

Интерфейсы коллекций

Java Collections
6 уровень, 3 лекция

ОТКРЫТА

— Привет, Амиго!

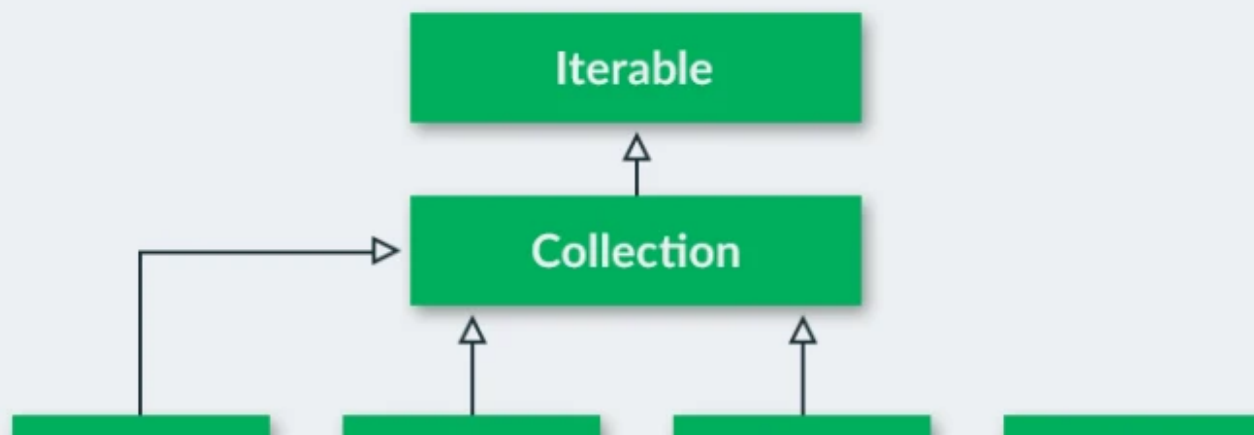
— Привет, Элли!

— Сегодня мы разберемся с устройством коллекций раз и навсегда.

— Давно этого ждал.

— Ты уже знаешь, что такое коллекции, и знаешь, как с ними работать. Пришло время выстроить твои знания в строгую систему. Тогда отпадут многие вопросы — почему, как, зачем — и большинство вещей станет очевидно.

Интерфейсы коллекций. Структура наследования **интерфейсов** коллекций выглядит примерно так:



НАЧАТЬ ОБУЧЕНИЕ

Обрати внимание, на две вещи.

Во-первых, **все, что ты тут видишь – это интерфейсы.**

Во-вторых, **стрелочки обозначают «наследуется от».**

— Т.е. `List`, `Set`, `Queue` наследуются от `Collection`, а `Map` – нет?

— Ага. Потом от этих интерфейсов наследуются абстрактные классы, а от них в свою очередь – известные тебе реализации: **`ArrayList`**, **`Hashtable`**, **`TreeSet`**,...

— Есть такое дело.

Теперь давай посмотрим, что за методы есть у этих интерфейсов:

Методы интерфейса **`Iterable<E>`**:

Методы	Описание
<code>Iterator<T> iterator();</code>	Возвращает объект-итератор.

— Маловато как-то.

— Ну, сколько есть. Все ради итераторов, прямо сейчас мы их рассматривать не будем, но скоро уже рассмотрим во всех подробностях.

Методы интерфейса **`Collection<E>`**:

Методы	Описание
<code>boolean add(E e);</code>	Добавляет элемент в коллекцию
<code>boolean addAll(Collection<? extends E> c);</code>	Добавляет элементы в коллекцию
<code>void clear();</code>	Удаляет все элементы из коллекции
<code>boolean contains(Object o);</code>	Проверяет – есть ли в коллекции элемент?
<code>boolean containsAll(Collection<?> c);</code>	Проверяет – есть ли в коллекции элементы?

