

# Java Advanced

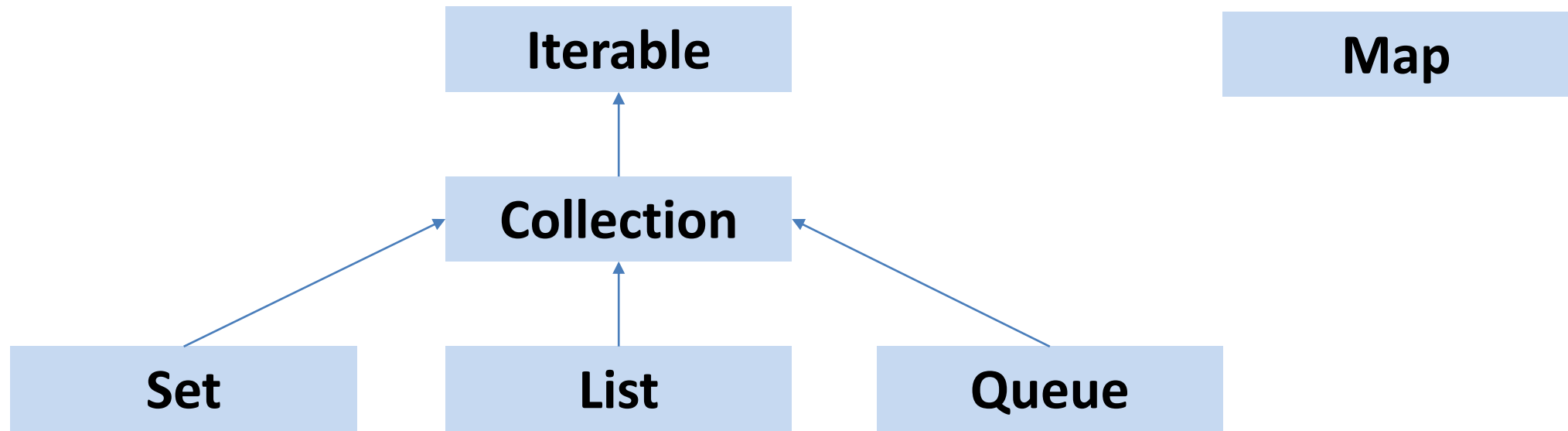
Java Collection



## Java Collection

# Структура Java Collections Framework

## Java Collections Framework Structure



Интерфейс

# Компаратор

## Comparator

### Comparator

#### **compare (Object o1, Object o2)**

Упорядочивает объекты класса.  
Возвращает -1, 0, 1.

#### **equals (Object o)**

Сравнивает компараторы  
объектов. Возвращает true или  
false.

### Comparable

#### **compareTo (Object o)**

Предназначен для реализации  
упорядочивания объектов  
класса.  
Возвращает -1, 0, 1.

# Итератор

Iterator

**Iterator**

**hasNext ()**

Проверяет наличие следующего элемента

**next ()**

Возвращает следующий элемент

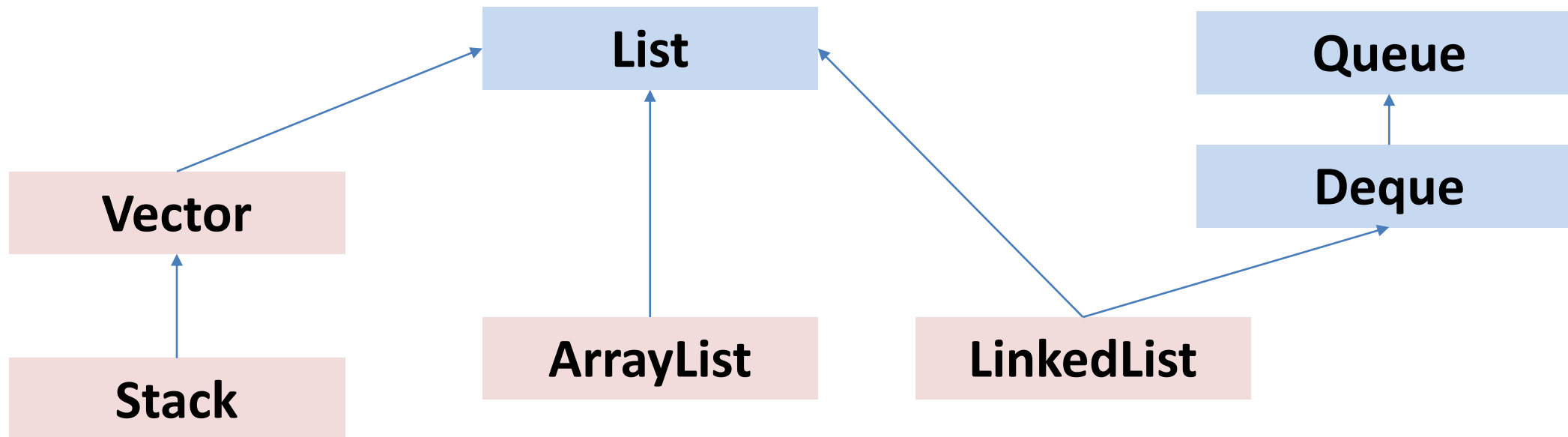
**Iterable**

**Iterator<T> iterator()**

Возвращает объект класса Iterator.

