

Лабораторне заняття №1

з навчальної дисципліни

Спеціалізовані мови програмування

Python (advanced)

на тему:

ЗМІННІ ТА ТИПИ ДАНИХ

Мета роботи

Ознайомитися з основними типами, математичними та логічними операціями, вводом та виводом даних мови програмування Python3

Хід роботи

- Самостійно на ПК реалізувати програмний код наведений нижче

```
# числа в десятковій системі числення
```

```
dec1 = 8
```

```
dec2 = 42
```

```
dec3 = -3
```

```
dec4 = 25802836572356723058203845293402834028304820938402834023580235777082489436236
```

```
print(type(dec1))
```

```
print(dec2)
```

```
print(dec3)
```

```
print(type(dec4))
```

```
print()
```

```
"""
```

```
block comment
```

```
"""
```

```
# числа в шістнадцятиричній системі числення
```

```
hex1 = 0x9
```

```
hex2 = 0xA
```

```
hex3 = 0xFF
```

```
hex4 = 0x3de
```

```
#
```

```
print(hex1)
```

```
print(hex2)
```

```
print(hex3)
```

```
print(hex4)
```

```
print()
```

число в двійковій системі числення

```
bin1 = 0b11101101
```

```
print(bin1)
```

```
print()
```

число в восьмеричній системі числення

```
oct1 = 0o765
```

```
print(oct1)
```

```
#
```

```
#
```

ціле число з іншого значення

```
string = "15"
```

```
print(type(string))
```

```
number = int(string)
```

```
print(type(number))
```

```
print(number)
```

```
print(number + 5)
```

```
print(string + 5 ) #-- помилка
```

істинне значення

bool1 = True

неправдиве значення

bool2 = False

print(type(bool1))

print(bool2)

приклади дійсних чисел

a = .5123

b = 3.2

приклади дійсних чисел в експоненційній формі запису

c = 3.2e5 # 3.2 * 10**5

d = 1e-3 # 1 * 10**(-3)

print(a, b)

print(c, d , sep='---')

створення дійсного значення

some_float_var = float("0.5")

print(type(some_float_var)) # зі строки

print(float(3)) # з цілого числа

приклади комплексних чисел

$c1 = 2 + 3j$ # $2 + 3i$, 2 – дійсна частина, 3 – уявна частина

$c2 = 5 - 5j$ # $5 - 5i$

$c6 = 5 + 1j$

побудова комплексного числа з дійсних чисел

$a = 2$

$b = 3$

$c3 = \text{complex}(a, b)$

$c4 = \text{complex}(5, -5)$

$\text{print}(c1, c2, c6)$

$\text{print}(c3, c4)$

$\text{print}(\text{type}(c1))$


```
# змінні - імена, які прив'язуються до об'єктів
var = 'I am a string'
print(var)
print(type(var)) # type(var) повертає тип об'єкта, на який посилається var

print()
```

```
# їх можна прив'язувати до нових значень
var = 42
print(var)
print(type(var))

print()
```

```
# проте потрібно розуміти, який тип в даний момент може
# мати об'єкт і змішувати несумісні типи не можна
a = 5
b = 5
print(a + b) # два числа скласти можна
b = '5'
print(a + b) # число з рядком - ні (отримуємо помилку)
```

```
x = 2
y = 8
print(x + y) # додавання
print(x + 3)
print(x - y) # віднімання
print(x * y) # добуток
print(x / y) # ділення
print(x // y) # цілочислене ділення
print(x % y) # остача від ділення
print(x ** y) # внесення в ступінь
print(3.2 * 0.8 - 2 * 5 - 3**3) # арифметичний вираз
print(4 ** 0.5) # внесення в ступінь 0.5 – квадратний корінь
```

```
z = -2
print(abs(z)) # модуль числа
print(pow(z, 2), z ** 2) # квадрат числа
```

```
m = 3.6687656
print(round(m), round(m, 5)) # округлення числа до цілого і до п'яти знаків після коми
```

```
import math # імпортуємо модуль math
```

```
x = 3.265
```

```
PI = math.pi
```

```
NUMBER_E = math.e
```

```
CONST = 5
```

```
print(CONST)
```

```
CONST = 10
```

```
print(CONST)
```

```
# ціле число, найближче ціле знизу, найближче ціле зверху
```

```
print(math.trunc(x), math.floor(x), math.ceil(x))
```

```
print(PI) # константа Пі
```

```
print(NUMBER_E) # число Ейлера
```

```
y = math.sin(PI / 4) # math.sin – синус
```

```
print(round(y, 2))
```

```
#
```

```
y = 1 / math.sqrt(2) # math.sqrt – квадратний корінь
```

```
print(round(y, 2))
```

