



StackTrace

Java Syntax (/quests/QUEST_JAVA_SYNTAX)

9 уровень (/quests/lectures/?quest=QUEST_JAVA_SYNTAX&level=9), 1 лекция (/quests/lectures/questsyntax.level09.lecture01)

ОТКРЫТА



— Привет! Сегодня я расскажу тебе, что такое стек-трейс. Но сначала расскажу, что такое стек.

— Представь себе стопку бумаг — деловых поручений для некоторого исполнителя. Сверху на стопку можно класть новое задание, и с верха стопки задание можно брать. При таком подходе задания будут исполняться не по порядку поступления. Задание, положенное самым последним, будет взято исполнителем самым первым. Такая структура элементов коллекции называется стеком — стопкой.

— В Java для этого есть специальная коллекция — Stack. Это коллекция, у которой есть методы «добавить элемент» и «взять(достать/забрать) элемент». Как ты уже догадался, первым будет взят элемент, добавленный самым последним.

— Хм. Вроде не сложно и понятно.

— Отлично. Тогда сейчас объясню, что такое стек-трейс.

— Представь себе, что в Java функция А вызвала функцию Б, а та вызвала функцию В, а та, в свою очередь, функцию Г. Так вот, чтобы выйти из функции Б, нужно сначала выйти из функции В, а для этого выйти из функции Г. Это очень похоже на стек.

— А чем похоже?

— В стопке тоже, чтобы добраться до какого-то листка с заданием, надо довыполнить все задания, которые положили сверху.

— Ну, некоторая аналогия есть, но не уверен, что я все понял правильно.

— Смотри. Стек — это набор элементов. Как листы в стопке. Чтобы взять третий сверху лист, надо сначала взять второй, а для этого взять первый. Класть и брать листы можно всегда, но всегда взять можно только самый верхний.

— С вызовом функций то же самое. Функция А вызывает функцию Б, а та вызывает функцию В. И чтобы выйти из А, надо сначала выйти из Б, а для этого надо выйти из В.

— Подожди. Если я все правильно понял, то весь этот стек сведется к «взять можно только самый последний положенный лист», «выйти можно только из последней функции, в которую зашли». Так?


— Да. Так вот — последовательность вызовов функций — это и есть «стек вызовов функций», он же просто «стек вызовов». Функция, вызванная последней, должна завершиться самой первой. Давай посмотрим это на примере:

Получение и вывод текущего стека вызовов:

Вот какой результат мы получим:

```
1 public class ExceptionExample
2 {
3     public static void main(String[] args)
4     {
5         method1();
6     }
7
8     public static void method1()
9     {
10        method2();
11    }
12
13    public static void method2()
14    {
15        method3();
16    }
17
18    public static void method3()
19    {
20        StackTraceElement[] stackTraceElements = Thread.currentThread().getStackTrace();
21        for (StackTraceElement element : stackTraceElements)
22        {
23            System.out.println(element.getMethodName());
24        }
25    }
26 }
```

getStackTrace
method3
method2
method1
main

ЗАДАЧА  Java Syntax, 9 уровень, 1 лекция

РЕШЕНА

★★★★☆

Набираем код

×10

Иногда думать не надо, строчить надо! Как ни парадоксально звучит, порой пальцы «запоминают» лучше, чем сознание. Вот почему во время обучения в секретном центре JavaRush вы иногда встречаете задания на набор кода. Набирая код, вы привыкаете к синтаксису и зарабатываете немного материи. А ещё — боретесь с ленью.

Открыть

— Ок. С вызовом функций похоже все понятно. А что это еще за `StackTraceElement`?

— Java-машина ведет запись всех вызовов функций. У нее есть для этого специальная коллекция — стек. Когда одна функция вызывает другую, Java-машина помещает в этот стек новый элемент `StackTraceElement`. Когда функция завершается этот элемент удаляется из стека. Таким образом, в этом стеке всегда хранится актуальная информация о текущем состоянии «стека вызовов функций».

— Каждый `StackTraceElement` содержит информацию о методе, который был вызван. В частности можно получить имя этого метода с помощью функции `getMethodName`.

— В примере выше ты можешь видеть демонстрацию этого дела:

- 1) Получаем «стек вызовов»:
- 2) Проходимся по нему с помощью цикла `for-each`. Надеюсь, ты его еще не забыл.
- 3) Печатаем в `System.out` имена методов.

— Интересная штука и, похоже, совсем не сложная. Спасибо, Риша!

< (/quests/lectures/questsyntax.level09.lecture00)

×9 > (/quests/lectures/questsyntax.level09.lecture02)



0



1



0



0



0

+21

Комментарии (35)

популярные

новые

старые

Никита

Витёк (Vitek) 9 уровень

позавчера, 16:43



— Интересная штука и, похоже, совсем не сложная.

↘(ツ)↗↘(ツ)↗↘(ツ)↗↘(ツ)↗

Ответить

0

skybright 17 уровень

21 февраля, 22:49



в 20 строке знак '=' не возможно правильно установить

Ответить

0

Ренат 10 уровень, Уфа

9 марта, 22:28



ты скорее всего перед этим знаком неверно написал слово "elements", НАВЕРНЯКА АБЫЛ БУКВУ "s" на конце

Ответить

0

Stanislav Zabiya 21 уровень, Днепр

27 декабря 2017, 15:08



Я не совсем понял, почему в лекции стэк называют коллекцией? по всем признакам - это массив. Или я не прав?