# Список

List



Интерфейс



Класс

# Реализация интерфейса List

## List implementation

**Vector** – реализация динамического массива объектов.

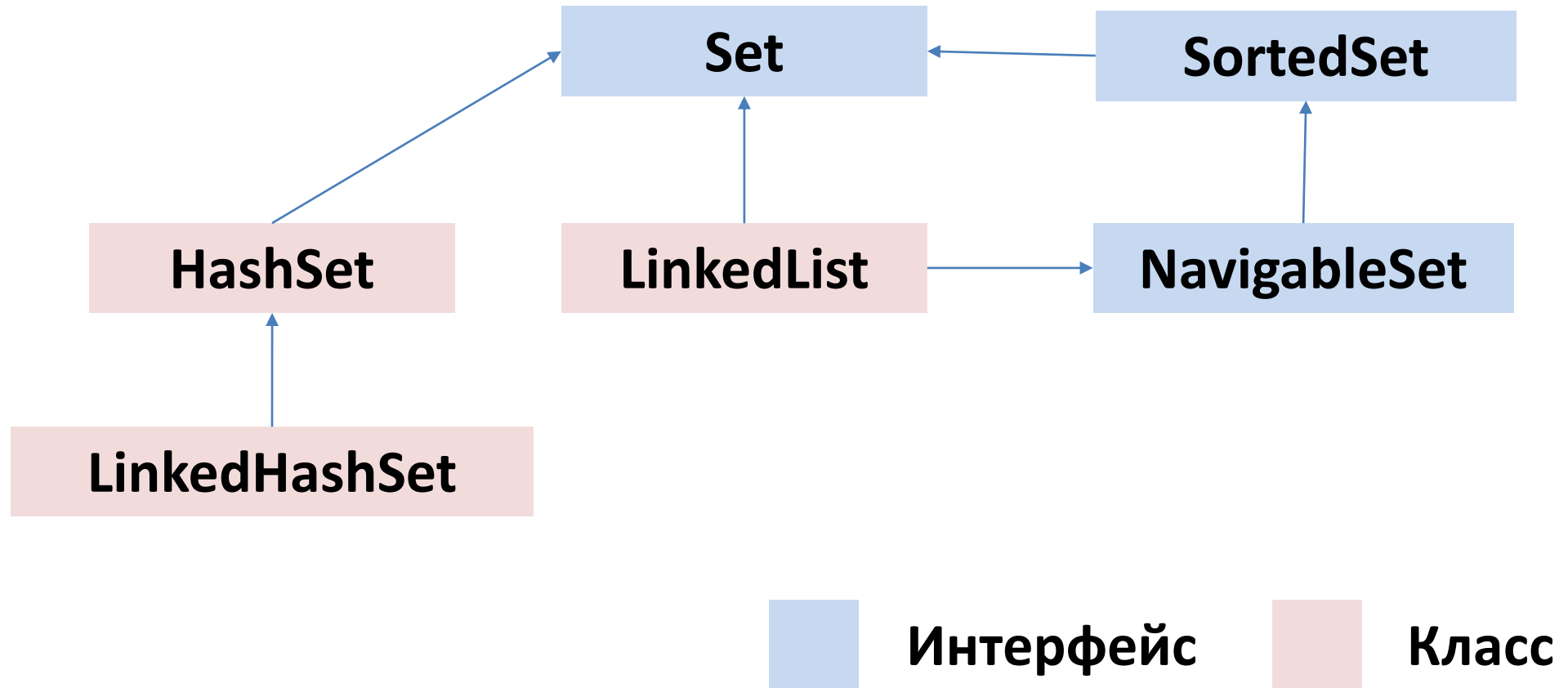
**Stack** – реализация стека LIFO.

