

Реализации интерфейса Set, Queue

Java Collections
6 уровень, 9 лекция

ОТКРЫТА

— Ну как твой процессор?

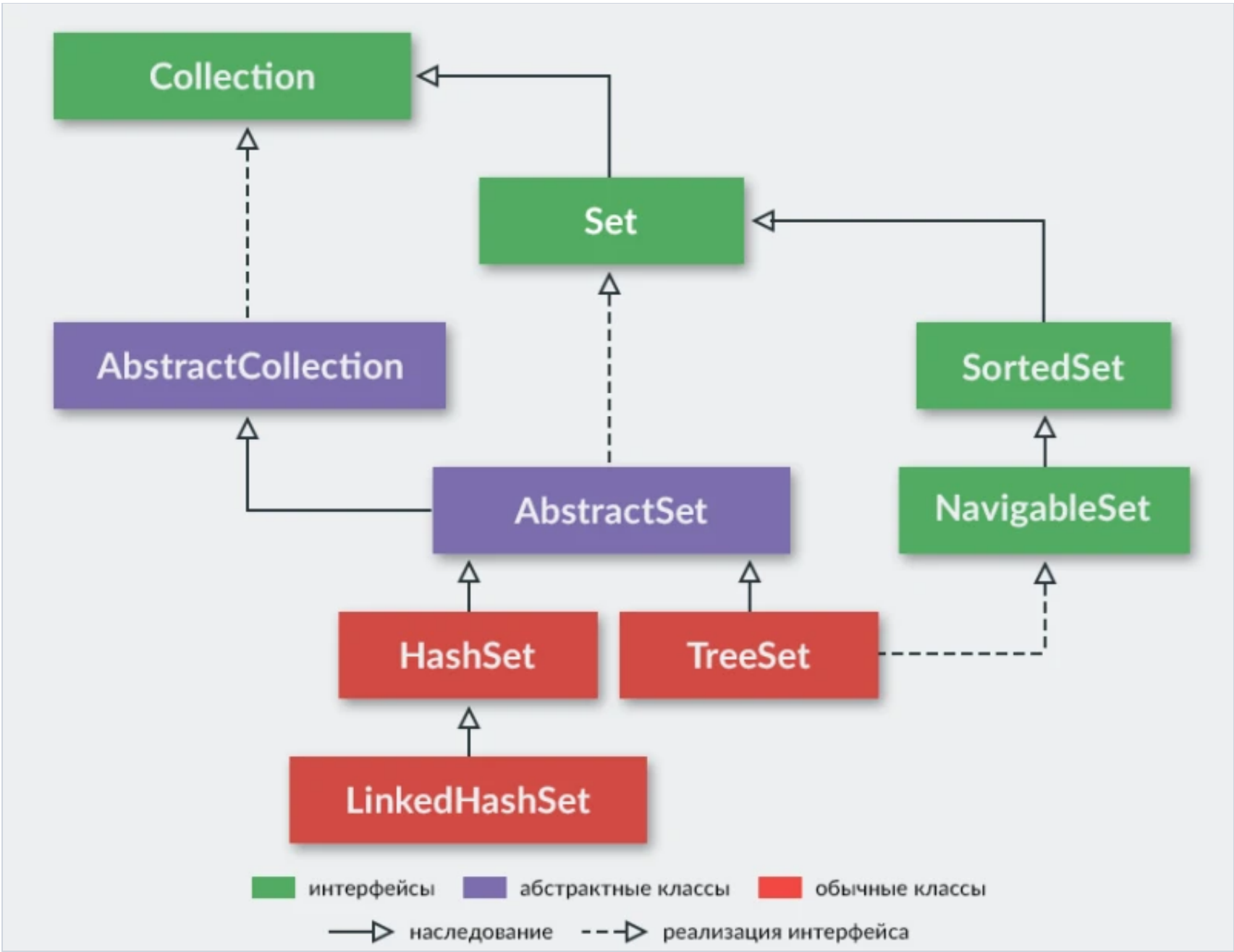
— Норм. Посидел час в жидком азоте и как новенький!

— Отлично. Тогда давай продолжим.

Коллекции Set.

Set переводится как множество. А множество, с математической точки зрения, — это набор уникальных элементов. Но т.к. не все программисты – математики, то обычно говорят, что Set – это коллекция уникальных элементов, или коллекция, которая не позволяет хранить одинаковые элементы.

Не знаю, давала Элли тебе иерархию наследования для Set, но если нет, то вот она:



HashSet – это коллекция, которая для хранения элементов внутри использует их хэш-значения, которые возвращает метод **hashCode()**.

