Java Professional module #2

lecture #5. Java Set implementation. Exercises.

Mentor: Rustam Khakov

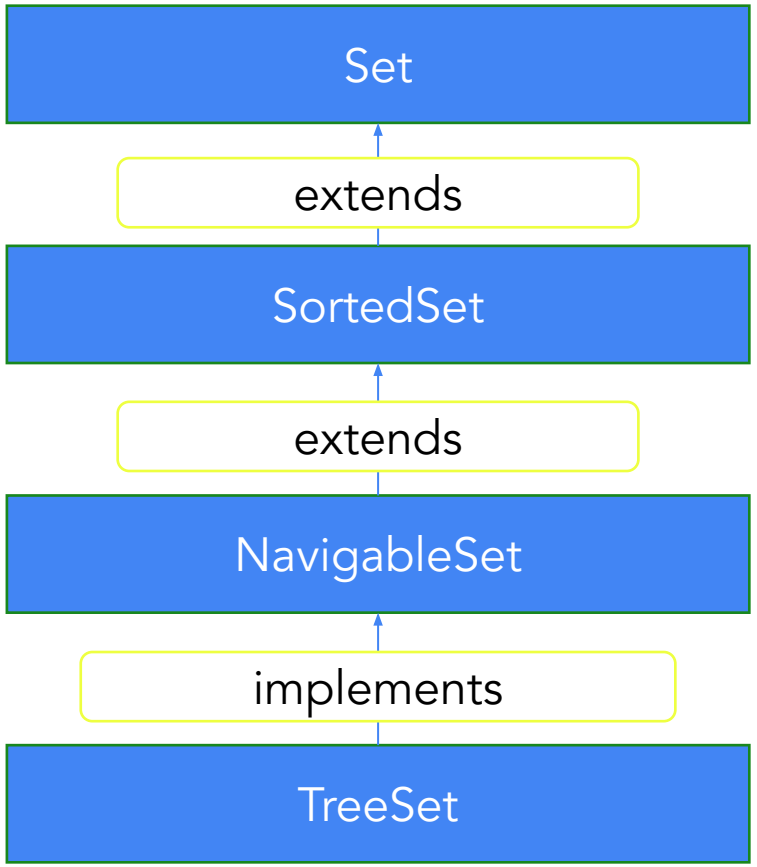
Lecture #5. Java Set implementation. Exercises.

- Set in Java
- Set Objects
- Operations on the Set Interface
- HashSet basic operations
- Classes that implement the Set interface
- Array to Set

