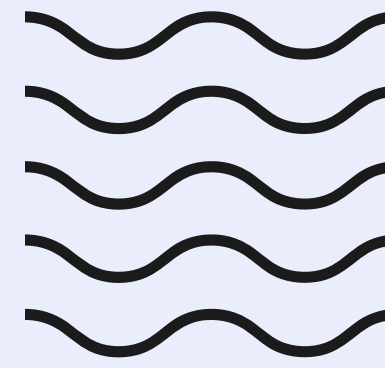


HashSet



ПОДГОТОВИЛ: АЛИШЕР ХАМИДОВ



Основные моменты:

План лекции

1. Как создать?
2. Методы
3. Set vs List
4. Важные замечания



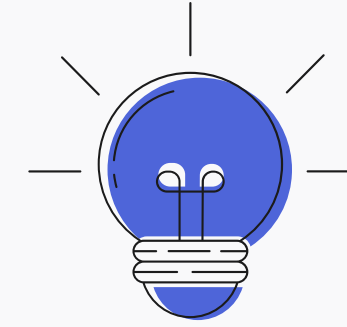


Что такое Set?

Ваши ассоциации

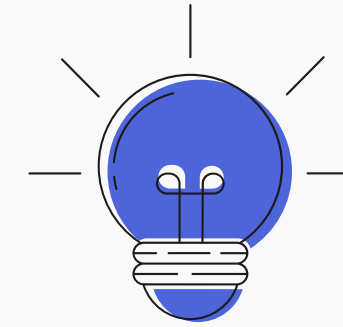


Как создать?



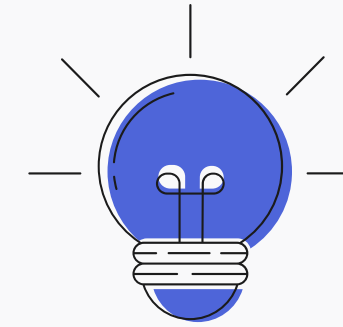
```
Set<String> fruits = new HashSet<>();
```

Как создать?



```
Set<String> fruits = new HashSet<>();  
fruits.add("apple");  
fruits.add("pineapple");
```

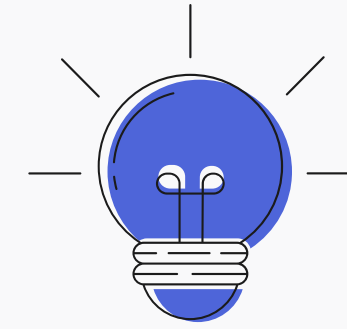
Как создать?



```
Set<String> fruits = new HashSet<>(previousFruits);
```

// можно создать взяв элементы из ArrayList или другой реализации коллекции

Как добавить?



```
Set<String> fruits = new HashSet<>();
```

```
fruits.add("apple");
```

```
fruits.add("pineapple");
```

```
System.out.println(fruits); // [apple, pineapple]
```

Как добавить?



```
Set<String> fruits = new HashSet<>();
```

```
fruits.add("apple");
```

```
fruits.add("pineapple");
```

```
fruits.add("apple"); // false
```

```
// повторяющийся элемент не будет добавлен
```

```
System.out.println(fruits); // [apple, pineapple]
```


Как очистить от всех элементов?

```
Set<String> fruits = new HashSet<>();
```

```
fruits.add("apple");
```

```
fruits.add("pineapple");
```

```
fruits.clear();
```

```
System.out.println(fruits); // []
```



Как проверить на пустоту?

```
Set<String> fruits = new HashSet<>();
```

```
fruits.add("apple");
```

```
fruits.add("pineapple");
```

```
fruits.clear();
```

```
System.out.println(fruits.isEmpty()); // true
```



Как проверить содержится ли элемент?

```
Set<String> fruits = new HashSet<>();
```

```
fruits.add("apple");
```

```
fruits.add("apple");
```

```
System.out.println(fruits.contains("orange")); // false
```



Как пройти циклом?

```
Set<String> fruits = new HashSet<>();  
fruits.add("apple");  
fruits.add("orange");  
for (String i : fruits) {  
    System.out.println(i);  
}
```



HashSet

- нет повторяющихся элементов
- не сохраняется порядок добавления
- нельзя получить элемент по индексу

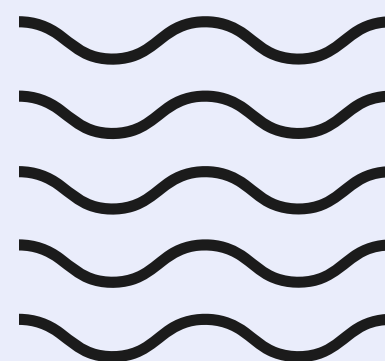
VS

List

- могут быть повторяющиеся элементы
- сохраняется порядок добавления
- можно получить элемент по индексу



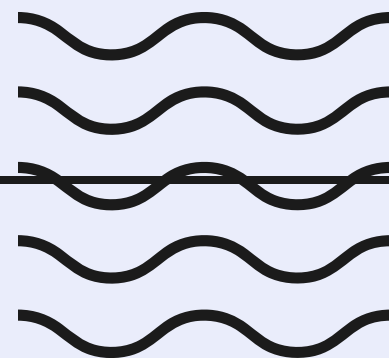
HashSet основан на хэш-таблице, и операции добавления, удаления или поиска в среднем будут выполняться за константное ($O(1)$) время.



Важное замечание



Если мы создаем хеш сет, в котором будем хранить объекты класса, следует обязательно переопределить (override) в этом классе методы **equals** и **hashCode** (их можно просто сгенерировать).



Документация Oracle

<https://docs.oracle.com/javase/7/docs/api/java/util/HashSet.html>