Для простоты внутри `HashSet<E>` хранится объект `HashMap<E, Object>`, который и хранит в качестве ключей значения `HashSet`.

— Ничего себе!

— Использование hash-кодов позволяет довольно быстро искать, добавлять и удалять элементы из множества (Set).

И тот и другой активно используются внутри **HashSet/HashMap**.

Если ты забудешь реализовать метод **hashCode()**, то рискуешь, что твой объект в коллекции Set не будет найден, даже если он там есть.

— Да, помню, я помню. Ты мне уже рассказывал раньше об этом. Все робоуши прожужжал.

— Ок. Тогда вот тебе еще полезная информация.

Допустим, ты правильно реализовал **hashCode** и **equals** в своем классе и такой весь радостный хранишь объекты в Set’е.

Но потом ты взял и поменял один из объектов, при этом поменялись его внутренние данные, которые используются в вычислении **хэша**. И хэш объекта стал другим.

А это значит, что когда ты будешь его искать в Set’е, его скорее всего не найдут.

— Ничего себе! Это как же?

— Это всем известный косяк с работой хешей. Грубо говоря, поиск в HashSet (и в HashMap) гарантированно работает правильно, только если объекты – **immutable**.

— Ничего себе! И что, никто ничего не делает?

— Все делают вид, что проблемы не существует. Но на собеседованиях это частенько спрашивают, так что возможно, тебе стоит запомнить этот факт...

LinkedHashSet – это HashSet, в котором элементы хранятся еще и в связном списке. Обычный HashSet не поддерживает порядок элементов. Во-первых, официально его просто нет, во-вторых, даже внутренний порядок может сильно поменяться при добавлении всего одного элемента.

А у **LinkedHashSet** можно получить итератор и с его помощью обойти все элементы именно в том порядке, в котором они добавлялись в **LinkedHashSet**. Не часто, но иногда это может очень понадобится.

— Ясно. Люблю, когда у классов есть такие «на всякий случай» разновидности. Обычно такие случаи наступают не так уж и редко.

— **TreeSet** – это коллекция, которая хранит элементы в виде упорядоченного по значениям дерева. Внутри **TreeSet<E>** содержится **TreeMap<E, Object>** который и хранит все эти значения. А этот **TreeMap** использует **красно—**
черное сбалансированное бинарное дерево для хранения элементов. Поэтому у него очень быстрые операции **add, remove, contains**.

— Ага. Я помню, мы же совсем недавно это разбирали. А я еще думал – и где это применяется.

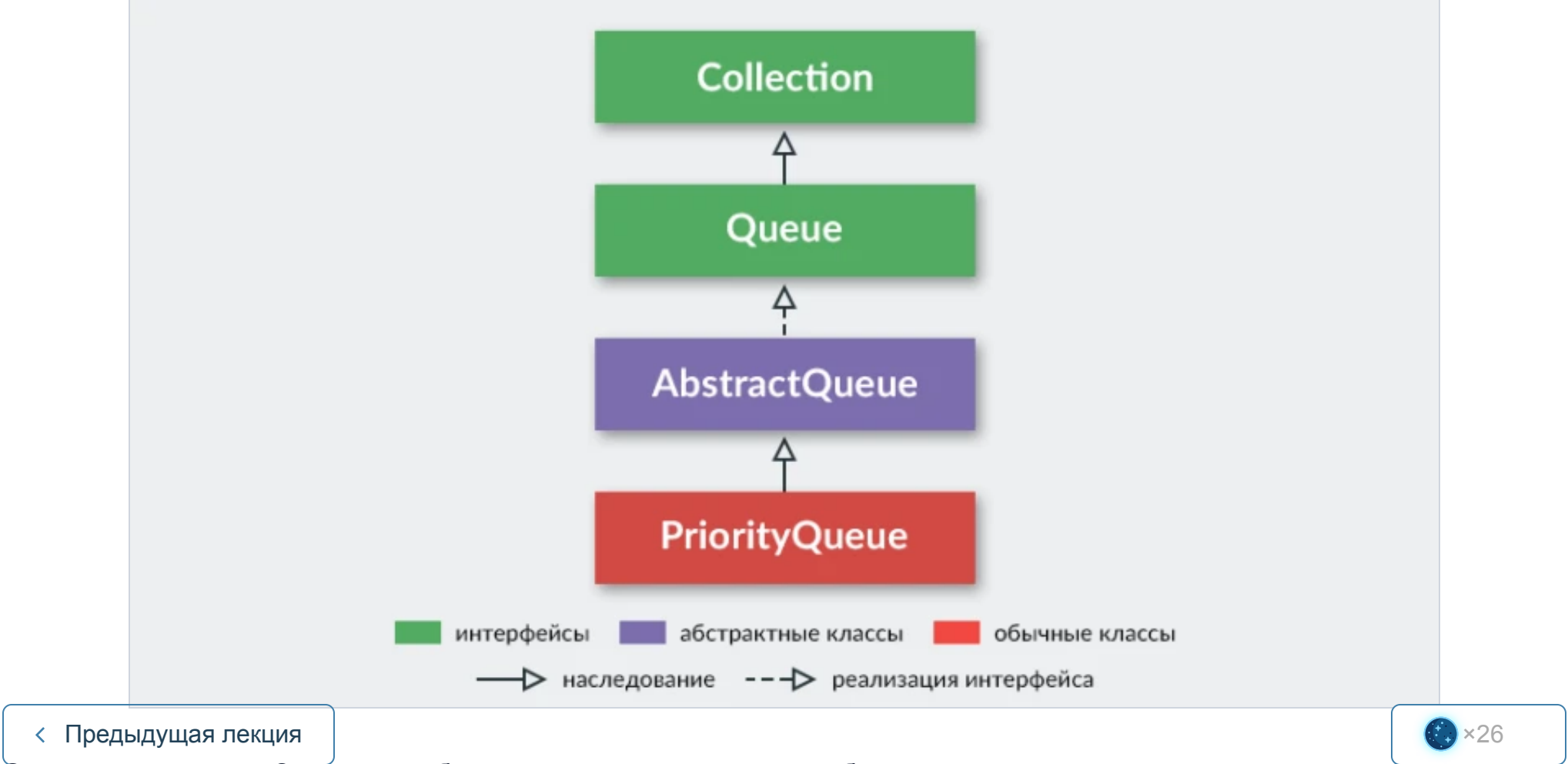
А оказывается, одни из самых популярных коллекций в Java используют это.

