

Урок №34. Змінні. Прості типи даних Python. Дії з типами даних.

```
a='Hello world!'
b='Привіт світ!'
print(a)
print(b)
```

```
a='Hello world!'
b='Привіт світ!'
print(a), print(b)
```

Слово **Змінна** у програмуванні позначає місце для зберігання інформації, такої як числа, текст, переліки чисел та тексту. У програмуванні слово змінна позначає іменоване місце для зберігання даних, наприклад чисел, тексту, списків з числами або символами і так далі. Також змінну можна розглядати як ярлик, яким позначені якісь дані.

Символ “=” в Python називають символом **присвоєння**. Він використовується для надання значень певним іменам.

1. **Назва змінної** може складатися лише з малих (a..z) та великих (A..Z) літер англійського алфавіту, цифр (0..9) та знаку нижнього підкреслення (“_”);
2. Імена не можуть починатися з цифр;
3. Ім'я змінної повинно мати певний сенс (не обов'язково)
4. Ім'я змінної залежить від регістра (variable, VaRiAbLe, VARIABLE – це різні імена)

У виразі присвоєння змінна завжди фігурує зліва від знака рівності (=).

Але вона може з'явитися і праворуч. Це вкрай корисна властивість, з якою ви зіштовхнетеся в багатьох програмах. Найчастіше вона застосовується в операції інкремента змінної (збільшення на певну величину) або в протилежній операції - декремента (зменшення на певну величину).

Змінній в будь-якому місці програми можна привласнити інше значення (ярлик переміщається на новий елемент).

У програмі на мові Python зв'язок між даними і змінними встановлюється за допомогою знака =. Така операція називається **присвоєнням**. Наприклад, вираз **a=4** означає, що на об'єкт (дані) у певній області пам'яті посилається ім'я a і звертатися до них тепер слід за цим іменем.

Алгоритм створення змінної:

- Вводимо змінну і присвоюємо їй числове значення
- Надрукувати змінну
- Переглядаємо результат

Дані та їх типи

Можна помітити, що все, що ми робимо, ми робимо над чимось - якимись предметами або об'єктами. Ми міняємо властивості об'єктів та їх можливості. Програми для комп'ютерів також маніпулюють якимись об'єктами (назвемо їх поки даними).

Очевидно, дані бувають різними. Дуже часто комп'ютерній програмі доводиться працювати з числами і рядками.

Числа у свою чергу також бувають різними: цілими, дробовими, можуть мати величезне значення або дуже довгу дробову частину. При знайомстві з мовою програмування Python ми точно зіткнемося з трьома типами даних:

Цілі числа (**integer**) - додатні і від'ємні цілі числа, а також 0 (наприклад, 4, 687, -45, 0).

Числа з плаваючою крапкою (**float point**) - дробові числа (наприклад, 1.45, -3.789654, 0.00453). Примітка: роздільником цілої і дробової частини служить крапка, а не кома.

Рядки (**string**) - набір символів, укладених в лапки (наприклад, "ball", "What is your name?", 'DkfjUUv', '6589'). Примітка: лапки в Python можуть бути одинарними або подвійними.

Перетворення типів

Часто виникає необхідність перейти від одного типу даних до іншого. Наприклад, число, яке виводиться на екран, спочатку може знадобитися перетворити в текст. Цю операцію за вас виконує команда `print`, але бувають ситуації, коли потрібно поміняти тип без виведення інформації на екран або здійснити перетворення рядка в число (таке перетворення команда `print` виконувати не вміє). Ця операція називається перетворенням типів (`type conversion`). Яким чином вона реалізується? Насправді інтерпретатор Python не «перетворює» типи один в одного. Він створює з вихідного елемента новий елемент потрібного вам типу. Ось ряд функцій, які вміють виконувати цю операцію:

float () створює нове число з плаваючою точкою (десятькове) з рядка або цілого числа.

int () створює нове ціле число з рядка або числа з плаваючою крапкою.

str () створює новий рядок з числа (або значення будь-якого іншого типу).