Set

- Нет дубликатов
- Порядок вставки не важен
- Нет доступа по индексу

Set


Set
extends
SortedSet
extends
NavigableSet
implements
TreeSet

SortedSet

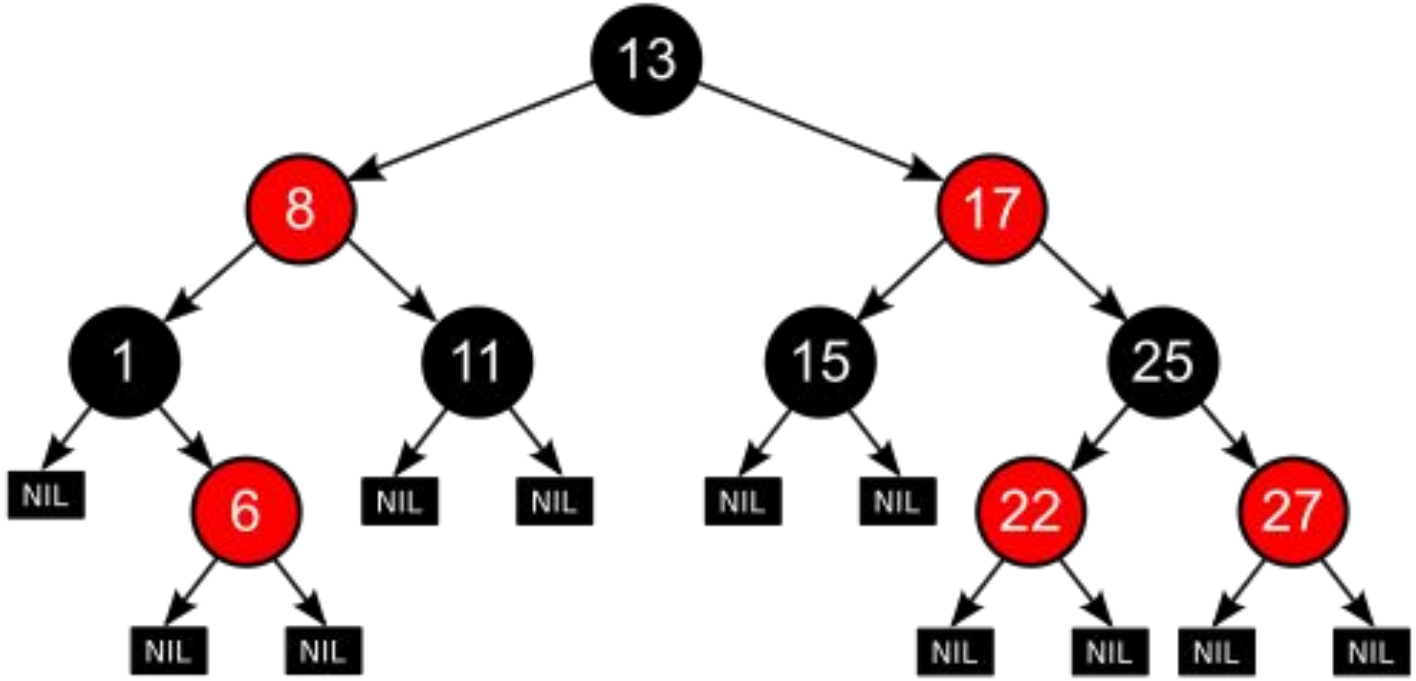
- Расширяет интерфейс Set
- Отсортирован
- Все элементы SortedSet должны реализовывать интерфейс Comparable или быть приняты Comparator по умолчанию (подробнее поговорим на следующей лекции)
- SortedSet — это интерфейс
- TreeSet — это класс, реализующий интерфейс SortedSet

NavigableSet

- Наследуется от интерфейса SortedSet.
- Есть методы навигации (обратный порядок)
- Примеры, реализующие этот интерфейс: TreeSet и ConcurrentSkipListSet.

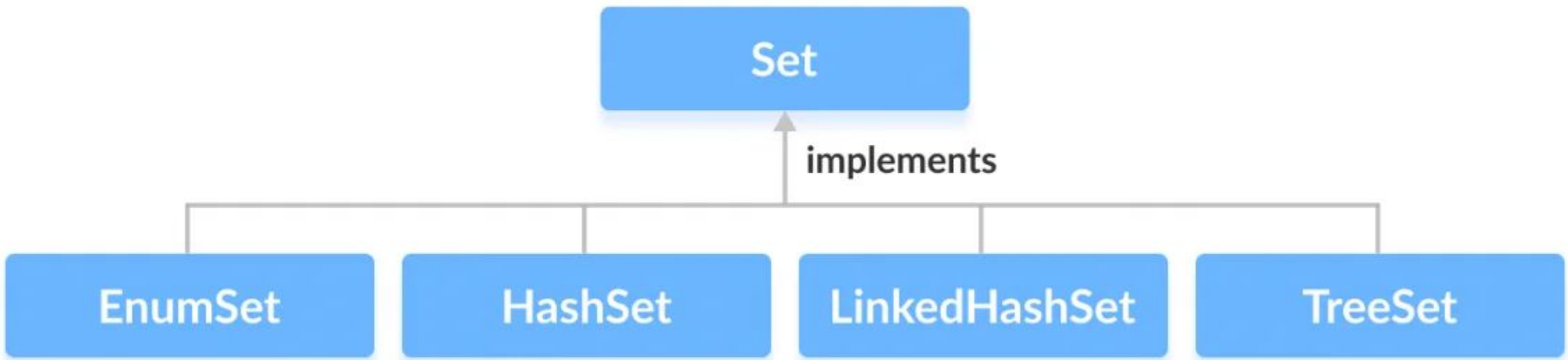
TreeSet

- Бинарное дерево поиска
- Большой объем отсортированной информации с быстрым доступом.
- Вставка null значений вызывает исключение NullPointerException
- Не потокобезопасный



