

# Словари. Множества



Гежин Олег



**Олег Гежин**

Python - разработчик, специалист SQL, фрилансер.



# План занятия

1. [Словари](#)
2. [Операции над словарями](#)
3. [Множества](#)
4. [Операции над множествами](#)



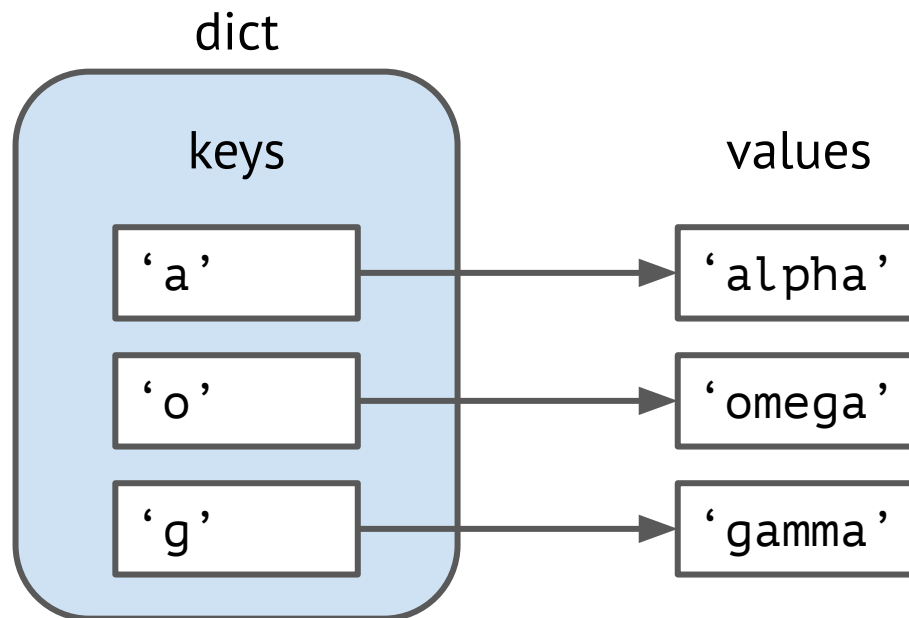
# Словари

# Словари

Словарь - это тип данных, который представляет собой упорядоченную (начиная с версии python 3.7) коллекцию, каждому элементу которой соответствует свой уникальный ключ. То есть весь словарь состоит из пар **ключ-значение**, каждому значению присвоен свой уникальный ключ, по которому мы можем извлечь это значение из словаря.

Элементы в словарях хранятся в формате *key:value*.

```
dict = {'a': 'alpha', 'o': 'omega', 'g': 'gamma'}
```



# Словари

Ключами могут быть любые неизменяемые типы данных: *strings*, *booleans*, *integers*, *floats*, *tuple*.

```
key      : value
telephones_dict = {88009991122: 'Оля', 89008882211: 'Коля'}
capitals_dict = {'Russia': 'Moscow', 'Italy': 'Rome'}
```

Все ключи в словаре должны быть уникальными.

# Словари

Получение значения из словаря по ключу:

```
capitals_dict = {'Russia': 'Moscow', 'Italy': 'Rome'}  
print(capitals_dict['Russia'])
```

Итерация по словарю:

```
for country, capital in capitals_dict.items():  
    print(country, '->', capital)
```

Добавление нового элемента в словарь:

```
capitals_dict['France'] = 'Paris'
```



# Операции над словарями



# Операции над словарями

- `del (dict[key])` удаляет элемент из списка по ключу;
- `keys()` позволяет получить все ключи словаря;
- `values()` позволяет получить все значения словаря;
- `items()` позволяет получить ключи и значения словаря;
- `get(key)` “безопасно” возвращает значение по ключу (при отсутствии ключа ошибка не возникает);
- `setdefault(key, default)` позволяет получить значение по ключу, автоматически добавляет элемент в словарь, если его нет.



# Множества



# Множества

Множества (sets) — «контейнер», содержащий неповторяющиеся элементы в случайном порядке.

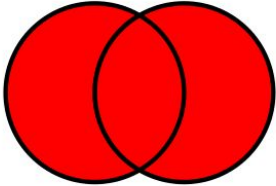
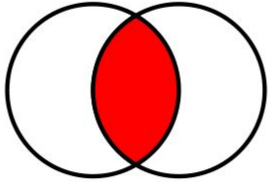
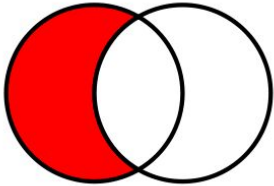
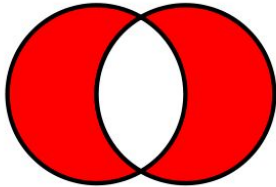
Множество инициализируется при помощи **set()**. Обычно создается из списков.

