

Тема: Кортежі. Множини. Словники.

Мета: Pozнайти основні принципи роботи та отримати практичні навички використання та обробки кортежів, множин, словників на мові Python.

Література

Васильєв О.М. Програмування мовою Python – Тернопіль: Навчальна книга – Богдан, 2019. – 504с. стор. 243-247, 251-298

Зміст роботи.

Завдання 1.

Створіть довільний кортеж із 10 елементів, визначте його тип та перетворіть у список. Результати виведіть на екран.

Завдання 2.

З раніше створеного кортежу виведіть на екран елементи з 4 по 8.

Підрахуйте скільки елементів кортежу дорівнюють 5.

Завдання 3.

Створіть список з елементами що повторюються. Перетворіть його в множину. Результат виведіть на екран. Як змінився зміст множини по відношенню до списку. Висновок вивести на екран.

Завдання 4.

Відпрацюйте команди доповнення множини елементами, вилучення елементів, пошуку елемента. Результати вивести на екран.

Завдання 5.

Задайте дві множини числами в діапазоні [1,20]. За допомогою функції **intersection()** визначте їх перетин та об'єднання за допомогою функції **union()**. Отримані множини виведіть на екран.

Завдання 6.

Дано два словника: `dictionary_1 = {'a': 300, 'b': 400}` і `dictionary_2 = {'c': 500, 'd': 600}`. Об'єднайте їх в один за допомогою вбудованих функцій мови Python.

Завдання 7.

Задайте два списки однакової довжини. Необхідно створити з них словник таким чином, щоб елементи першого списку були ключами, а елементи другого - відповідно значеннями нашого словника.

Методичні рекомендації

Кортеж із заданим вмістом створюється також як список, тільки замість квадратних дужок використовуються круглі.

```
>>> a = (1, 2, 3, 4, 5)
```

```
>>> print(type(a))
```

```
<class 'tuple'>
```

```
>>> print(a)
```

```
(1, 2, 3, 4, 5)
```

можно використати функцію `tuple()`.

```
>>> a = tuple((1, 2, 3, 4))
```

```
>>> print(a)
```

```
(1, 2, 3, 4)
```

Для перетворення списку в кортеж і навпаки:

```
>>> tpl = tuple(lst)
```

```
>>> lst = list(tpl)
```

Створити порожню множину можна за допомогою функції `set()`. А якщо з елементами, то за допомогою їх перерахування в фігурних дужках

```
>>> v = {'A', 'C', 4, '5', 'B'}
```

```
>>> z = set([0,1,2,3,4,5])      # z = {0, 1, 2, 3, 4, 5}
```

Мова Python підтримує наступні операції над множинами:

- `in` – перевірка елементу на входження в множину;
- `-` (мінус) – різниця множин;
- `|` – об'єднання множин;
- `&` – перетин множин;
- `^` – симетрична різниця.

Python дозволяє нам вносити нові елементи в множину за допомогою функції `add()`.

Елементи можуть бути видалені за допомогою обох методів `discard()` і `remove()`.

Пам'ятайте, що метод `discard()` не видаватиме помилку, якщо елемент не був знайдений в безлічі. Однак, якщо метод `remove()` використовується і елемент не був знайдений, виникне помилка.

```
num_set = {1, 2, 3, 4, 5, 6}
```

```
num_set.discard(7)
```

```
print(num_set)
```

У мові Python, щоб створити словник можна використати один з наступних способів:

- з допомогою оператора присвоювання `=` та фігурних дужок, в яких через кому розміщуються пари **ключ:значення**;
- з допомогою конструктора `dict()`.

```
C = { 'Pi':3.1415, 'Exp':1.71 }
```

```
dictionary = {'drink': 'juice', 'food': 'spaghetti', 'cutlery': 'fork'}  
print(dictionary['cutlery'])
```

Результат: fork

Щоб видалити значення із словника, також використовують ключ та функцію `del`.

```
del dictionary['cutlery']  
print(dictionary)
```

Результат: {'drink': 'juice', 'food': 'spaghetti'}

Також використовуємо ключ, якщо необхідно змінити значення у словнику.

```
dictionary['cutlery'] = 'spoon'  
print(dictionary)  
Результат: {'drink': 'juice', 'food': 'spaghetti', 'cutlery': 'spoon'}
```

Функція `zip()` дозволяє створити словник шляхом об'єднання списків ключів та значень.

```
# Створення словника з списків ключів та значень  
Numbers = dict(zip([1, 2, 3], ['One', 'Two', 'Three']))  
print(Numbers) # {1: 'One', 2: 'Two', 3: 'Three'}
```

Контрольні запитання

1. Що спільне та чим відрізняються кортежі та списки у мові Python?
2. Назвіть способи створення кортежів, множин, словників у Python.
3. Яким чином здійснюється звернення до елементів кортежів, множин, словників в мові програмування Python?
4. З якого числа починається нумерація елементів кортежів, множин, словників в мові Python?
5. Яка функція в мові Python дозволяє отримати кількість елементів кортежів, множин, словників?
6. Як можна змінити значення елемента кортежів, множин, словників?