НАЧАТЬ ОБУЧЕНИЕ

<code>int hashCode();</code>	Возвращает хэш-код
<code>boolean isEmpty();</code>	Проверяет – пуста ли коллекция?
<code>Iterator<E> iterator();</code>	Возвращает объект-итератор
<code>boolean remove(Object o);</code>	Удаляет элемент из коллекции
<code>boolean removeAll(Collection<?> c);</code>	Удаляет элементы из коллекции
<code>boolean retainAll(Collection<?> c);</code>	Удаляет все элементы, которых нет «с»
<code>int size();</code>	Возвращает размер коллекции
<code>Object[] toArray();</code>	Преобразовывает коллекцию к массиву
<code><T> T[] toArray(T[] a);</code>	Преобразовывает коллекцию к массиву

— Тут уже посolidнее все. Половину из этих методов я использовал, а со второй сталкивался.

— Отлично, тогда продолжим.

Методы интерфейса **List<E>**:

Методы	Описание
<code>void add(int index, E element);</code>	Добавляет элементы в середину коллекции
<code>boolean addAll(int index, Collection<? extends E> c);</code>	Добавляет элементы в коллекцию
<code>E get(int index);</code>	Возвращает элемент по номеру
<code>int indexOf(Object o);</code>	Возвращает индекс(номер) элемента
<code>int lastIndexOf(Object o);</code>	Возвращает последний индекс

НАЧАТЬ ОБУЧЕНИЕ

<code>ListIterator<E> listIterator();</code>	Возвращает итератор для списка
<code>ListIterator<E> listIterator(int index);</code>	Возвращает итератор для списка
<code>E remove(int index);</code>	Удаляет элемент по индексу
<code>E set(int index, E element);</code>	Устанавливает новое значение по индексу
<code>List<E> subList(int fromIndex, int toIndex);</code>	Возвращает подколлекцию

— Тоже ничего кардинально нового. Я уже практически все знаю по коллекциям, что не может не радовать.

— Ну, я думаю, у меня найдется, чем тебя удивить. Но давай продолжим изучать интерфейсы:

Методы интерфейса **Set<E>**:

Методы	Описание
нет методов	

Интерфейс Set не содержит новых методов, только унаследованные.

— Да, я смотрю, интерфейс Iterable был еще ничего.

С другой стороны – меньше методов – меньше запоминать!

— Меня радует твой жизнеутверждающий оптимизм.

