Карта квестов Лекции CS50 Android

Рекурсия

Java Collections 4 уровень, 1 лекция

ОТКРЫТА

— Привет, Амиго. Сегодня Билаабо расскажет тебе, что такое рекурсия.



Как ты знаешь, в Java одни методы вызывают другие методы. При этом, при вызове метода, в него передаются конкретные значения аргументов, а во время его работы локальные переменные метода принимают некоторые значения.

— Ага.

— И как ты знаешь, внутренние переменные разных методов независимы друг от друга.

— Ага.

— Так вот, представь ситуацию, когда метод может вызвать сам себя. Именно она называется рекурсией. Пример:

```
Пример
```

```
public static void main(String[] args)
 1
 2
 3
      countDown(10);
 4
     }
 5
     public static void countDown(int x)
 6
 7
     {
      if (x <=0)
 8
       System.out.println("Boom!");
9
      else
10
      {
11
       System.out.println(x);
12
       countDown(x - 1);
13
      }
14
15
     }
```

```
1
      10
 2
      9
 3
      8
      7
 5
      6
      5
 6
 7
 8
      3
 9
      2
10
      1
11
      Boom!
```

- Честно говоря, вижу, что метод в коде сам себя вызывает, но не понимаю, что именно при этом происходит.
- Да примерно то же, что и при вызове другого метода.
- Нет, я спрашиваю, что происходит с переменными? С их значениями? И как мы выходим из метода? Или мы выходим из всех сразу?
- Господи. Да все гораздо проще. Представь что метод, который вызывает сам себя, размножили много раз. Тогда будет аналогичная ситуация:

Рекурсивный вызов метода

```
public static void main(String[] args)
1
 2
     {
 3
      countDown(3);
 4
     }
 5
     public static void countDown(int x)
 6
7
     {
      if (x <=0)
8
       System.out.println("Boom!");
9
      else
10
      {
11
       System.out.println(x);
12
       countDown(x - 1);
13
      }
14
15
     }
```

Что происходит «на самом деле»

```
public static void main(String[] args)
 1
 2
     {
      countDown1(3);
 3
 4
     }
 5
     public static void countDown1(int x)
 6
 7
     {
      if (x <=0)
 8
       System.out.println("Boom!");
 9
      else
10
11
        System.out.println(x);
12
        countDown2(x - 1);
13
      }
14
15
     public static void countDown2(int x)
16
17
     {
      if (x <=0)
18
       System.out.println("Boom!");
19
      else
20
      {
21
       System.out.println(x);
22
        countDown3(x - 1);
23
      }
24
25
     }
     public static void countDown3(int x)
26
     {
27
      if (x <=0)
28
        System.out.println("Boom!");
29
      else
30
```

```
countDown4(x - 1);
33
       }
34
35
      }
36
37
      public static void countDown4(int x)
38
       if (x <= 0)
39
        System.out.println("Boom!");
40
41
       else
       {
42
        System.out.println(x);
43
        countDown5(x - 1);
44
45
       }
      }
46
```

```
Вывод на экран:
                                                             Вывод на экран:
         3
                                                                      3
    1
                                                                 1
    2
         2
                                                                 2
                                                                      2
    3
         1
                                                                 3
                                                                      1
    4
         Boom!
                                                                 4
                                                                      Boom!
```

Т.е. каждый раз при вызове метода (даже самого себя) создаются новые переменные, которые хранят данные для этого метода. Никаких общих переменных нет.

При каждом вызове в памяти появляется еще одна копия аргументов метода, но уже с новыми значениями. При возвращении в старый метод, там используются его переменные. Т.е. при рекурсии фактически мы вызываем другой метод, но с таким же кодом как наш!

- Ясно. А как работает выход из этих методов? Можно пример?
- Ладно. Один пример стоит тысячи слов. Вот тебе пример:

```
Рекурсивный вызов метода
        public static void main(String[] args)
   1
   2
        {
   3
         print(3);
   4
        }
   5
        public static void print(int x)
   6
   7
         if (x <=0)
   8
          System.out.println("Boom!");
   9
         else
  10
         {
  11
          System.out.println(x);
  12
          print(x - 1);
  13
          System.out.println(x);
  14
  15
         }
        }
  16
```

Что происходит «на самом деле»

```
public static void main(String[] args)
1
 2
     {
      print1(3);
 3
     }
 5
     public static void print1(int x)
6
7
      if (x <=0)
8
       System.out.println("Boom!");
      else
10
      {
11
       System.out.println(x);
12
       print2(x - 1);
13
       System.out.println(x);
14
15
      }
     }
16
17
```

```
if (x <=0)
20
        System.out.println("Boom!");
21
       else
       {
23
24
        System.out.println(x);
        print3(x - 1);
25
        System.out.println(x);
26
27
      }
28
      }
29
30
      •••
```

```
Вывод на экран:
                                                                Вывод на экран:
    1
         3
                                                                    1
                                                                          3
    2
         2
                                                                    2
                                                                          2
    3
         1
                                                                    3
                                                                          1
         Boom!
                                                                          Boom!
    4
                                                                    4
    5
                                                                    5
         1
                                                                          1
    6
         2
                                                                    6
                                                                          2
    7
                                                                    7
         3
                                                                          3
```

— Ок. Вроде понял. А зачем нужна рекурсия?

