

Python: кортежи | Я.Шпора

Кортеж — неизменяемая последовательность: после того, как кортеж создан, в него нельзя добавить элементы, удалить или изменить их. Кортежи занимают в памяти меньше места, чем списки.

Как объявить кортеж

Перечислить элементы в круглых скобках, разделяя их запятыми:

```
digits = (1, 2, 3, 4, 5)
fruits = ('яблоко', 'банан', 'вишня')
print(type(fruits))
# Вывод в терминал: <class 'tuple'>
```

С помощью функции `tuple()`, которая принимает в качестве аргумента коллекцию (например, строку или список) и преобразует её в кортеж:

```
# Передать строку в функцию tuple().
letters = tuple('яблоко')
print(type(letters), letters)

# Передать список в функцию tuple().
apples = tuple(['Голден', 'Карамелька', 'Антоновка'])
print(type(apples), apples)

# Вывод в терминал:
# <class 'tuple'> ('я', 'б', 'л', 'о', 'к', 'о')
# <class 'tuple'> ('Голден', 'Карамелька', 'Антоновка')
```

Можно создать и пустой кортеж: `tuple()`. Но особого смысла в таком кортеже нет, так как в него нельзя добавить новые элементы.

Если присвоить переменной несколько значений, разделённых запятыми, Python автоматически «упакует» эти значения в кортеж:

```
fruits = 'яблоко', 'банан', 'вишня'  
print(type(fruits))  
print(fruits)
```

```
# Вывод в терминал:  
# <class 'tuple'>  
# ('яблоко', 'банан', 'вишня')
```

Работа с кортежами

Обращение к элементу кортежа

Получить значение элемента кортежа можно по индексу. Индексация начинается с нуля.

Распаковка

Операция позволяет присвоить значения элементов кортежа переменным:

```
fruits = 'яблоко', 'банан', 'вишня'  
  
tree_1, tree_2, tree_3 = fruits  
  
print(tree_1)  
# Вывод в терминал: яблоко  
print(tree_3)  
# Вывод в терминал: вишня
```

Сортировка

Кортеж нельзя отсортировать. Однако из неотсортированного кортежа можно создать новый, отсортированный с помощью функции `sorted()`:

```
vegetables = ('Огурцы', 'Помидоры', 'Баклажаны', 'Кабачки')  
print(vegetables)  
# Вывод в терминал: ('Огурцы', 'Помидоры', 'Баклажаны', 'Кабачки')  
  
# Переменной vegetables_sorted присвоен результат выполнения функции sorted()  
vegetables_sorted = sorted(vegetables)
```

```
print(vegetables_sorted)
# Вывод в терминал: ['Баклажаны', 'Кабачки', 'Огурцы', 'Помидоры']

# Распечатать исходный кортеж.
print(vegetables)
# Вывод в терминал: ('Огурцы', 'Помидоры', 'Баклажаны', 'Кабачки')
```

По умолчанию функция `sorted()` сортирует элементы в порядке возрастания, от меньшего к большему, и возвращает список. Можно изменить направление сортировки, добавив параметр `reverse=True`, и сортировка будет выполнена от большего к меньшему, по убыванию.

Удаление

Из кортежа нельзя удалить отдельные элементы, можно удалить только весь кортеж с помощью оператора `del`:

```
garden_vegetables = ('Огурцы', 'Помидоры', 'Баклажаны', 'Кабачки')

del garden_vegetables
```

Параметр `*args`: произвольное число аргументов в функции

```
def boasting(*args):
    # В теле функции обрабатывается параметр args (здесь уже без звёздочки)
    # чтобы выяснить, какого типа данные переданы в тело функции.
    print(type(args))
    print('Теперь у меня на грядке растёт', args)

# Вызов функции с одним аргументом.
boasting('арбуз')

# Вызов функции с несколькими аргументами.
boasting('тыква', 'ананас', 'картошка', 'турнепс')

# Вызов функции без аргументов.
boasting()
```

```
# Вывод в терминал:
# <class 'tuple'>
# Теперь у меня на грядке растёт ('арбуз',)
# <class 'tuple'>
# Теперь у меня на грядке растёт ('тыква', 'ананас', 'картошка', 'турнепс')
# <class 'tuple'>
# Теперь у меня на грядке растёт ()
```

Параметр `*args` можно совмещать с позиционными и именованными параметрами. При этом:

- Позиционные параметры должны быть указаны до `*args`.
- Именованные параметры (или со значениями по умолчанию) должны следовать после `*args`.

```
def boasting(farm_name, *args):
    print(type(args))
    print('На ферме', farm_name, 'на грядках растёт', args)

# Вызов с несколькими аргументами. Первым - название фермы:
boasting('Овощной Клондайк', 'тыква', 'ананас', 'картошка', 'турнепс')

# Вывод в терминал:
# <class 'tuple'>
# На ферме Овощной Клондайк на грядках растёт ('тыква', 'ананас', 'картошка', 'турнепс')
```

Мудрец знает: шпаргалка хранит, а человек - действует.

Аль Горитм, персидский просветитель