

Работа со словарями

Списки

Словарь (dictionary) в языке Python хранит коллекцию элементов, где каждый элемент имеет уникальный ключ и ассоциированное с ним некоторое значение.

```
dictionary = { ключ1:значение1, ключ2:значение2, ....}
```

Пример

Определим словарь

```
users = {1: "Tom", 2: "Bob", 3: "Bill"}  
emails = {"tom@gmail.com": "Tom", "bob@gmail.com":  
"Bob", "sam@gmail.com": "Sam"}
```

Пример

Но обязательно ключи и строки должны быть однотипными. Они могут представлять разные типы:

```
objects = {1: "Tom", "2": True, 3: 100.6}
```

Пустой словарь

Мы можем также вообще определить пустой словарь без элементов:

```
Objects = {} #вар 1
```

```
objects = dict() #вар 2
```

Обращение к элементам

Для обращения к элементам словаря после его названия в квадратных скобках указывается ключ элемента:

```
dictionary[ключ]
```

Пример

```
users = {  
    "+11111111": "Tom",  
    "+33333333": "Bob"  
}  
# получаем элемент с ключом "+11111111"  
print(users["+11111111"])    # Tom  
# установка значения элемента с ключом "+33333333"  
users["+33333333"] = "Bob Smith"  
print(users["+33333333"])    # Bob Smith
```

Возможные ошибки

Если элемента с указанным ключом нет, он будет добавлен:

```
users["+4444444"] = "Sam"
```

При попытке получить значение по несуществующему ключу `KeyError`

```
user = users["+4444444"] # KeyError
```


Метод get

`get(key)`: возвращает из словаря элемент с ключом `key`. Если элемента нет, возвращает значение `None`

`get(key, default)`: возвращает из словаря элемент с ключом `key`. Если элемента нет, то возвращает значение по умолчанию `default`

```
user3 = users.get("+44444444", "Unknown user")  
print(user3)  # Unknown user
```

Удаление

Для удаления элемента по ключу применяется оператор `del`:

```
users = {  
    "+11111111": "Tom",  
    "+55555555": "Alice"  
}  
  
del users["+1111111"]  
print(users)  # { "+55555555": "Alice"}
```

Проверка существования

```
users = {  
    "+11111111": "Tom",  
    "+55555555": "Alice"  
}  
key = "+55555555"  
if key in users:  
    del users[key]  
    print(f"Элемент с ключом {key} удален")  
else:  
    print("Элемент не найден")
```

Удаление

`pop(key)`: удаляет элемент по ключу `key` и возвращает удаленный элемент.

`pop(key, default)`: удаляет элемент по ключу `key` и возвращает удаленный элемент. Если элемент с данным ключом отсутствует, то возвращается значение `default`

Удаление

```
users = {  
    "+11111111": "Tom",  
    "+55555555": "Alice"  
}  
key = "+55555555"  
user = users.pop(key)  
print(user)    # Alice  
  
user = users.pop("+4444444", "Unknown user")  
print(user)    # Unknown user
```

Удаление

Если необходимо удалить все элементы, то в этом случае можно воспользоваться методом `clear()`:

```
users.clear()
```

Копирование словаря

Метод `copy()` копирует содержимое словаря, возвращая новый словарь:

```
users = {"+1111111": "Tom", "+3333333": "Bob", "+5555555":  
"Alice"}  
  
students = users.copy()  
  
print(students)      # {"+1111111": "Tom", "+3333333": "Bob",  
"+5555555": "Alice"}
```

Объединение словарей

Метод `update()` объединяет два словаря:

```
users = {"+1111111": "Tom", "+3333333": "Bob"}
users2 = {"+2222222": "Sam", "+6666666": "Kate"}
users.update(users2)

print(users)      # {"+1111111": "Tom", "+3333333": "Bob",
"+2222222": "Sam", "+6666666": "Kate"}
print(users2)     # {"+2222222": "Sam", "+6666666": "Kate"}
```


Перебор словаря

Для перебора словаря можно воспользоваться циклом `for`:

```
users = {  
    "+11111111": "Tom",  
    "+33333333": "Bob",  
    "+55555555": "Alice"  
}  
for key in users:  
    print(f"Phone: {key} User: {users[key]} ")
```

Задание

- Реализовать приложение «Телефонная книга»
- Добавить возможность добавлять новых абонентов, удалять, просматривать телефонную книгу
- Использовать словари
- Номер телефона — ключ, имя - значение