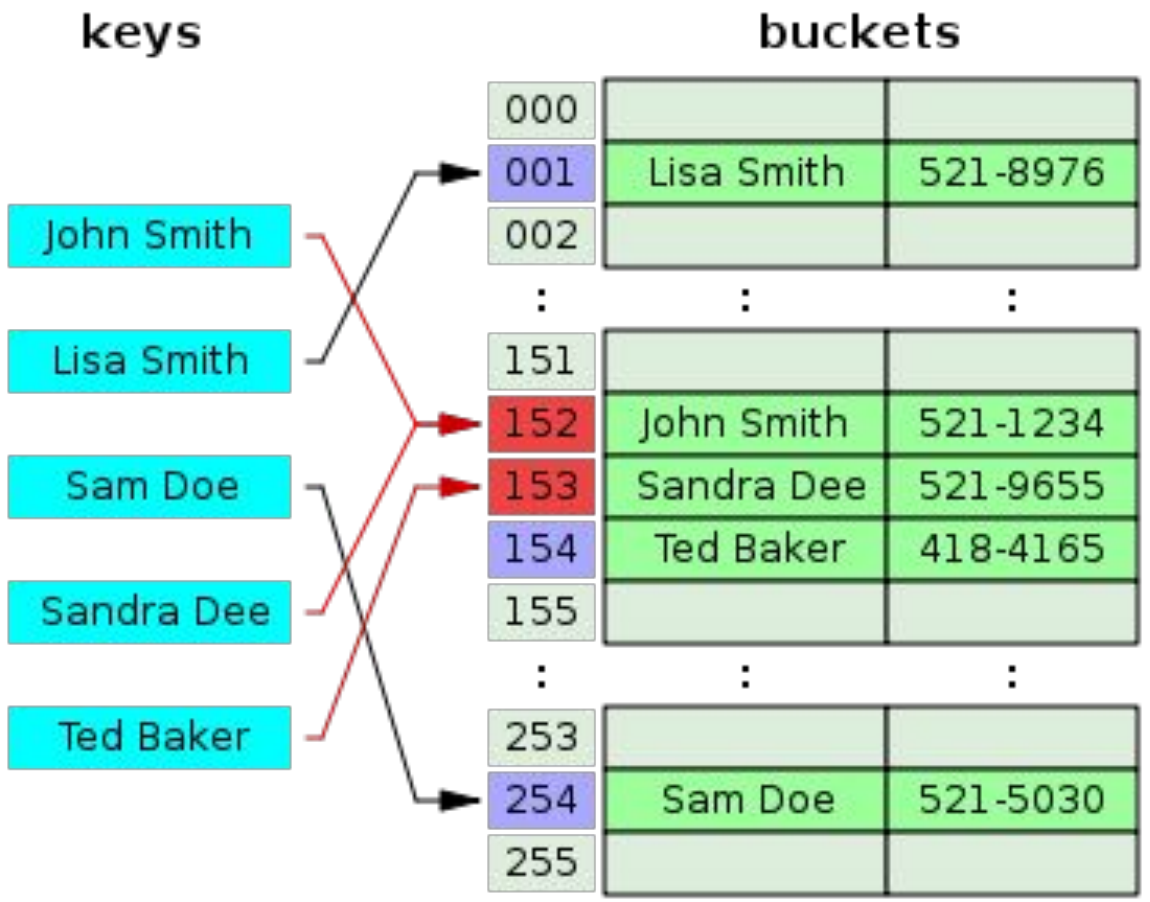
ConcurrentSkipListSet

- Параллельная версия NavigableSet.
- Естественный порядок
- Потокобезопасен


Set
implements
EnumSet
HashSet
LinkedHashSet
TreeSet

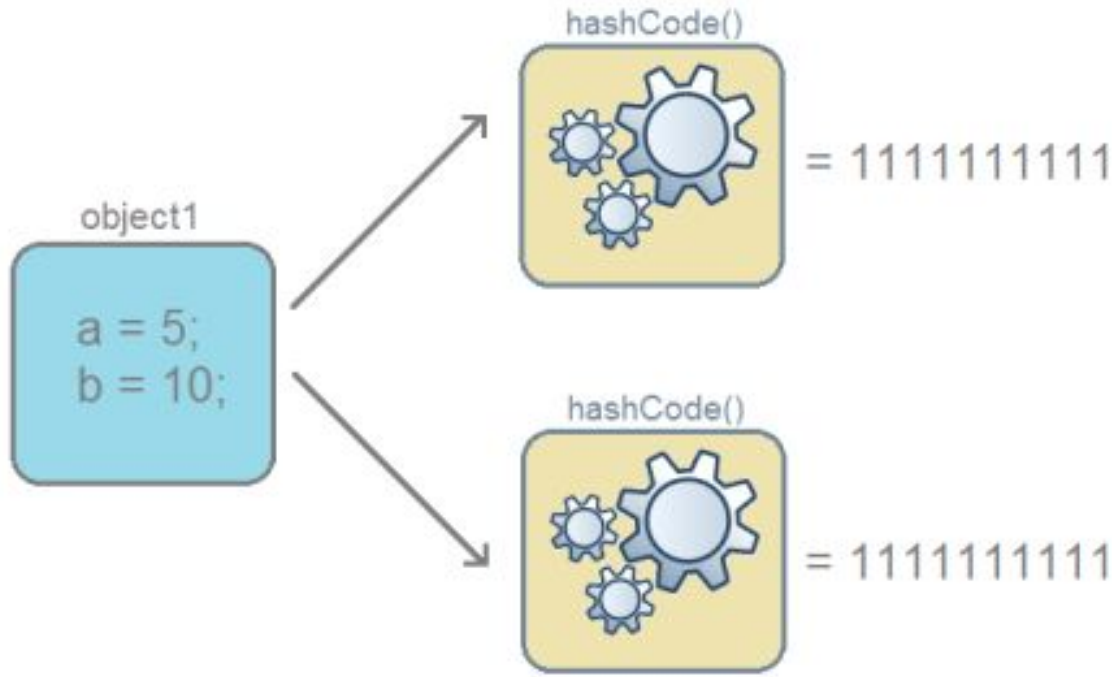
HashSet

- Интерфейс Set
- Не гарантируется порядок
- Разрешает null элемент
- HashSet – O(1)
- Хэш-код
- Подкласс HashSet — LinkedHashSet



HashSet

1. boolean add(E e): добавляет элемент в HashSet, если таковой отсутствует, если же такой элемент уже присутствует, метод возвращает false.
2. void clear(): удаляет все элементы.
3. boolean contains(Object o): Возвращает true, если данный элемент присутствует.
4. boolean remove(Object o): удаляет данный элемент, если таковой присутствует.
5. Iterator iterator(): возвращает итератор для элементов.
6. boolean isEmpty(): возвращает true, если нет элементов.
7. Object clone(): выполняет поверхностное клонирование HashSet.



1111111111 = 1111111111

	HashSet	TreeSet
Порядок	нет	сортирует