**Arraylist** – реализация динамического массива объектов в виде односвязного списка.

**LinkedList** – реализация динамического массива объектов в виде двусвязного списка.

# Множество

Set





# Реализация интерфейса Set

## Set implementation

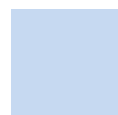
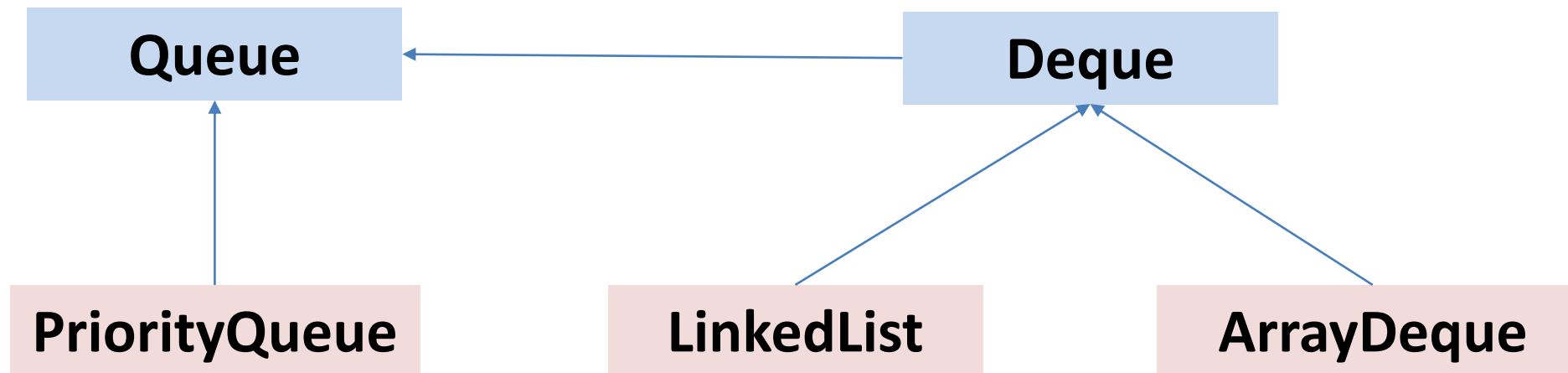
**HashSet** – реализация интерфейса Set, базирующаяся на HashMap.

**LinkedHashSet** – реализация интерфейса Set, базирующаяся на LinkedHashMap.

**TreeSet** – реализация интерфейса Set, базирующаяся на NavigableMap.

# Очередь

Queue



Интерфейс



Класс

# Реализация интерфейса Queue

## Queue implementation

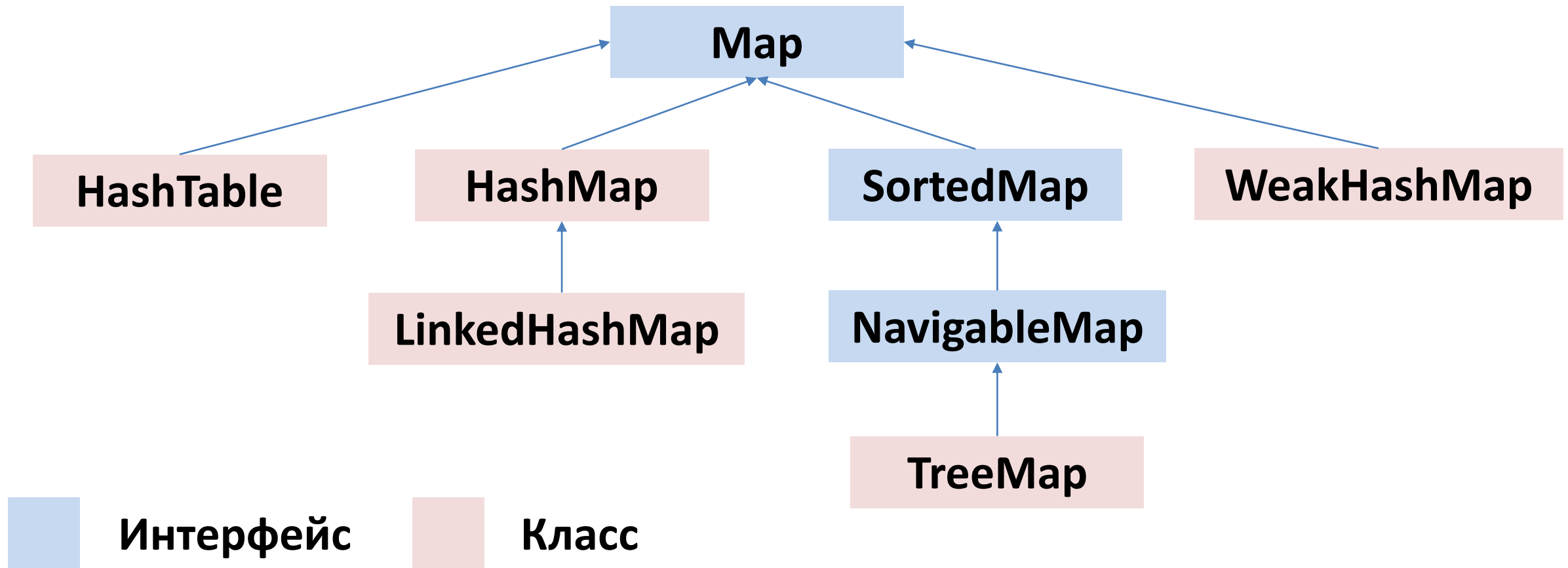
**PriorityQueue** – единственная прямая реализация интерфейса Queue, реализовывает конструкцию LIFO.

