



Как работают исключения

Java Syntax (/quests/QUEST_JAVA_SYNTAX)

9 уровень (/quests/lectures/?quest=QUEST_JAVA_SYNTAX&level=9), 5 лекция (/quests/lectures/questsyntax.level09.lecture05)

ОТКРЫТА



— Хочу рассказать тебе немного о том, как работают исключения. Приведённый ниже пример будет показывать примерную логику работы:

Код с использованием исключений

```
1 class ExceptionExampleOriginal
2 {
3
4
5     public static void main(String[] args)
6     {
7         System.out.println("main begin");
8         try
9         {
10             System.out.println("main before call");
11
12             method1();
13
14
15             System.out.println("main after call");
16         }
17         catch (RuntimeException e)
18         {
19
20
21             String s = e.getMessage();
22             System.out.println(s);
23         }
24         System.out.println("main end");
25     }
26
27     public static void method1()
28     {
29         System.out.println("method1 begin");
30         method2();
31
32         System.out.println("method1 end");
33     }
34
35     public static void method2()
36     {
37         System.out.println("method2");
38         String s = "Message: Unknown Exception";
39         throw new RuntimeException(s);
40     }
41
42 }
43 }
```

Примерная расшифровка

```
1  public class ExceptionExample
2  {
3      private static Exception exception = null;
4
5      public static void main(String[] args)
6      {
7          System.out.println("main begin");
8
9
10         System.out.println("main before call");
11
12         method1();
13
14         if (exception == null)
15         {
16             System.out.println("main after call");
17         }
18         else if (exception instanceof RuntimeException)
19         {
20             RuntimeException e = (RuntimeException) exception;
21             exception = null;
22             String s = e.getMessage();
23             System.out.println(s);
24         }
25         System.out.println("main end");
26     }
27
28     public static void method1()
29     {
30         System.out.println("method1 begin");
31         method2();
32         if (exception != null) return;
33         System.out.println("method1 end");
34     }
35
36     public static void method2()
37     {
38         System.out.println("method2");
39         String s = "Message: Unknown Exception";
40         exception = new RuntimeException(s);
41         return;
42     }
43 }
```

— Ничего не понятно.

— Ок. Давай я тебе объясню, что происходит.

— В примере слева мы по цепочке вызываем несколько методов. В `method2` мы специально создаем и выкидываем исключение (иницилируем ошибку).

— В примере справа показано, что примерно при этом происходит.

— Посмотри на `method2`. Создание исключение превратилось вот во что: создали объект типа `RuntimeException`, сохранили его в специальную переменную `exception` и тут же вышли из метода — `return`.

— В методе `method1`, после вызова `method2` стоит проверка — **есть исключение или нет, если исключение есть, тогда метод `method1` тут же завершается**. Такая проверка неявно производится после вызова каждого(!) метода в Java.

— Ого!

— Вот тебе и ого.

— В колонке справа в методе `main` я написал, что примерно происходит при перехвате исключения с помощью конструкции `try-catch`. Если исключения не было, то все продолжает работать, как и запланировано. Если исключение было, и оно было такого типа, как указано в `catch`, тогда мы его обрабатываем.

— А что значит `throw` и `instanceof`?

— Посмотри на последнюю строку `throw new RuntimeException(s)`. Таким способом мы создаем и кидаем исключение. Пока мы так делать не будем. Это только для примера.

— А с помощью команды « `a instanceof B` » мы проверяем, имеет ли объект `a` тип `B`. Т.е. имеет ли объект, который хранится в переменной `exception`, тип `RuntimeException`. Это логическое условие.

— Ну, вроде понятно. Немного.

[< \(/quests/lectures/questsyntax.level09.lecture04\)](/quests/lectures/questsyntax.level09.lecture04)

[×9 > \(/quests/lectures/questsyntax.level09.lecture06\)](/quests/lectures/questsyntax.level09.lecture06)



0



0



0



0



0

+20

Комментарии (29)

популярные

новые

старые

Никита

Anastasiia 11 уровень

4 марта, 15:53



Ответить

0

Радхараман Скороход 14 уровень

16 марта, 08:16



Мы же с переменной в этом коде работаем. А значит - ее нужно создать и проинициализировать. Иначе, если вы напишете exception = new RuntimeException(s), вас не поймут и спросят - что за exception такой? Какой тип, модификатор доступа и тд.

Ответить

0

Anastasiia 11 уровень

18 марта, 16:09



```
else if (exception instanceof RuntimeException)
{
    RuntimeException e = (RuntimeException) exception;
    exception = null;
}
```

все равно эта конструкция не очень ясна. Если исключение является типом RuntimeException, то что мы делаем здесь RuntimeException e = (RuntimeException) exception ?

Ответить

0

Радхараман Скороход 14 уровень

18 марта, 20:27



Я извиняюсь, я вас не понял, хотя вы явно сказали про elseif, почему-то я подумал, что речь идет о конструкции в начале.

```
1 private static Exception exception = null;
```

Тут как. Если объект exception так или иначе [произошел](#) от класса RuntimeException то мы запикиваем это исключение в переменную e типа RuntimeException (уже конкретного исключения) и обнуляем изначальное общее исключение (уничтожаем его, чтобы согласно заданному изначально поведению дальше не летело). Ну а потом соответственно выполняем тело кода уже с e.