Ответить

0

GAZ 21 уровень

27 декабря 2017, 16:26



Массив это частный случай коллекции : <https://javatalks.ru/topics/42051>

Ответить

0

Stanislav Zabiya 21 уровень, Днепр

27 декабря 2017, 17:44



Разве? В официальных доках ничего подобного нет
<https://docs.oracle.com/javase/7/docs/api/java/util/Collections.html>
значит это разные классы

Ответить

0

Redas Shuliakas 21 уровень

6 февраля, 08:00



Массив это не класс. А ArrayList это класс в который помещён массив, такое чувство что вам не хочется получить ответ на свой вопрос, а хочется тупо поспорить.

Ответить

+1

Stanislav Zabiya 21 уровень, Днепр

12 февраля, 09:54



Массив не класс?))) Вы поэтому его экземпляр создаёте при помощи оператора new?
Удивительное рядом (с)

Ответить

0

Redas Shuliakas 21 уровень

13 февраля, 08:13



Причём тут оператор new? Всё классы в джава пишутся с большой буквы, а не с маленькой, это раз, и два, набери в гугле "массив класс java" и найдёшь ноль совпадений своего утверждения.

Ответить

0

Stanislav Zabiya 21 уровень, Днепр

13 февраля, 13:22



Открываем официальные доки <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html> и что же мы видим в первой же строке? О чудо! "An array is a container object...". И, по секрету - оператор new создаёт экземпляр класса - то есть объект. ;-)

Ответить

0

Redas Shuliakas 21 уровень

13 февраля, 16:51



"An array is a container object that holds a fixed number of values of a single type." тут написано что это объект, то есть сущность, но не объекты о которых мы говорим, на английском объект в ООП называется instance. 7 Лет проживания в Великобритании.

Ответить

+2

Stanislav Zabiya 21 уровень, Днепр

14 февраля, 20:21



Да, пожалуй, Вы правы - массив - это не класс в привычном понимании, а специализированный объект. Но вопрос остаётся открытым - стэктрейс имеет все признаки именно массива, обращение к его элементам мы осуществляем по индексу [index] , а не вызовом метода get(index). Почему тогда написано - коллекция?

Ответить

0

Redas Shuliakas 21 уровень

18 февраля, 19:46

...

На этом ресурсе куча ошибок, допустим перепутаны сужение и расширение типов, в чём признался мне сам создатель ресурса, ошибка известна уже 2+ лет, и до сих не исправлена, вывод? Ошибка может просто, автор не господь бог же, не парься.

Ответить

0

Николай 9 уровень

1 декабря 2017, 15:38

...

Это не слышано, теперь чтобы повторно набрать код материала нужно подписка на ПРЕМИУМ???! До этого было без ограничений...

Ответить

+1

Michail Neger 21 уровень, Санкт-Петербург

5 августа 2017, 20:02

...

2) Проходимся по нему с помощью цикла for-each. Надеюсь, ты его еще не забыл. Забудешь это после уровня 8 :)

Ответить

+13

Степан Карсаков 21 уровень, Минск

10 октября 2017, 00:06

...

Хм тм можно было не пользоваться на 8 уровне.

Ответить

0

Дима Авангард 10 уровень

17 июля 2017, 21:43

...

за открытие этой лекции с меня 2 раза списали по 9 материи

Ответить

+2

Степан Карсаков 21 уровень, Минск

10 октября 2017, 00:11

...

А нафиг она тебе нужна?

Ответить

+5

Света Светикова 9 уровень, Москва

вчера, 10:34

...

ипотеку ими гасить будет))

Ответить

0

Александр Баглюк 16 уровень

15 июля 2017, 16:22

...

может порекомендует кто какие видео уроки на эту тему

Ответить

0

Марат 1988 12 уровень

12 июня 2017, 23:34

...

Может вопрос мой покажется глупым. Строго не судите, только учусь. Зачем нужно изучать стэки? Какую это несет за собой практическую пользу (ну, например, при написании софта)?

Ответить

+1

Vaiki 18 уровень, Минск

16 июня 2017, 09:23

...

для поиска исключений

Ответить

0

Валерий Бойко 23 уровень, Киев

27 июня 2017, 21:30

...

И для понимания рекурсии (в более крупных задачах очень полезная вещь)

Ответить

0

Harvey 27 уровень, Минск

29 августа 2017, 21:01

...

Когда начнется многопоточность (здесь нам уже показали немного, когда набирали код, класс Thread), чтобы видеть на каком моменте возникла проблема в работе программе и в каком потоке, трассировка стека помогает увидеть все в деталях.

Ответить

+7

Валерий 16 уровень

12 июня 2017, 23:32

...

копипаст не пробовали?

Ответить

0

Allesh001 21 уровень

28 августа 2017, 18:11

...

начинающим разработчикам категорически не рекомендуется использовать его...
Причем вообще. Даже System.out.println() / in

Ответить +5

Alex Fedoniuk 15 уровень

12 июня 2017, 12:15 ...

Все работает! Аккуратнее набирайте, пробелы все учитывайте

Ответить

0

[↺](#) Загрузить еще

[javarush.ru/](#), [G+](#) [_](https://plus.google.com/114772402300089087607)(https://plus.google.com/114772402300089087607), [_](https://twitter.com/javarush_ru)(https://twitter.com/javarush_ru), [i](#)



Программистами не рождаются
© 2018