', 'Hi', ...
- В езикът Python типът се определя от стойността, която се присвоява





Четене на потребителски вход

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:

```
name = input()
```

 - Връща ни текстът, въведен от потребителя



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

```
name = input()  
print(name)
```

Примерен вход

Изход

```
D:\SimpleOperationsAndCalculations  
George  
George  
  
Process finished with exit code 0
```

Четене на числа

- Четене на цяло число:

```
data = input()  
num = int(data)
```

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

```
a = int(input())  
area = a * a  
print(area)
```

Прочитане на цяло
число на един ред

Тестване на решението: <https://judge.softuni.bg/Contests/2423>



Четене на дробно число

- Четене на дробно число от конзолата:

```
data = input()  
num = float(data)
```

- Пример: конвертиране от инчове в сантиметри:

```
inches = float(input())  
centimeters = inches * 2.54  
print(centimeters)
```

Прочитане на дробно
число на един ред





Прости операции

Работа с текст и числа

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!



HELLO

Поздрав по име – решение

```
name = input()
print('Hello, ', end=' ')
print(name)
```

Курсорът остава на
същия ред

```
name = input()
print('Hello, ' + name)
```

Долепяне

Тестване на решението: <https://judge.softuni.bg/Contests/2423>

Съединяване на текст и число

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

Резултатът е
долепяне/конкатенация

```
first_name= 'Maria'  
last_name= 'Ivanova'  
age = 19  
str = first_name + ' ' + last_name + ' @ ' + str(age)  
print(str)
```

Maria Ivanova @ 19

Превръщане на числена
стойност в текст

```
a = 1.5  
b = 2.5  
sum = 'The sum is: ' + str(a) + str(b)  
print(sum)
```

The sum is 1.52.5



Аритметични операции: + и -

- Събиране на числа (**оператор +**):

```
a = 5  
b = 7  
sum = a + b    # 12
```

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

```
a = int(input())  
b = int(input())  
result = a - b  
print(result)
```



Аритметични операции: $*$, $/$, $//$

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

```
a = 5  
b = 7  
product = a * b    # 35
```


- Деление на числа (**оператор $/$ и $//$**):

```
a = 25  
f = a / 4           # 6.25  
i = a // 4          # 6 - целочисленно деление  
error = a / 0       # Грешка: деление на 0
```

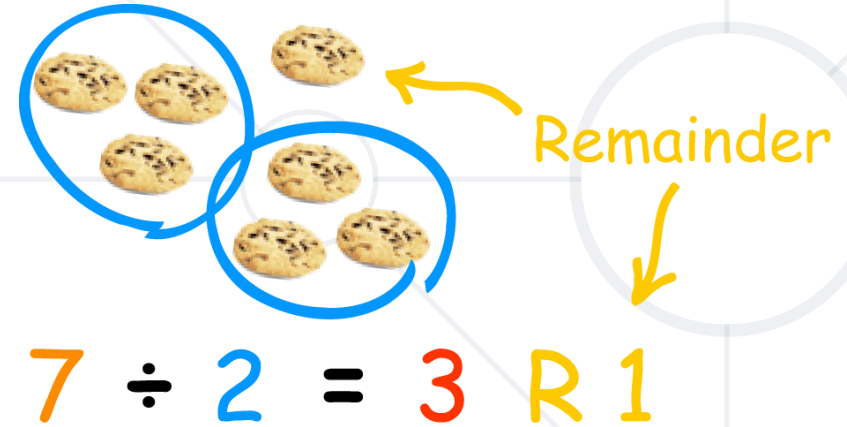


Аритметични операции: %

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



```
a = 7  
b = 2  
product = a % b    # 1
```



```
odd = 3 % 2    # 1 – числото 3 е нечетно  
even = 4 % 2   # 0 – числото 4 е четно  
error = 3 % 0  # Грешка: деление на 0
```




Печатане на конзолата

- Можем да форматираме изхода чрез **интерполация**, която се означава със символа 'f':

```
first_name = input()
last_name = input()
age = int(input())
town = input()
print(f"You are {first_name} {last_name}, а {age}-years old person from {town}.")
```

