



Python Basic

Заняття 2:





Зміні

- Python змінні - це просто імена
- У python все є об'єкти
- Об'єкт можна уявити як скриньку, у якій міститься фрагмент даних. Об'єкт має тип (наприклад, тип булевих значень або тип цілих чисел), який визначає, що можна зробити з цими даними.



- Необхідно дотримуватись певних вимог використання імен змінних. Імена змінних можуть містити тільки такі символи:
 - літери в нижньому регістрі (a-z)
 - літери у верхньому регістрі (A-Z)
 - цифри (0-9)
 - нижнє підкреслення (_)
 - Імена не можуть починатися з цифри.



Що таке PER8 ?



- `snake_case` - для змін і функцій.
- `PascalCase` - для класів.
- `SCREAMING_SNAKE_CASE` - для констант



Типи даних



Boolean Type – логічний тип



Numeric Type – числа:

int – ціле число;

float – число з плаваючою точкою (дійсне число);

complex – комплексні число.



Sequence Type – послідовності:

list – список;

tuple – кортеж;

range – діапазон.



Text Sequence Type – рядки;

`str` – рядки.

Це упорядкована послідовність символів яку використовують для зберігання та представлення текстової інформації.



Binary Sequence Types – бінарні послідовності:

`bytes` – байти;

`bytearray` – масиви байт 24;

`memoryview` – спеціальні об'єкти для доступу до внутрішніх даних об'єкта через `protocol buffer`.



Set Types – множини:

`set` – множина;

`frozenset` – незмінювана множина.



Binary Sequence Types – бінарні послідовності:

`bytes` – байти;

`bytearray` – масиви байт 24;

`memoryview` – спеціальні об'єкти для доступу до внутрішніх даних об'єкта через `protocol buffer`.



Mapping Types – словники:
dict – словник.



Коментарі

- Коментарі можуть бути використані для пояснення коду Python.
- Коментарі можна використовувати, щоб зробити код більш читабельним.
- Коментарі можна використовувати для запобігання виконанню під час тестування коду.



Арифметичні операції

- Піднесення до ступеня (**) Піднімає лівий операнд до рівня правого.
- Додавання (+) Оператор додавання
- Віднімання (-) Оператор віднімання
- Множення (*) Оператор множення
- Ділення (/) Оператор ділення
- Обчислення залишку (%) Залишок від ділення
- Обчислення цілої частини (//) Ціла частина від ділення



Оператори порівняння

Більше ніж ($>$) Істинно, якщо лівий операнд більший за правий

Менше ніж ($<$) Істинно, якщо лівий операнд менший за правий

Дорівнює ($==$) Істинно, якщо обидва операнди рівні

Не дорівнює ($!=$) Істинно, якщо операнди не рівні

Більше або дорівнює ($>=$) Істинно, якщо лівий операнд більше або дорівнює правому

Менше або дорівнює ($<=$) Істинно, якщо лівий операнд менше або дорівнює правому



Моржовий оператор



`::=`



Побітові оператори

- & Побітове І
- | Побітове АБО
- ~ Побітове НЕ
- ^ Побітовий XOR
- >> Побітове зрушення вправо
- << Побітове зсув вліво



Операторів ідентифікації

is Істинно, якщо операнди ідентичні

is not Істинно, якщо операнди не ідентичні

Обидва використовуються для порівняння, якщо два значення розташовані в одній і тій же частині пам'яті. Дві рівні мінливі не означає, що вони ідентичні.



Операторів ідентифікації

in Істинно, якщо значення знайдено в послідовності

not in Істинно, якщо значення не знайдено в послідовності

Обидва вони використовуються для перевірки того, чи є значення чи змінна в послідовності



Built-in Functions



Built-in Functions

A

abs()
aiter()
all()
any()
anext()
ascii()

B

bin()
bool()
breakpoint()
bytearray()
bytes()

C

callable()
chr()
classmethod()
compile()
complex()

D

delattr()
dict()
dir()
divmod()

E

enumerate()
eval()
exec()

F

filter()
float()
format()
frozenset()

G

getattr()
globals()

H

hasattr()
hash()
help()
hex()

I

id()
input()
int()
isinstance()
issubclass()
iter()

L

len()
list()
locals()

M

map()
max()
memoryview()
min()

N

next()

O

object()
oct()
open()
ord()

P

pow()
print()
property()

R

range()
repr()
reversed()
round()

S

set()
setattr()
slice()
sorted()
staticmethod()
str()
sum()
super()

T

tuple()
type()

V

vars()

Z

zip()

__import__()



Input() and print()



type() , id()



If... elif ... else in python



```
if expression1:  
    statement(s)  
elif expression2:  
    statement(s)  
elif expression3:  
    statement(s)  
else:  
    statement(s)
```