Ответить

+2

Anastasiia 11 уровень

19 марта, 02:12



Ответить

0

Алексей 21 уровень

26 февраля, 13:15



Ответить

0

Alexandr Dudyak 9 уровень

22 сентября 2017, 16:06



Ответить

0

Дмитрий 31 уровень

8 октября 2017, 19:14



Ответить

0

Егор 18 уровень

3 ноября 2017, 18:46



интересно, у меня 343 её сейчас
при этом не выполнено 3 задания примерно на 15 материи. Все видео просмотрены, все написания текста тоже. Ты лгун)

Ответить

0

Дмитрий 31 уровень

3 ноября 2017, 20:27



Прошел уже конечно месяц, и материи у меня уже 991, но я точно помню, что на тот момент у меня было где-то 403-406 материи, как раз, наверное, те пару заданий и были на 50 материи, ~~было~~ ты не упустил шанс проанализировать, который тоже было на моем

лишь ты не учитываешь премиум задания, которых тоже было не мало

Ответить

0

Егор 18 уровень

5 ноября 2017, 07:23

...

Дмитрий, извини, я не подумал. У меня нет премиум подписки.

Ответить

0

Юрий Кузнецов 15 уровень

15 января, 17:47

...

у меня 362 мат.

Ответить

0

Антон 19 уровень

1 февраля, 14:20

...

выполнены все задачи 368 мат.

Ответить

+3

Игорь Богачик 9 уровень

3 февраля, 16:34

...

у меня тоже все и 365

Ответить

0

Shohrux Yusupov 12 уровень

10 сентября 2017, 14:06

...

Нормальное же объяснение. С первого раза улавливаю. Задачи решаю по часу. А когда читаю комментарии о том, кто как решил, думаю что я самый отстающий. А тут наоборот

Ответить

+5

ilya 35 уровень

25 ноября 2017, 08:58

...

бро, я тебя понимаю)

Ответить

0

Антон 19 уровень

1 февраля, 14:22

...

Те кто кричит, что все просто либо вундеры, либо брехуны. Вундеров не много:))) А так, такая же чепуха, некоторые задачи дня по два решал, пока не проспиться ничего в голову не приходит. Общаюсь с другими ребятами понимаю-это нормально. Программистами не рождаются. Терпение и труд.

Ответить

+3

P0huber 14 уровень

15 августа 2017, 18:34

...

Ну а теперь прочтите объяснения Вани Головача на Хабре:

[Исключения в Java, Часть I \(try-catch-finally\)](#)

[Исключения в Java, Часть II \(checked/unchecked\)](#)

Не забудьте плюсовать полезные комментарии, это поможет другим быстрее найти полезную информацию. Рад был помочь, Ваш P0huber

Ответить

+162

Андрей Кожарин 16 уровень

9 октября 2017, 16:02

...

Благодарю за ссылки

Ответить

0

Виталий Малый 21 уровень, Днепр

29 июля 2017, 19:11

...

это ж надо такую тему так запутанно объяснить

Ответить

+5

Влад Макаренко 11 уровень, Киев

3 августа 2017, 16:35

...

LOL, после гугла то же подумал))

Ответить

0

ufypbyf 17 уровень

23 июля 2017, 01:25

...

FYI, в IDEA для обертывания блока кода в try-cath, можно использовать горячие клавиши Ctrl+Alt+T, выделив перед этим нужный блок

Ответить

+32

Max Miheev 24 уровень, Москва

24 мая 2017, 15:48

...

Вот тебе и ого. есть ли адекватные видосы про исключения ?

Ответить

0

Игорь 19 уровень, Новосибирск

10 мая 2017, 07:01

Непонятно как он определяет какое исключение надо бросить. Не тогда, когда мы его явно бросаем (throw), а когда выполняем какое-то действие. Например, делим на 0 или читаем файл, которого нет. Посмотрел коды наследуемых от Exception классов - они практически пустые.

Ответить

0

Алексей Горяйнов 16 уровень, Москва

11 июня 2017, 08:54

Как я понимаю, исключения бросают сами методы (IDE показывает в описании метода), т.е. что-то вроде "эм... а как мне вывести null?" и ждет что либо мы обработаем ошибку и скажем ему это, либо прекратим издеваться)))

Ответить

0

Dimenterious Fox 11 уровень

8 июля 2017, 16:53

Я полагаю это уже захардкожено в реализации JVM.

Ответить

+1

Igor Kouzmin 17 уровень

30 марта 2017, 10:24

теперь могу сам себе ответить. потому что RuntimeException() это непроверяемое исключение и его можно не пробрасывать, т.к компилятор не требует обязательной их обработки.

Ответить

0

Alex False 10 уровень, Киев

17 апреля 2017, 22:57

красава, к успеху пришел)

Ответить

0

Наиль Гарифуллов 16 уровень, Самара

25 апреля 2017, 17:57

Спасибо! Задавался таким же вопросом.

Ответить

0

[jsh.ru/](http://javarush.ru/), [G+ \(https://plus.google.com/114772402300089087607/\)](https://plus.google.com/114772402300089087607/), [Twitter \(https://twitter.com/javarush_ru/\)](https://twitter.com/javarush_ru/)



Программистами не рождаются
© 2018