— Ага, кстати, на собеседованиях часто спрашивают про **TreeSet**. Обычно стараются подловить. Мол, если в TreeSet используется бинарное дерево, то тогда все элементы могут образовывать одну длинную ветку и при этом поиск будет очень долгим. Тут, кстати, стоит поставить такого наглеца на место, и заявить, что даже ребенку известно, что TreeSet и TreeMap используют сбалансированные **красно-черные** бинарные деревья, и поэтому такая ситуация невозможна в принципе.

— Ага. Хотелось бы увидеть лицо спрашивающего в этот момент. Я, пожалуй, даже заучу эту фразу. ...

А в принципе, Set оказался не таким уж и простым, как мне казалось в самом начале.

— Зато с **Queue** ситуация гораздо проще:



Queue – это очередь. Элементы добавляются в конец очереди, а выбираются из ее начала.

−+61+

Комментарии (24)

популярные

новые

старые

JavaCoder

Введите текст комментария

Лиза Воренувкина

Уровень 43, Кривой Рог, Ukraine

25 сентября 2021, 17:38

⋮

Давай До свидания!!! хххаааааа

Ответить

−+2+

Pig Man

Главная свинья в Свиарнике

14 февраля 2021, 16:27

⋮

Тебе задают вопросы на собеседовании не потому, что не знают ответ, а потому что хотят понять, знаешь ли ты его (как и на любом экзамене). Поэтому если бы я проводил собеседование и мне так ответили, этот человек просто пошел бы нахер и на работу мог бы не рассчитывать. Вот на его лицо в этот момент я бы посмотрел. Вывод: не надо так отвечать на любой вопрос кому-то на собеседовании

Ответить

−+7+

savc

Уровень 41, Санкт-Петербург, Россия

14 апреля 2021, 18:02

⋮

Да я думаю, что 99.9% людей на собеседовании не будут так выражаться. Многие боятся сказать что-то лишнее)

Ответить

−0+

Maks Panteleev

Java Developer в Bell Integrator

23 июля 2021, 11:31

⋮

если ты агрессивный и неприятный тип, это не значит, что все такие)

Ответить

−+1+

Дмитрий

Уровень 35, Москва

30 августа 2021, 14:45

⋮

ну да, может кому то нравится, что его унижают))))

Ответить

−+1+

Nikita Serkov

Backend Developer в Kameleoon

27 октября 2021, 18:29

⋮

Не уверен, что такой ответ кого то может обидеть. Нет в ответе ничего обидного, он всего лишь показывает, что ты это знаешь и это азы.