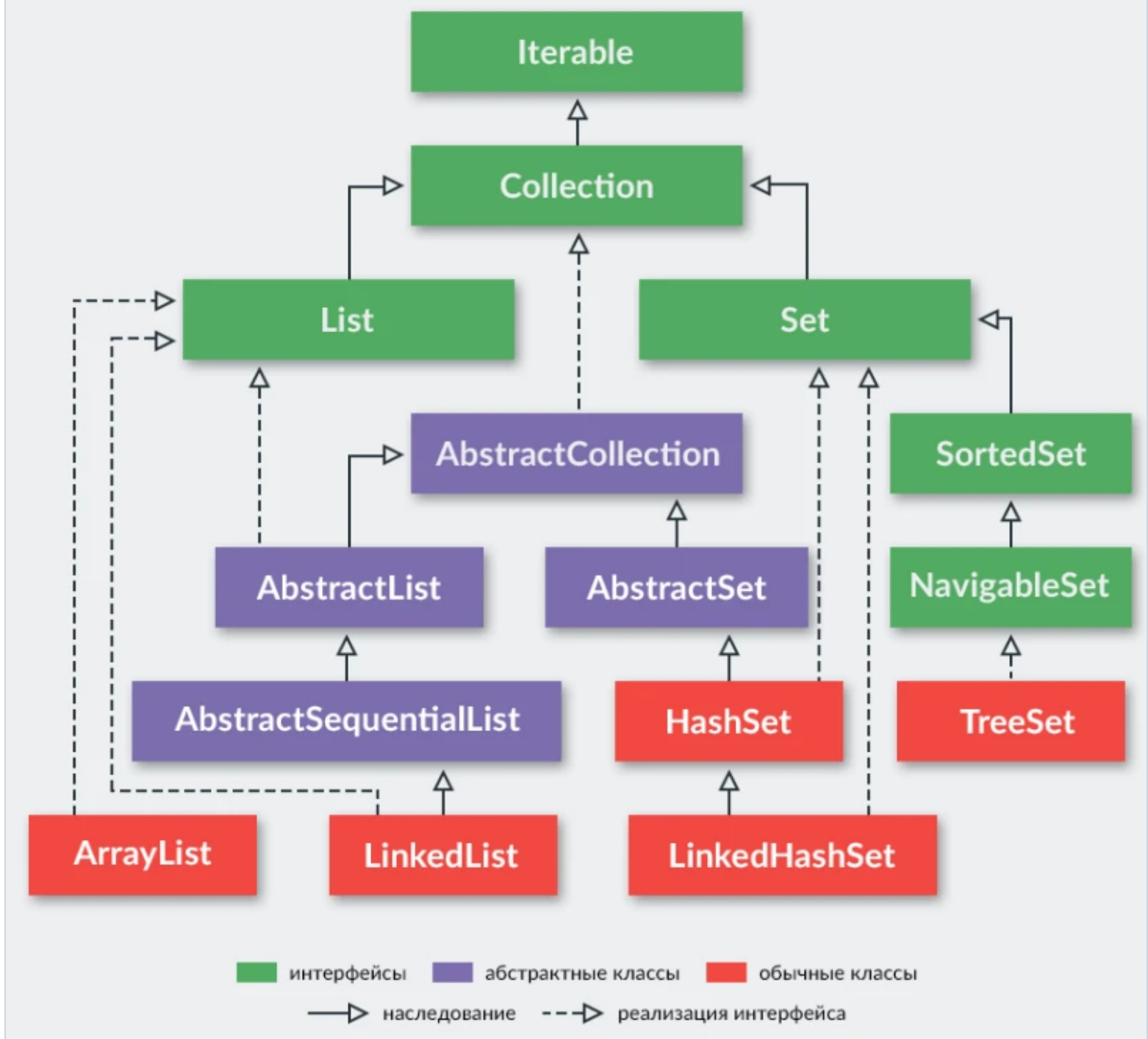
У интерфейса Set есть два интерфейса-наследника с методами: SortedSet и NavigableSet, но я не буду их приводить, а то мы никогда не закончим.

Давай я лучше дам тебе общую картину классов и интерфейсов, описывающих коллекции в Java.

— Давай.

— Тогда держи:

НАЧАТЬ ОБУЧЕНИЕ



— Ничего себе, да она просто огромная!

— Ну, не такая уж она и большая. Кроме того, запоминать абстрактные классы совсем не обязательно. А вот какой класс, какие интерфейсы реализует лучше запомнить – такое может пригодиться довольно часто.

— Ну, еще хотелось бы отметить, что некоторые коллекции были признаны устаревшими.

— Это какие же?

[НАЧАТЬ ОБУЧЕНИЕ](#)

Но в Java появилась специальная библиотека — **concurrency**, где содержится очень много коллекций, к которым не только можно обращаться из других потоков/нитей, но и которые написаны гораздо эффективнее. **ConcurrentHashMap** гораздо эффективнее **Hashtable**.

Использовать коллекции Vector, Stack, Dictionary, Hashtable можно, но не рекомендуется.

— Ясно, буду иметь в виду.

Спасибо, Элли!

< Предыдущая

 ×26

 +70 

Комментарии (35)

популярные новые старые

JavaCoder

Введите текст комментария

НАЧАТЬ ОБУЧЕНИЕ

Первое что бросается в глаза на картинке - нет стрелочки от ArrayList с наследованием AbstractList

Ответить

0

Макс Дудин Уровень 39, Калининград, Россия

4 мая, 11:38

...