— Есть очень много задач, которые разбиваются на отдельные подзадачи, идентичные первоначальной задаче. Например, надо обойти все элементы XML-дерева. У каждого элемента может быть несколько дочерних элементов, а у них — свои.

Или тебе нужно вывести список файлов, которые есть в директории и все ее поддиректориях. Тогда ты пишешь метод, который выводит файлы текущей директории, а потом для получения файлов всех поддиректорий вызываешь его же, но с другим параметром – поддиректорией.

Пример:

```
Вывод всех файлов на экран из директории и её поддиректорий
      public static void main(String[] args)
 1
 2
 3
       printAllFiles(new File("c:/windows/"));
 4
      }
 5
 6
      public static void printAllFiles(File dir)
 7
       for (File file : dir.listFiles())
 8
 9
       {
10
        if (file.isDirectory())
         printAllFiles(file);
11
        else
12
         System.out.println(file.getAbsolutePath());
13
14
       }
15
      }
```

Строка 8 – получаем список всех файлов (и директорий) в директории dir.

— Ок. Вроде понял. Спасибо, Билаабо.

< Предыдущая лекция





ментарии (55)	популяры	ные	— HOBЫӨ	старь
JavaCoder				
Введите т	екст комментария			
LuneFox инх	кенер по сопровождению в BIFIT ехрегт		17 января	, 03:30
Если не до на <u>этой</u> стр	конца поняли, что такое рекурсия, прочитайте мой комментарий от 1 анице.	7 янв	заря 2022 го	да вот
Ответить				+40
Frey	Green Уровень 41, Нижний Новгород, Россия		24 января	, 23:03
Ген	ий			
Ответ	ИТЬ			+3
hidd	en #2595317 Уровень 45		24 февраля	, 13:19
ТЫ	ллохой			
Ответ	ИТЬ			+2
Рома	н Кончалов Уровень 28, Россия ехрект		30 марта	, 13:19
Поі	нимание рекурсии – условие выхода из рекурсии, гениально			
Ответ	ИТЬ			+ 1
Ален	сандр Горохов Уровень 25, Дятьково, Россия		26 июля	, 15:40
Пха	ixxaxa!)))			
Ответ	ИТЬ			# +1
Artu	Z Уровень 37, Russian Federation		2 августа	, 11:50
Пеј	реходил по ссылкам два часа кряду и преисполнился			
Ответ	ИТЬ			+2
Lune	Fox инженер по сопровождению в BIFIT ехрект		2 августа	, 23:43
Вкг	адк оверфлоу?)			
Ответ	ИТЬ			• 0
Artu	Z Уровень 37, Russian Federation		4 августа	, 17:05
что	бы прийти в себя пришлось идти за всяким, случилось zakladkOverFl	ow		
Ответ	ИТЬ			© 0
Maks Pantel	ееv Java Developer в Bell Integrator		7 июля 2021	, 10:02
Самое вре	ия начать изучение рекурсии			
Ответить				+10
it008	8 Уровень 40, Худжанд, Таджикистан		7 июля 2021	, 13:05
нет	думаю еще рано))			
Ответ	ИТЬ			+7
Max	Zap Уровень 41	26	октября 2021	. 11:57 🗔