Ответить

−0+

НАЧАТЬ ОБУЧЕНИЕ

Это ж то же самое, что "и ёжику понятно", что здесь некрасивого? Конечно, достаточно часто бывает, что люди оскорбляются с какой-нибудь самой безобидной фигни, а потом доказывай сиди, что ничего плохого не имел в виду. Можно даже не понять, что ты сделал не так, а люди тебя уже запомнили и смотрят исподлобья, как на врага народа.

Ответить

−

+1

+

Hidden #213

Уровень 13 (Forever&Ever)

11 мая, 10:45

...

Ты на собесе тоже дерзил? Не отвечай, я знаю ответ. Небось накатил вискарика и разнёс всё шарашкину контору)))

Ответить

−

0

+

Maks Panteleev

Java Developer в Bell Integrator

11 мая, 13:17

...

да не, собеседы все проходили +- благоприятно, хотя я очковал конечно дико) ну сейчас бы разнес конечно если б пошел на собес, я теперь царь джавы))

Ответить

−

0

+

wan-derer.ru

Уровень 40, Москва, Россия

21 декабря 2020, 16:25

...

Вот про это изменение объекта с последующим разрушением хэша и immutable чтобы этого избежать - хотелось бы поподробнее!

Ответить

−

0

+

Илья

Backend Developer в СберТех

11 февраля 2021, 11:24

...

не меняй объекты)))

Ответить

−

+6

+

Артём В.

Уровень 41, Санкт-Петербург, Россия

12 декабря 2020, 13:47

...

"Даже ребенку известно, что TreeSet и TreeMap используют сбалансированные красно-черные бинарные деревья, и поэтому такая ситуация невозможна в принципе." - посмотрим на лицо собеседующего, когда я ему это заявлю ;D

Ответить

−

+6

+

Кех

Уровень 38, Тольятти

EXPERT

17 июля 2020, 13:41

...

Задался вопросом про null, может ли принимать сеты нуль [вот вообще ответ на этот вопрос](#) PS самое интересное там в коментах

Ответить

−

+14

+

Евгений

Ведущий инженер в ПАО Сбербанк

EXPERT

8 июля 2020, 22:13

...

Странно, что PriorityQueue является единственной официальной реализацией Queue, так как сама по себе эта реализация довольно специфична, и я пользовался ей один-два раза от силы. Из-за этого, когда я хочу сделать обычную очередь по принципу FIFO, приходится использовать LinkedBlockingQueue. Или я дурачок и есть обычная очередь, которую я не смог найти? Мне так-то обычно не требуется потокобезопасность.

Ответить

−

+1

+

Кех

Уровень 38, Тольятти

EXPERT

20 июля 2020, 11:15

...

а не подойдет ли для обычного FIFO и не потокобезопасный - LinkedList ?

Ответить

−

+1

+

Евгений

Ведущий инженер в ПАО Сбербанк

EXPERT

20 июля 2020, 11:38

...

Хе-хе, ну да. Недавно только про него читал. Подойдёт. Да и интерфейсы реализует те же, просто название не очень похожее на остальные Queue.

Ответить

−

+1

+

Soros

Уровень 39, Харьков, Украина

19 апреля 2020, 12:14

...

"Ясно. Люблю, когда у классов есть такие «на всякий случай» разновидности. Обычно такие случаи наступают не так уж и редко.", - сказал умудрённый профессиональным опытом программист Амиго.

Ответить

−

0

+

skybright

Project Manager в construction company

EXPERT

20 октября 2019, 19:55

...

Воу-воу-воу, полегче. Наконец то интересная лекция с офигенными нюансами, спасибо.

Ответить

−

+12

+

Евгений Пехтерев

Уровень 41, Санкт-Петербург, Россия

12 апреля 2018, 18:20

...

все видео и задачи пройдены успешно, а материи всего 279 и она очень стремительно уменьшается. чувствую, что скоро придется в саппорт писать.

НАЧАТЬ ОБУЧЕНИЕ

очень странно... есть пара не решенных задач и 994 материи...

Ответить

+2

Ярослав

Java Developer

MASTER

11 июля 2018, 18:03

Не решена одна задача (математическая рекурсия), а так же архиватор и ресторан из мультитрэда, всего 223 сейчас материи :)

Ответить

0

Юрій Юсько

Уровень 41, Украина

24 октября 2018, 21:31

Не решено 4 задачи - 907.
Знакомство с тегами
Рекурсия для мат. выражения
Определяем порядок захвата монитора
Ресторан(19)

Ответить

0

drive-by

Уровень 16, Москва, Россия

3 октября 2017, 18:31

yo

Ответить

0

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

JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский

"Программистами не рождаются" © 2022 JavaRush