**Deque** – интерфейс для реализации двусторонней очереди.

**ArrayDeque** – реализация интерфейса Deque, позволяющая реализовать конструкцию FIFO.

# Карта

Map



# Реализация интерфейса Map

## Map implementation

**HashTable** – реализация такой структуры данных, как хэш-таблица.

**HashMap** – коллекция, альтернативная HashTable.

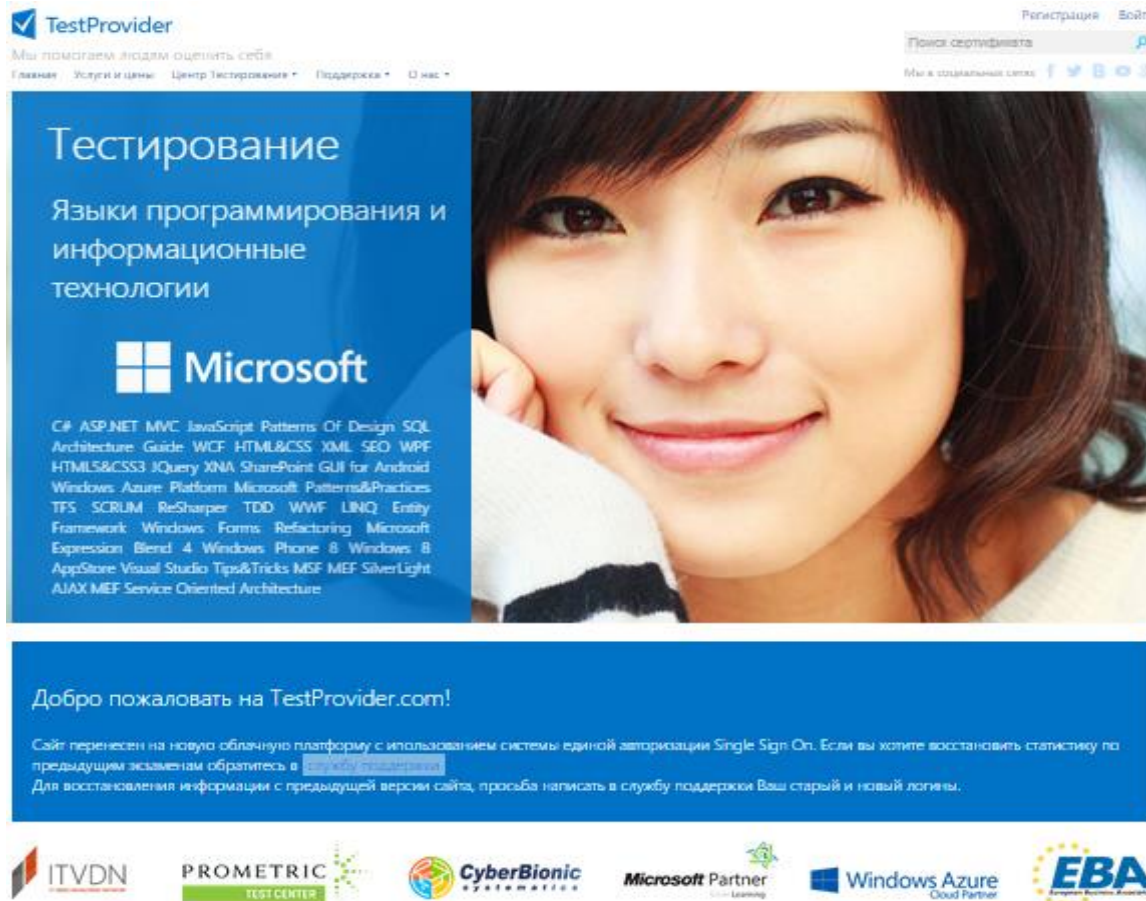
**LinkedHashMap** – упорядоченная реализация хэш-таблицы.

**TreeMap** – реализация Map, основанная на красно-чёрных деревьях.

**WeakHashMap** – реализация хэш-таблицы, с использованием *weak references*.

# Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



Q&A