Дубликаты	нет дубликатов	нет дубликатов
Структура под капотом	хэш таблица	красно-чёрное дерево
null	null возможен	нельзя
Реализация	под капотом HashMap	под капотом TreeMap
Производительность	быстрее	медленнее

	Временная сложность							
	Среднее				*Худшее*			
	Индекс	Поиск	Вставка	Удаление	Индекс	Поиск	Вставка	Удаление
ArrayList	O(1)	O(n)	O(n)	O(n)	O(1)	O(n)	O(n)	O(n)
Vector	O(1)	O(n)	O(n)	O(n)	O(1)	O(n)	O(n)	O(n)
LinkedList	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)
Hashtable	n/a	O(1)	O(1)	O(1)	n/a	O(n)	O(n)	O(n)
HashMap	n/a	O(1)	O(1)	O(1)	n/a	O(n)	O(n)	O(n)
LinkedHashMap	n/a	O(1)	O(1)	O(1)	n/a	O(n)	O(n)	O(n)
TreeMap	n/a	O(log(n))	O(log(n))	O(log(n))	n/a	O(log(n))	O(log(n))	O(log(n))
HashSet	n/a	O(1)	O(1)	O(1)	n/a	O(n)	O(n)	O(n)
LinkedHashSet	n/a	O(1)	O(1)	O(1)	n/a	O(n)	O(n)	O(n)
TreeSet	n/a	O(log(n))	O(log(n))	O(log(n))	n/a	O(log(n))	O(log(n))	O(log(n))

Array to Set

- Наивный метод
- Arrays.asList()
- Collections.addAll()
- Java 8 Stream API
- Set.of()