

Кортеж

Кортеж (tuple) – это неизменяемая структура данных, которая по своему подобию очень похожа на список. Кортежи могут состоять из элементов разных типов, перечисленных через запятую. Кортежи заключаются в круглые скобки. Изменять его элементы нельзя (TypeError).

Преимущества кортежей:

1. Позволяют обезопасить данные от случайного изменения.
2. Экономия места - кортежи в памяти занимают меньший объем по сравнению со списками
3. Прирост производительности, который связан с тем, что кортежи работают быстрее, чем списки (т.е. на операции перебора элементов и т.п. будет тратиться меньше времени).
4. Можно использовать в качестве ключа у словаря

Для tuple определены операции конкатенации и повторения:

Раздаточный материал № 75

```
storm_1 = ('Lightning')
Union = (' and ')
storm_2 = ('Thunder')
print(storm_1 + Union + storm_2)
```

Результат: Lightning and Thunder

```
dog_do = ('woof!',)
print(dog_do * 3)
```

Результат: ('woof!', 'woof!', 'woof!')

Из кортежа можно извлекать элементы и брать срезы:

Раздаточный материал № 76

```
>>> a[3]
89
>>> a[1:3]
(2.13, 'square')
```

Создание пустого кортежа

Раздаточный материал № 77

```
>>> a = ()
>>> print(type(a))
<class 'tuple'>
```

Кортеж с заданным содержимым:

Раздаточный материал № 78

```
>>> a = (1, 2, 3, 4, 5)
>>> print(type(a))
<class 'tuple'>
>>> print(a)
(1, 2, 3, 4, 5)
>>> print(*a)
1 2 3 4 5
```

```
>>> a = tuple('hello, world!')
```

```
>>> a  
('h', 'e', 'T', 'T', 'o', ',', "'", 'w', 'o', 'r', 'T', 'd', '!')
```

Можно воспользоваться функцией tuple():

Раздаточный материал № 79

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

Доступ к элементам кортежа осуществляется через указание индекса.

Раздаточный материал № 80

```
>>> a = (1, 2, 3, 4, 5)  
>>> print(a[0])  
1  
>>> print(a[1:3])  
(2, 3)
```

Удалить отдельные элементы из кортежа невозможно. Но можно удалить кортеж целиком:

Раздаточный материал № 81

```
>>> del a  
>>> print(a)  
Traceback (most recent call last):  
File "<pyshell#28>", line 1, in <module>  
print(a)  
NameError: name 'a' is not defined
```

На базе кортежа можно создать список, верно и обратное утверждение.

Раздаточный материал № 82

```
>>> lst = [1, 2, 3, 4, 5]  
>>> print(type(lst))  
<class 'list'>  
>>> print(lst)  
[1, 2, 3, 4, 5]  
>>> tpl = tuple(lst)  
>>> print(type(tpl))  
<class 'tuple'>  
>>> print(tpl)  
(1, 2, 3, 4, 5)
```

Обратная операция также является корректной:

```
>>> tpl = (2, 4, 6, 8, 10)  
>>> print(type(tpl))  
<class 'tuple'>  
>>> print(tpl)  
(2, 4, 6, 8, 10)  
>>> lst = list(tpl)  
>>> print(type(lst))  
<class 'list'>  
>>> print(lst)  
[2, 4, 6, 8, 10]
```

Кортежи могут содержать списки, также как списки могут быть вложенными в другие списки.

Раздаточный материал № 83

```
>>> nested = (1, "do", ["param", 10, 20])
```

Список внутри кортежа изменить можно

```
>>> nested[2][1] = 15
```

```
>>> nested
```

```
(1, 'do', ['param', 15, 20])
```

Выражения типа nested[2][1] используются для обращения к вложенным объектам. Первый индекс указывает на позицию вложенного объекта, второй – индекс элемента внутри вложенного объекта.

Для объектов кортежей определены два метода index и count.

Раздаточный материал № 84

```
>>> T = (1, 2, 3, 2, 4, 2)      # Методы кортежей в Python 2.6, 3.0  
                                # и последующих версиях  
>>> T.index(2)                # Смещение первого появления элемента 2  
1  
>>> T.index(2, 2)              # Смещение появления элемента 2 после  
смещения 2  
3  
>>> T.count(2)                # Сколько всего элементов 2?  
3
```

Распаковка кортежа требуется для обработки значений кортежа и предполагает, что элементы кортежа будут присвоены отдельным переменным

Раздаточный материал № 85

```
tuplex = (4, 6, 2, 8, 3, 1)  
a, b, *c = tuplex
```

Результат: 4 6 [2, 8, 3, 1, 9]

Переменные a и b содержат целочисленные переменные, в c помещается список.
Исходный кортеж tuplex остается неизменным.

Вопросы для самоподготовки

Упражнения к раздаточному материалу № 85.

1. Вывести на экран кортеж tuplex два раза.
2. Вывести на экран из кортежа tuplex элементы со 2 по 4
3. Найти сумму первого и второго элементов кортежа tuplex.
4. Найти сумму первого и последнего элементов кортежа tuplex.
5. Найти разницу первого и предпоследнего элементов кортежа tuplex.
6. Найти количество элементов списка с.
7. Вывести на экран элемент кортежа tuplex со значением 8.