Реализуют теорию множеств в Python ([см. круги Эйлера](#)).



# Операции над множествами

# Операции над множествами

Объединение(union) Логическое “ИЛИ” (or)		<pre>set_a = {1, 2, 3} set_b = {3, 4, 5} set_c = set_a   set_b # {1, 2, 3, 4, 5}</pre>
Пересечение(intersection) Логическое “И” (and)		<pre>set_c = set_a &amp; set_b # {3}</pre>
Разность (difference)		<pre>set_c = set_a - set_b # {1, 2}</pre>
Симметричная разность (symmetric difference)		<pre>set_c = set_a ^ set_b # {1, 2, 4, 5}</pre>

# Операции над множествами

- **add(el)** добавляет элемент в множество;
- **update(set)** соединяет множество с другим множеством/списком;
- **discard(el)** удаляет элемент из множества по его значению;
- **union(set)** объединяет множества (логическое “ИЛИ”);
- **intersection(set)** — пересечение множеств (логическое “И”);
- **difference(set)** возвращает элементы одного множества, которые не принадлежат другому множеству (разность множеств);
- **symmetric\_difference(set)** возвращает элементы, которые встречаются в одном множестве, но не встречаются в обоих.

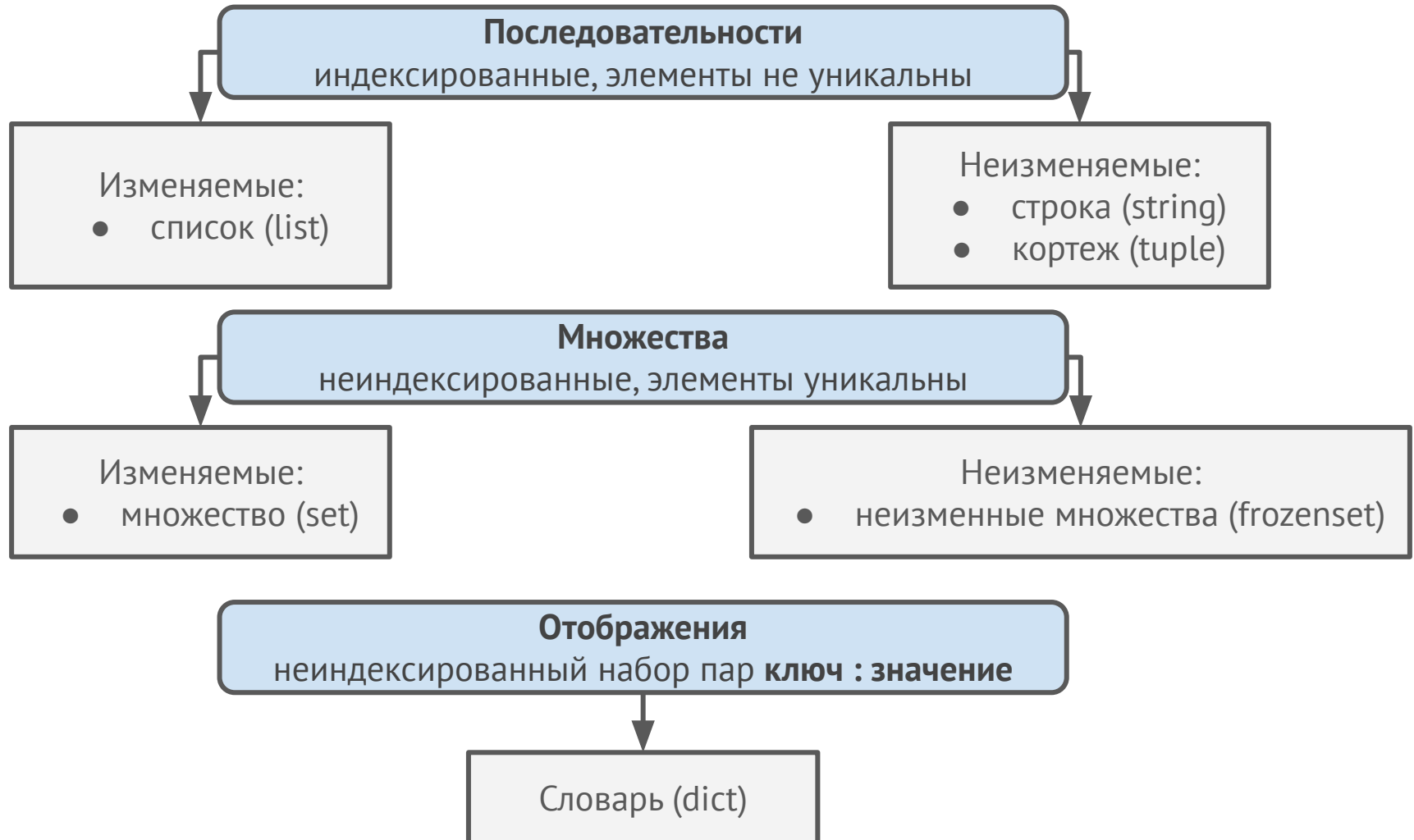


# Статические множества

Статические множества (*frozenset*) — неизменяемое множество.

Над *frozenset* доступны все операции, применяемые к множествам, за исключением тех, которые их изменяют.

# Классификация коллекций





---

## Домашнее задание

Перед выполнением задания прочитайте короткую статью про [типы данных](#) и [отличную статью на Хабре](#)

## Задание 1

Дан список с визитами по городам и странам. Напишите код, который возвращает отфильтрованный список `geo_logs`, содержащий только визиты из России."

```
geo_logs = [  
    {'visit1': ['Москва', 'Россия']},  
    {'visit2': ['Дели', 'Индия']},  
    {'visit3': ['Владимир', 'Россия']},  
    {'visit4': ['Лиссабон', 'Португалия']},  
    {'visit5': ['Париж', 'Франция']},  
    {'visit6': ['Лиссабон', 'Португалия']},  
    {'visit7': ['Тула', 'Россия']},  
    {'visit8': ['Тула', 'Россия']},  
    {'visit9': ['Курск', 'Россия']},  
    {'visit10': ['Архангельск', 'Россия']}  
]