логічні операції

```
print('and:')  
print(False and False)  
print(False and True)  
print(True and False)  
print(True and True)  
print()
```

```
print('or:')  
# print(False or False)  
# print(False or True)  
print(True or False)  
print(True or True)  
# print()
```

```
print('not:')  
print(not False)  
print(not True)  
print()
```

логічні вирази

a = True

b = False

c = True

f = a and not b or c or (a and (b or c))

print(f)

```
a = 2
```

```
b = 5
```

```
print(a < b)    # менше
```

```
print(b > 3)    # більше
```

```
print(a <= 2)   # менше або дорівнює
```

```
print(b >= 7)   # більше або дорівнює
```

```
print(a < 3 < b) # подвійне порівняння
```

```
print(a == b)   # рівність
```

```
print(a != b)   # нерівність
```

```
print(a is b)   # ідентичність об'єктів в пам'яті
```

```
print(a is not b) # a и b – різні об'єкти (хоча значення їх можуть бути рівні)
```

```
string = "some string"
```

```
second_string = string
```

```
third_string = input('Введіть строку: ')
```

```
print(string is second_string)
```

```
print(string is third_string)
```

```
# приклади строкових значень
```

```
# -*-utf-8*-
```

```
s1 = "Строка 1"
```

```
s2 = 'Строка 2'
```

```
print(s1, s2)
```

```
# конструювання рядка з іншого значення
```

```
s3 = str(8)
```

```
print(s3)
```

```
# багаторядкові рядки
```

```
s4 = """Lesson2. Variables and Data Types
```

```
Some data types explained in this lesson:
```

```
    - int
```

```
    - bool
```

```
    - float
```

```
    - complex
```

```
    - str
```

```
"""
```

```
print(s4)
```

```
# \ використовується, щоб продовжити рядок  
# або будь-який вираз в Python з наступного рядка кода  
s5 = "started\  
    continued"  
print(s5)
```

```
def f():  
    """  
    adasdasd  
    :return:  
    """  
    pass
```

```
print(f.__doc__)
```



```
import math
```

```
str1 = 'hel'
```

```
str2 = 'lo'
```

```
result = str1 + str2 # конкатенація строк
```

```
print(result)
```

```
# msg = f('str' , str1)
```

```
# форматування строк
```

```
a = 48
```

```
b = 73
```

```
message1 = '%d + %d = %d' % (a, b, a + b)
```

```
print(message1)
```

```
t = '{1} - {2} = {0}'
```

```
message2 = t.format(a, b, a - b)
```

```
print(message2)
```

індексація строк

s = 'hello ,World!'

print(s[0]) # індексація починається з нуля

print(s[4]) # четвертий (п'ятий логічно) елемент (символ)

print(s[-1]) # від'ємні числа - індексація з кінця

print(s[2:7]) # символи із другого (включно) по сьомий (не включно)

print(s[2:7:2]) # те ж саме, з крокомдва

print(s.replace(' ', '@'))

Розбиття рядка по роздільнику

print(s.split(" "))

Чи складається рядок з цифр

print(s.isdigit())

print('-----')

Чи складається рядок з літер

print(s.isalpha())

Чи складається рядок з цифр або літер

print(s.isalnum())

Чи складається рядок із символів в нижньому регістрі

print(s.islower())

Чи складається рядок із символів в верхньому регістрі

print(s.isupper())

print('-----')

Чи починаються слова в рядку з великої літери

```
print(s.istitle())
```

Перетворення рядка до верхнього регістру

```
print(s.upper())
```

Перетворення рядка до нижнього регістру

```
print(s.lower())
```

Чи починається строка s із шаблону str

```
print(s.startswith('H'))
```

Чи закінчується строка s шаблоном str

```
print(s.endswith('w'))
```

Символ в ASCII

```
print(ord('a'))
```

Код ASCII в символ

```
print(chr(65))
```

Переводить перший символ рядка в верхній регістр, а всі інші в нижній

```
print(s.capitalize())
```

Повертає відцентрований рядок, по краях якої стоїть символ fill (відступ за замовчуванням)

```
s23 = s.center(50 , '*' )
```

```
print(s23)
```

```
print(len(s23))
```

вказуємо розподільник

```
print(2, 3, 5, sep='; ')
```

```
print('he', 'llo', sep='') # він може бути порожньою строкою
```

вказуємо кінець строки

```
print(1, end=' ')
```

```
print(2, end='\n\n') # два переводи строки
```

```
print('he', end='') # пустий кінець строки
```

```
print('llo')
```

```
# Введення строки
string = input('Введіть строку: ')
print('Ви ввели "', string, '"', sep="")
```

```
input()
```

```
# введення двох чисел
n = int(input('Введіть перше число: '))
m = int(input('Введіть друге число: '))
print('{} + {} = {}'.format(n, m, n + m))
```

Завдання на самостійну роботу

Оформити звіт

Заняття закінчено.
Дякую за увагу!