Дужки в кінці вказують, що перед нами не команди мови Python (такі як `print`), а вбудовані в нього функції. Детально функції будуть розглянуті трохи пізніше, а на даному етапі вам досить знати, що значення яке підлягає перетворенню вказується в дужках. Найкраще помітно це на прикладах.

Перетворення цілого в десяткове

Візьмемо ціле число і за допомогою функції `float ()` створимо на його основі число з плаваючою точкою:

```
a = 12
b = float(a)
print(b)
```

Зверніть увагу на десяткову точку і завершальний 0 при виведенні змінної `b`. Саме за цими ознаками ми бачимо, що перед нами не ціле, а десяткове число. При цьому змінна `a` не змінюється, так як на початкове значення функція `float ()` не впливає - вона створює нове значення. В інтерактивному режимі досить ввести ім'я змінної (без команди `print`), і інтерпретатор Python виведе на екран її значення. (Ви вже робили так) Однак в програмах це не працює.

Перетворення десяткового в ціле

Тепер спробуємо зробити зворотний перехід - візьмемо десяткове число і за допомогою функції `int ()` отримаємо з нього ціле:

```
c = 36.6  
d = int(c)  
print(d)
```

Ми створили нову змінну `d`, яка представляє собою цілу частину змінної `c`. Розглянемо ще один приклад:

```
e = 99.98  
f = int(e)  
print(f)
```

Незважаючи на те що 99.98 - це майже 100, відповідь буде 99. Функція `int()` не виконує округлення, а просто відкидає дробову частину. Тому вона дає нам не найближче ціле, а менше з двох цілих.

Функція `type ()` для змінної `a` виводить `class 'str'`, що відповідає рядку, для змінної `b` - `class 'float'`, а , для змінної `c` - `class 'int'`. Вам більше не потрібно нічого вгадувати!

Е-нотація

В е-нотації наше число виводиться на екран як `3.8E16` або як `3.8e16`. В даному випадку буква `E` позначає ступінь 10. Тобто фактично у нас написано 3.8×10^{16} .

У більшості мов програмування, в тому числі і в Python, букву `E` можна писати як в верхньому (`E`), так і в нижньому (`e`) регістрі .

При записі дуже маленьких чисел, наприклад `0.00000000000001752`, застосовується піднесення в негативну ступінь. Тобто можна записати 1.752×10^{-13} , або `1.752e-13`. Негативна ступінь означає, що десяткову точку потрібно зрушити на вказану кількість розрядів не праворуч, а ліворуч:

Увага! Під час роботи з комп'ютером дотримуйтеся правил безпеки та санітарно-гігієнічних норм.

Алгоритм роботи

Рівень 1. (1-3 бали)

Частина 1. Змінні та операції з ними

1. Створити змінні та присвоїти їм значення

Змінній **DOG** присвоїти значення 22

Змінній **FROG** присвоїти значення 4

Змінній **CAT** присвоїти значення 98

Змінній **HORCE** присвоїти значення 13

2. Додати всі змінні та надрукувати результат на екран

Рівень 2. (3-6 балів)

Частина 2. Перетворення типів даних

1. Присвоїти змінній **a** значення 61.7
2. Перетворити значення на ціле число і присвоїти отримане число змінній **b**
3. Надрукувати змінну **b**
4. Присвоїти змінній **c** значення 45
5. Перетворити значення на дійсне число і присвоїти отримане число змінній **d**
6. Надрукувати змінну **d**

Рівень 3.(6-9 балів)

Частина 3. Обчисліть площу і периметр прямокутної кімнати шириною 12.5 метрів і довжиною 16.7 метра

Підказка. Ширині і довжині присвоїти числові значення і виконати арифметичні дії

Рівень 4 (10-12 балів)

Частина 4. Як дізнатися, скільки часу займає подорож на автомобілі? Словами формула виражається так: «час подорожі дорівнює відстані, поділеній на швидкість». Напишіть програму, яка порахує, скільки часу займе переміщення на 200 кілометрів зі швидкістю 80 кілометрів на годину, і відобразить відповідь на екрані.