```

## Задание 2

Выведите на экран все уникальные гео-ID из значений словаря ids.

Т.е. список вида [213, 15, 54, 119, 98, 35]

```
ids = {'user1': [213, 213, 213, 15, 213],  
       'user2': [54, 54, 119, 119, 119],  
       'user3': [213, 98, 98, 35]}
```

## Задание 3

Дан список поисковых запросов. Получить распределение количества слов в них. Т.е. поисковых запросов из одного - слова 5%, из двух - 7%, из трех - 3% и т.д.

```
queries = [  
    'смотреть сериалы онлайн',  
    'новости спорта',  
    'афиша кино',  
    'курс доллара',  
    'сериалы этим летом',  
    'курс по питону',  
    'сериалы про спорт'  
]
```

## Задание 4

Дана статистика рекламных каналов по объемам продаж.

Напишите скрипт, который возвращает название канала с максимальным объемом.

Т.е. в данном примере скрипт должен возвращать 'yandex'.

```
...
```

```
stats = {'yandex': 120, 'vk': 115, 'google': 99, 'email': 42, 'ok': 98}
```

```
...
```

# Задание 5(Необязательное)

Напишите код для преобразования произвольного списка вида `['2018-01-01', 'yandex', 'cpc', 100]` (он может быть любой длины) в словарь `{'2018-01-01': {'yandex': {'cpc': 100}}}`

Для подготовки к следующей лекции прочитайте про [функции](#)

## Инструкция по выполнению домашнего задания:

1. Зарегистрируйтесь на сайте [Repl.IT](#).
2. Перейдите в раздел **my repls**.
3. Нажмите кнопку **Start coding now!**, если приступаете впервые, или **New Repl**, если у вас уже есть работы.
4. В списке языков выберите Python.
5. Код пишите в левой части окна.
6. Посмотреть результат выполнения файла можно, нажав на кнопку **Run**. Результат появится в правой части окна.
7. После окончания работы нажмите кнопку **Share** и скопируйте ссылку из поля *Share link*.
8. В личном кабинете на сайте [netology.ru](#) в поле комментария к домашней работе вставьте скопированную ссылку и отправьте работу на проверку.

*Помимо ссылки, пожалуйста, прикрепите к вашему решению любой файл (например, скриншот любой части вашего кода).*

⌘ нетология

**Задавайте вопросы и  
пишите отзыв о лекции!**

Олег Гежин