O 0 **C** Maks Panteleev Java Developer в Bell Integrator 26 октября 2021, 12:15 действительно, ты же не можешь выучить рекурсию, пока не выучишь рекурсию, получается я не понимал рекурсию, понимая рекурсию) Ответить **+1 (7)** Александр Черенков Уровень 37, Бердск, Россия 15 апреля 2021, 09:22 Начался 34 уровень, как и в 33 предыдущих тема первой лекции берется из пройденного материала (очевиден подход: повторенье - мать ученья), но комменты всё те же - что происходит? мы это уже проходили!. Видимо оставшиеся 6 уровней будут нанинаться так же... Ответить 0 0 **Ars** Уровень 41 20 ноября 2021, 18:51 Вот если бы лекции не делали вид, что дают новую тему, а начинались бы со слов "Давай ещё раз закрепим пройденное", то комментов на эту тему было бы меньше! Ответить 0 0 Justinian Judge B Mega City One MASTER 23 ноября 2021, 20:13 Все-равно бы нашлись те, у кого был бы свой взгляд, когда и что следует изучать. Те кто ездил в долгие поездки с родственниками, у которых нет водительских прав, но зато есть активная жизненная позиция и свое мнение на все случаи жизни, могли сталкиваться с их советами относительно дорожной обстановки. Такая природа человека. Ответить **+1 (7) Ars** Уровень 41 23 ноября 2021, 20:21 Всегда найдутся люди, у которого есть свой другой взгляд. Это не повод не делать лекции так, чтобы у людей было меньше вопросов и возмущений. Я же в своём предыдущем комментарии пишу не о том когда и что изучать. А том, как подавать лекцию. Ответить **+2 (1)** Justinian Judge B Mega City One MASTER 23 ноября 2021, 20:55 Ты писал о том, что можно написать, мол это повторение. Но я честно не помню, чтобы была отдельная лекция/тема по рекурсии, ты можешь сказать когда это было, а то я запмятовал. Если была, то и действительно, гораздо лучше написать что это повторение Ответить **C** +1 **C Ars** Уровень 41 24 ноября 2021, 10:51 Было как минимум две задачи на рекурсии в разных местах. Сначала давали задачу на рекурсию, просто написав в условии что используй рекурсию. Много позже была задача на рекурсию и было написано прочитать про неё дополнительно самому. Поэтому после того как авторы курса явно сказали, что рекурсия есть и узнай о ней сам, возникает непонимание/удивление/возмущение, когда о ней рассказывают без учёта этого факта. Ответить +2 barracuda Уровень 41, Санкт-Петербург, Россия 9 февраля 2021, 00:03 Если бы в JR был 50-й уровень, на нем бы нам рассказали, что поля класса инициализируются значением по умолчанию. А на 60-м - что за парематры (String[] args) у метода main. **Anonymous #2631266** Уровень 41, Одесса 19 мая 2021, 20:49 Г••• Большая задача Hello, World! Ответить +14 Dmitry Gorchakov Уровень 33, Москва, Россия 30 июня 2021, 23:47 Задание 1. Создай класс и назови "Solution" Задание 2. Просто нажми "Пройдено" Ответить 0 **Ars** Уровень 41 20 ноября 2021, 18:52 Второе есть в каждой большой задачке. Иногда даже два раза. Так что нужно что нибудь попроще. Открыл задачу и сразу засчиталось выполнение. Ответить **+1 (7)** THE TOLEN VANDOUS OF MANUEL DOS

Ответить		⇔ +5 €
ΙN	пья Backend Developer в СберТех	7 февраля 2021, 12:53
	поддерживаю, в свое время как-то посмотрел и сразу все поня нужно держать в голове мысленно return (в том числе и в мето	
От	гветить	9 0 9
Александ	цр один Эс Backend Developer	20 ноября 2020, 09:19
Повторе	ение – мать учения	
Ответить		⇔ +3 €
Дмитрий	Уровень 26, Ровно, Украина	21 июля 2020, 23:00 •••
	адач о построении дерева, факториала (через Big Integer) пора набились мы серилизацией и xml парсингами.	а и рекурсию изучить. А то уже
Ответить		⇔ +8 €
Aleksey S	Serikov Уровень 41, Гомель, Беларусь	10 мая 2020, 18:37
	о не понял почему в примере такой вывод в консоль: 321 boom eo: <u>рекурсия</u>	ı! 321, то рекомендую посмотреть
Ответить		⇔ +9 €
lva	an D Уровень 35	5 декабря 2020, 07:23
3	3 2 1 1 2 3	
От	гветить	⇔ +1 €
Anton Ryı	marenko Уровень 41	24 февраля 2020, 19:19 •••
Спасибо	о что через два уровня после задач на рекурсию мы учим реку	урсию))
Ответить		+20
VI	adimir Kashin Backend Developer в ГОСНИИАС	4 апреля 2020, 14:05
Į	Действительно xD	
От	гветить	\$\phi +1 \$
Al	lexandr Ikert Уровень 35, Москва, Россия	15 июля 2020, 20:43
a	а это такая фишка JavaRush сразу отсев идет тех у кого пукан	горит)))
От	гветить	🗢 +5 🕻
Vorlock V	′ровень 31, Днепр, Украина	10 января 2020, 23:17 👓
		10 япваря 2020, 23.17
	рекурсия дается на уровне начитки java core общем)	
Ответить		⇔ +9 ©
	О Помости от се мосто посто по	
жжете в	рекурсия дается на уровне начитки java core общем) С Показать еще комментарии	⇔ +:

сообщество

КОМПАНИЯ

Курсы программирования О нас Пользователи

Kypc Java Статьи Контакты

Форум Отзывы Помощь по задачам

Чат FAQ Подписки

Истории успеха Поддержка Задачи-игры

Активности

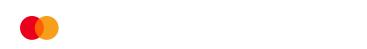


ОБУЧЕНИЕ

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА





"Программистами не рождаются" © 2022 JavaRush