вот так понятней...

`boolean retainAll(Collection<?> c)`

Сохраняет в этой коллекции только те элементы, которые содержатся в указанной коллекции (дополнительная операция). Другими словами, удаляет из этой коллекции все ее элементы, не содержащиеся в указанной коллекции.

Ответить

+1

Aleksey Уровень 51, Пермь, Россия

16 ноября 2021, 12:11

...

Ну вообще-то TreeSet **наследуется** от AbstractSet

<https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/TreeSet.html>

Ответить

+1

Ян Уровень 41, Лида, Беларусь

22 ноября 2021, 23:28

...

точно так же как и ArrayList<E> наследуется от AbstractList<E>:

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

Ответить

0

Stepan Уровень 27, Москва, Россия

20 марта, 21:50

...

и AbstractSet реализует Set

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

Ответить

0

Aleksey Уровень 51, Пермь, Россия

21 марта, 10:08

...

если разобраться, то схема будет совсем другая

Ответить

0

Valua Sinicyn Уровень 41, Харьков, Украина

16 февраля 2021, 10:14

...

Спасибо за инфо-графику, забрал.

Ответить

0

Vladyslav Pelykh Уровень 41, Днепр, Украина

25 октября 2021, 21:14

...

++

В Шилдте все описывалось, ко именно инфо-графики не хватало...

Ответить

0

Vorlock Уровень 31, Днепр, Украина

29 января 2020, 20:50

...

НАЧАТЬ ОБУЧЕНИЕ



Всё ещё исправляют.

Ответить

0

Vorlock Уровень 31, Днепр, Украина

29 января 2020, 20:46



не, похоже дальтоник кто фронт писал\верстал

Ответить

+2

wrcraid Уровень 35, Россия

13 декабря 2019, 10:21



"Во-вторых, красные стрелочки обозначают «наследуется от»."
- похоже я дальтоник (

Ответить

+6

Даниил Salesforce Developer в **Viseven** MASTER

10 сентября 2019, 21:02



[Вот](#) нормальная схема иерархии коллекций.

Ответить

+18

Kamoliddin Уровень 41, Самарканд, Узбекистан

30 октября 2021, 06:44



в формате если кому надо

Ответить

+4

Максим Боев Уровень 47, Санкт-Петербург, Russian Federation

29 января, 20:33



Спасибо! Я не понимаю, почему в материале до сих пор не поправили..

Ответить

+1

Валерий Емельянов Уровень 41, Краснодар, Россия

18 июня 2019, 23:11



На картинке не вижу стрелки наследования от ArrayList к AbstractList?

Ответить

+5

Пётр Уровень 41, Москва, Россия EXPERT

25 апреля 2019, 16:06



57 материи, не решены только 4 большие задачи и одна решена до 5й ступени. Прямо силком заставляют их решать для материи)

Ответить

+3

Pineapple Уровень 45, Абакан, Россия

21 августа 2021, 09:27



224 материи
так же почти все решено

НАЧАТЬ ОБУЧЕНИЕ

 Показать еще комментарии

ОБУЧЕНИЕ

Курсы программирования

Курс Java

Помощь по задачам

Подписки

Задачи-игры

СООБЩЕСТВО

Пользователи

Статьи

Форум

Чат

Истории успеха

Активности

КОМПАНИЯ

О нас

Контакты

Отзывы

FAQ

Поддержка



RUSH

JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в онлайн-IDE, необходимый минимум теории по основам Java и

НАЧАТЬ ОБУЧЕНИЕ

ПОДПИСЫВАЙТЕСЬ

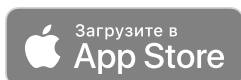
ЯЗЫК ИНТЕРФЕЙСА



Русский



СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ



"Программистами не рождаются" © 2022 JavaRush