

Практическое задание.

Тема: Словари и их методы.

Цель: Изучение создания словарей, их изменение и взаимодействие с ними через встроенные методы.

Задачи: Закрепить знания о создании словаря и его синтаксисе. Изучить и применить встроенные методы для работы со словарями.

Теоретическая часть.

Добавление элементов в словарь.

Для начала вспомним из чего состоит словарь: словарь = {ключ: значение}

Чтобы добавить элемент в словарь, вам нужно привязать значение к новому ключу. В отличие от списков и кортежей, в работе со словарями методы `add()`, `insert()` и `append()` вам не помощники. Тут необходимо создать новый ключ. Позже он будет использоваться для хранения значения.

Добавляются элементы в словарь так: `dict[key] = value`

`dict` – название словаря.

`key` – новый ключ словаря.

`value` – новое значение для ключа.

Пример добавления в словарь элементов.

Входные данные:

```
>>> my_dict = {'key1': 'значение1', 'key2': 'значение2'}
>>> print(my_dict)
>>> my_dict['key3'] = 'значение3'
>>> print(my_dict)
```

Выходные данные:

```
>>> {'key1': 'значение1', 'key2': 'значение2'}
>>> {'key1': 'значение1', 'key2': 'значение2', 'key3':
'значение3'}
```

Для того, чтобы пользователь мог ввести значение в словарь с клавиатуры, данные необходимо заменить переменными.

Пример добавления в словарь элементов с клавиатуры.

Входные данные:

```
>>> my_dict = {'key1': 'значение1', 'key2': 'значение2'}
>>> print(my_dict)
>>> k = input('Введите новый ключ: ')
>>> v = input('Введите новое значение: ')
>>> my_dict[k] = v
>>> print(my_dict)
```

Выходные данные:

```
>>> {'key1': 'значение1', 'key2': 'значение2'}
>>> Введите новый ключ: key3
>>> Введите новое значение: значение3
>>> {'key1': 'значение1', 'key2': 'значение2', 'key3':
'значение3'}
```

В зависимости от необходимых типов данных ключей или значений, тип данных ввода может отличаться str, int, float и т.д.

Удаление элементов из словаря.

Удаление элементов из словаря возможно несколькими способами. Рассмотрим наиболее простые и понятные из них.

Метод pop()

Синтаксис метода .pop():

`dict.pop(key, value)`

`dict` – название словаря.

`pop` – метод для удаления элементов.

`key` – Первый параметр. Ключ словаря, который необходимо удалить.

`value` – Второй параметр (необязательный). Значение ключа.

Пример удаления элементов из словаря методом pop.

Входные данные:

```
>>> my_dict = {'key1': 'значение1', 'key2': 'значение2',
'key3': 'значение3'}
>>> print(my_dict)
>>> my_dict.pop('key1')
```

```
>>> print(my_dict)
```

Выходные данные:

```
>>> {'key1': 'значение1', 'key2': 'значение2', 'key3':  
'значение3'}
```

```
>>> {'key2': 'значение2', 'key3': 'значение3'}
```

Ключевое слово del

Синтаксис del:

```
del dict[key]
```

dict – название словаря.

del – ключевое слово для удаления элементов.

key – ключ словаря, который необходимо удалить.

Пример удаления элементов из словаря ключевым словом del.

Входные данные:

```
>>> my_dict = {'key1': 'значение1', 'key2': 'значение2',  
'key3': 'значение3'}
```

```
>>> print(my_dict)
```

```
>>> del my_dict['key2']
```

```
>>> print(my_dict)
```

Выходные данные:

```
>>> {'key1': 'значение1', 'key2': 'значение2', 'key3':  
'значение3'}
```

```
>>> {'key1': 'значение1', 'key3': 'значение3'}
```

Решение для удаления нескольких ключей из словаря.

Самый лёгкий способ для решения данной проблемы – это цикл for.

Рассмотрим пример.

В качестве примера, рассмотрим словарь, в котором ключи будут предметами, а значения их количеством. Уберём из него не подходящие по теме «Комплектация ПК» предметы.

```
>>> my_dict = {'Клавиатура':12, 'Компьютерная мышь':20,  
'Молоко':12, 'Булочка с маком':8, 'Системный блок':26,  
'Монитор':12}
```

Здесь несколько предметов, которые необходимо убрать. Для удобства, можно воспользоваться списком и поместить удаляемые ключи в него.

```
>>> del_key = ['Молоко', 'Булочка с маком']
```

Теперь, начнём убирать предметы, по принципу «если такой ключ есть в словаре – удалить ключ»

Вся программа будет выглядеть следующим образом.

Входные данные:

```
>>> my_dict = {'Клавиатура':12, 'Компьютерная мышь':20,  
'Молоко':12, 'Булочка с маком':8, 'Системный блок':26,  
'Монитор':12}
```

```
>>> print(my_dict)
```

```
>>> del_key = ['Молоко', 'Булочка с маком']
```

```
>>> for i in del_key: my_dict.pop(i)
```

```
>>> print(my_dict)
```

Выходные данные:

```
>>> {'Клавиатура': 12, 'Компьютерная мышь': 20, 'Молоко': 12,  
'Булочка с маком': 8, 'Системный блок': 26, 'Монитор': 12}
```

```
>>> {'Клавиатура': 12, 'Компьютерная мышь': 20, 'Системный блок':  
26, 'Монитор': 12}
```

Практическая часть.

Задание 1.

Дан словарь:

```
my_dict = {1:'0101101', 2:'101110', 3:'1A14C', 4:'1100100',  
5:'101010'}
```

1. Выведите его в консоль.
2. С помощью метода pop или del, уберите из словаря ключ с шестнадцатеричной системой счисления и выведите словарь в консоль.
3. Добавьте в словарь значение 0100101 с ключом 10 и выведите словарь в консоль.

Задание 2.

Дан словарь:

```
sl = {'</>':13, 'script':1, '__init__':10, 'self':5,  
'number_9':6, '#comment':100}
```

1. Выведете его в консоль.
2. Сделайте так, чтобы пользователь смог добавить в него 1 элемент через ввод в консоли.
3. Выведете словарь.

Например:

```
>>> {'</>': 13, 'script': 1, '__init__': 10, 'self': 5,  
'number_9': 6, '#comment': 100}  
  
>>> key: _prog # ввод ключа  
  
>>> value: 7 # ввод значения  
  
>>> {'</>': 13, 'script': 1, '__init__': 10, 'self': 5,  
'number_9': 6, '#comment': 100, '_prog': '7'}
```

Задание 3.

Написать программу, где пользователь может вводить значение в словарь, программа будет работать до тех пор, пока в словаре не будет трёх элементов. При этом, ключ обязательно должен иметь числовой тип данных.

Алгоритм.

1. Создание словаря.
2. Цикл. Пока длина словаря не будет равна трём.
3. Ввод ключа.
4. Ввод значения.
5. Добавление ключа и значения в словарь
6. Вывод словаря

Задание 4.

Дан словарь:

```
all_d = {1:15, 4:80, 44:0, 256:15, 100:70, 101:70, 20:44, 3:9}
```

Убрать из него элементы с ключами 1, 101, 3.

Критерии выставления оценки.

Оценка 5 – сделаны все 4 задания.

Оценка 4 – сделаны 3 задания.

Оценка 3 - сделаны 2 задания.

Оценка 2 – сделано одно или меньше заданий.