

Основи синтаксису Python: змінні, оператори, типи даних

Змінні

- Змінна в Python - це як контейнер, у якому ви можете зберігати інформацію, наприклад, числа або текст. Цей контейнер має ім'я, яке ви придумуєте. Наприклад, ви можете створити змінну з ім'ям "число" і в ній зберегти число, наприклад, 5.
- Отже, змінна - це спосіб зберігати дані, щоб потім використовувати їх у вашій програмі. Ви можете змінювати дані у цьому контейнері, і вони будуть доступні для вас, коли вам потрібно.

Оператори

Оператори в Python - це спеціальні символи або ключові слова, які використовуються для виконання різних дій зі змінними та даними.

Ось кілька основних операторів:

Арифметичні оператори:

- `+` (додавання): Додає два числа.
- `*` (множення): Перемножує два числа.
- `/` (ділення): Ділить одне число на інше.
- `-` (віднімання): Віднімає одне число від іншого.

Оператори порівняння:

- `==` (рівно): Перевіряє, чи рівні два значення.
- `!=` (не рівно): Перевіряє, чи не рівні два значення.
- `<` (менше): Перевіряє, чи одне число менше за інше.
- `>` (більше): Перевіряє, чи одне число більше за інше.

Логічні оператори:

- `and` (логічне "і"): Повертає True, якщо обидва вирази є True.
- `or` (логічне "або"): Повертає True, якщо хоча б один із виразів є True.
- `not` (логічне "не"): Інвертує значення, змінюючи True на False і навпаки.

Типи даних

Змінні в Python мають типи даних, які визначають, які значення вони можуть містити і як з ними можна взаємодіяти. Основні типи даних в Python включають:

- Цілі числа (integers): Це цілі числа без десяткової частини. Наприклад, `5` або `-10`.
- Дійсні числа (floats): Це числа з десятковою частиною. Наприклад, `3.14` або `-0.5`.
- Рядки (strings): Рядки використовуються для зберігання тексту. Наприклад, `"Привіт, світ!"` або `'Python'`.
- Булеві значення (booleans): Це два можливі значення - `True` (правда) або `False` (неправда). Вони використовуються для умовних виразів і логічних операцій.
- Списки (lists): Список - це впорядкована послідовність елементів, які можуть бути різних типів. Наприклад, `[1, 2, 3]` або `['яблуко', 'банан', 'апельсин']`.
- Кортежі (tuples): Кортеж схожий на список, але він є не змінним, тобто його неможливо змінити після створення. Кортежі визначаються за допомогою круглих дужок. Наприклад, `(1, 2, 3)`.
- Словники (dictionaries): Словник - це колекція пар ключ-значення. Ви можете отримувати доступ до значень, використовуючи ключі. Наприклад, `{ 'ім'я': 'Петро', 'вік': 30 }`.
- Множини (sets): Множина - це колекція унікальних елементів без порядку. Наприклад, `{1, 2, 3}`.
- None: `None` - це спеціальне значення, яке вказує на відсутність даних або відсутність значення.
- Класи та об'єкти (classes and objects): Python підтримує об'єктно-орієнтоване програмування, тому ви можете створювати свої власні типи даних, використовуючи класи.

Вбудовані функції для роботи з ТИПАМИ ДАНИХ

У Python існують вбудовані функції, які допомагають працювати з різними типами даних.

Ось кілька прикладів цих функцій:

Функції для чисел:

- `int(x)`: Перетворює значення `x` у ціле число.
- `float(x)`: Перетворює значення `x` у дійсне число.

Функції для рядків:

- `len(s)`: Повертає довжину рядка `s`.
- `str(x)`: Перетворює значення `x` у рядок.

Функції для списків та кортежів:

- `len(lst)`: Повертає кількість елементів у списку чи кортежі `lst`.
- `lst.append(x)`: Додає елемент `x` в кінець списку `lst`.

Функції для роботи зі словниками:

- `dct.keys()`: Повертає список ключів у словнику `dct`.
- `dct.values()`: Повертає список значень у словнику `dct`.
- `dct.items()`: Повертає список кортежів (ключ, значення) у словнику `dct`.