



Класична гімназія при ЛНУ ім. Івана Франка

Основи програмування мовою Python Узагальнення вивченого матеріалу

Львів, 2016



План

- Проста арифметика
- Вбудовані типи даних
- Використання змінних
- Виведення даних на екран
- Введення даних з клавіатури
- Оператори порівняння
- Умовний оператор if / else
- Цикл while



Проста арифметика

Чотири математичні операції, які можуть виконуватись процесором:

- Додавання +
- Віднімання -
- Множення *
- Ділення /



Проста арифметика

```
In [37]: 2 + 2
```

```
Out[37]: 4
```

```
In [38]: 5 - 3
```

```
Out[38]: 2
```

```
In [39]: (3 + 1) * 2
```

```
Out[39]: 8
```

```
In [40]: 8 / 2
```

```
Out[40]: 4
```



Деякі вбудовані типи даних

- Цілі числа – **int**
 - Приклади: 1, 2, -1, 10, 0 і т. д.
- Дійсні числа – **float**
 - Приклади: 1.0, 0.2, -5.3, 0.25 і т. д.
- Логічний тип даних – **bool**
 - Приклади: True, False
- Рядок тексту – **str**
 - Приклади: "Hello, world!", "abc", "0" і т. д.



Використання змінних

- Змінна – це іменована частина пам'яті, що може містити дані певного типу
- Ім'я змінної – це довільна послідовність символів **A-Z**, **a-z**, **0-9** та символу нижнього підкреслення **_**, однак воно не може починатись з цифри
- Щоб створити змінну в мові Python, потрібно присвоїти їй якесь значення
- Для цього використовують оператор присвоєння **=**



Використання змінних

```
In [33]: x = 2  
         y = 3
```

```
In [34]: x + y
```

```
Out[34]: 5
```

```
In [35]: name = 'Oleksandr'
```

```
In [36]: print 'Hello, ' + name  
Hello, Oleksandr
```



Два варіанти виведення на екран

```
In [1]: print("Hello, world!")  
Hello, world!
```

```
In [2]: print "Hello, world!"  
Hello, world!
```

Python v.2.7 – обидва варіанти працюватимуть

Python v.3.5 – правильним буде тільки перший варіант, другий викличе помилку



Введення даних з клавіатури

```
In [*]: a = input("Чисельник: ")  
        b = input("Знаменник: ")
```

Чисельник: 10

Знаменник:



Введення даних з клавіатури

```
In [17]: a = input("Чисельник: ")  
         b = input("Знаменник: ")
```

Чисельник: 10

Знаменник: 5

```
In [18]: a / b
```

Out[18]: 2



Оператори порівняння

Ось шість операторів порівняння. Їх результатом завжди буде тип bool:

- Більше **>**
- Менше **<**
- Більше або дорівнює **>=**
- Менше або дорівнює **<=**
- Дорівнює **==**
- Не дорівнює **!=**



Умовний оператор

```
if умова:  
    команда 1  
    команда 2  
    ...
```

Команди блоку **if** будуть виконані, якщо умова поверне значення **True**. Інакше будуть виконані команди блоку **else**

```
if умова:  
    команда 1  
    команда 2  
    ...  
else:  
    команда А  
    команда Б  
    ...
```



Умовний оператор

```
In [19]: x = 7
          y = -1

          if x > y:
              print('x більше за y')
          else:
              print('x менше або дорівнює y')

          x більше за y
```



Умовний оператор

```
In [14]: a = input("Чисельник: ")  
         b = input("Знаменник: ")
```

```
Чисельник: 10  
Знаменник: 0
```

```
In [15]: if b == 0:  
         print('На нуль ділити не можна')  
     else:  
         print(a / b)
```

```
На нуль ділити не можна
```



Цикл while

```
while умова:  
    команда 1  
    команда 2  
    ...
```

Команди в блоці циклу while виконуватимуться доти, доки умова не поверне значення **False**



Цикл while

```
In [23]: x = 5  
  
while x > 0:  
    print(x)  
    x = x - 1
```

```
5  
4  
3  
2  
1
```




Цикл while

```
In [24]: x = 5  
  
while x > 0:  
    print(x)|
```

5
5
5
5
5
5
5
5
5
5
5
5
5



Цикл while

```
In [27]: x = 1  
  
while x <= 5:  
    print(x)  
    x = x + 1
```

```
1  
2  
3  
4  
5
```



Цикл while

```
In [28]: x = 10  
  
while x <= 50:  
    print(x)  
    x = x + 10
```

```
10  
20  
30  
40  
50
```



Цикл while

```
In [32]: a = 2

while a <= 20:
    is_prime = True
    b = 2

    while b < a:
        if a % b == 0:
            is_prime = False

        b = b + 1

    if is_prime:
        print(a)

    a = a + 1
```

2
3
5
7
11
13
17
19