В къдравите скоби поставяме името на променливата

- Понякога се налага да използваме вече готови програми, за да ни е по-лесно да напишем нашата:
 - За целта трябва да ги "заредим":

```
import Името на библиотеката
```

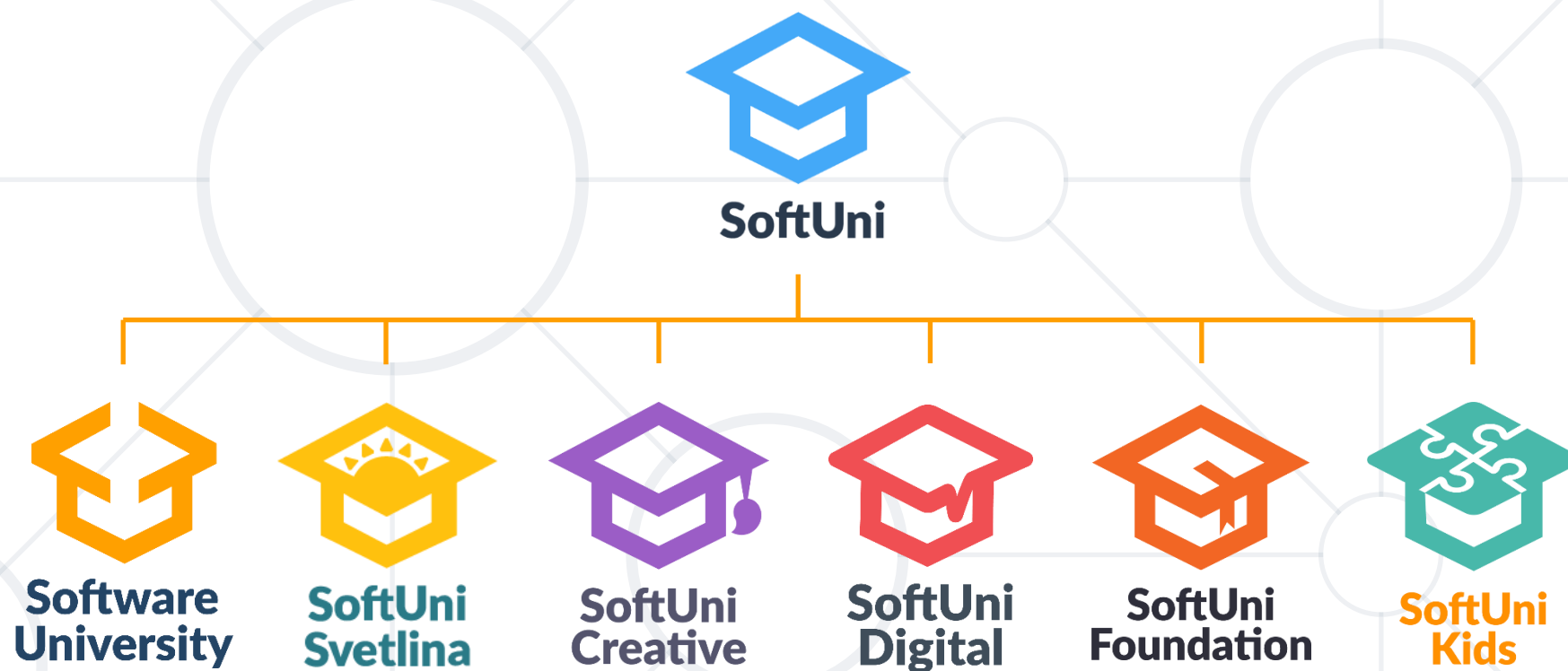
- Пример:

```
import math          # Зарежда библиотеката с име math
import sys           # Зарежда библиотеката с име sys
import math, sys     # Зарежда всички изредени библиотеки
```

- Компютърната програма е поредица команди
- Командите се пишат в .py файловете
- Печатаме с **print(...)**
- Въвеждане на текст и числа
- Аритметичните операции с числа :
+, -, *, /, //, %, ()
- Отпечатване на текст по шаблон



Въпроси?



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



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



Software University

