



Пишем свой поток — обертку на System.in

Java Core (/quests/QUEST_JAVA_CORE)
8 уровень (/quests/lectures/?quest=QUEST_JAVA_CORE&level=8), 7 лекция (/quests/lectures/questcore.level08.lecture07)

ОТКРЫТА

— Привет, Амиго! Сегодня мы будем заниматься очень интересной работой — подменой потока ввода — System.in.

System.in — это простая статическая переменная типа InputStream, но присвоить ей новое значение просто так нельзя. Зато можно воспользоваться методом System.setIn().

Для начала нам нужно создать буфер, потом положить в него какие-то значения. Затем завернуть в класс, который умеет из этого буфера читать данные по протоколу InputStream.

Вот как это выглядит:

Код

```
1 public static void main(String[] args) throws IOException
2 {
3     //кладем данные в строку
4     StringBuilder sb = new StringBuilder();
5     sb.append("Lena").append('\n');
6     sb.append("Olya").append('\n');
7     sb.append("Anya").append('\n');
8     String data = sb.toString();
9
10    //Оборачиваем строку в класс ByteArrayInputStream
11    InputStream is = new ByteArrayInputStream(data.getBytes());
12
13    //подменяем in
14    System.setIn(is);
15
16    //вызываем обычный метод, который не подозревает о наших манипуляциях
17    readAndPrintLine();
18 }
19
20 public static void readAndPrintLine() throws IOException
21 {
22     InputStreamReader isr = new InputStreamReader(System.in);
23     BufferedReader reader = new BufferedReader(isr);
24
25     while (true)
26     {
27         String line = reader.readLine();
28         if (line == null) break;
29         System.out.println(line);
30     }
31     reader.close();
32     isr.close();
33 }
```

— Билаабо! Это самый интересный пример из всего, что я видел. Я и не знал, что так можно делать. Спасибо.

— Не за что, друг Амиго.

< (/quests/lectures/questcore.level08.lecture06)

×18 > (/quests/lectures/questcore.level08.lecture08)



0



0



0



0



0

+5

Комментарии (40)

популярные

новые

старые

Никита

Ruben Hakobyan 19 уровень

22 февраля, 15:09

**System.in – это простая статическая переменная типа InputStream**

точнее было бы так:

System.in – где "in" - это простая статическая переменная типа InputStream класса System.

Ответить

+3

Александр 21 уровень

22 января, 12:45



Добрый день!

Ребята, помогите пожалуйста, ничего понять не могу, для чего это нужно, есть system.in, для чего нам нужно присваивать какие либо значения. Если есть какие нибудь ссылки где разжуют все подробно от чего и для чего, буду благодарен!

Ответить

0

Redas Shuliakas 21 уровень

9 февраля, 11:28



Ответить

+3

Джонни 23 уровень

8 января, 10:18



Не понял как работает подмена... вот мои 2 примера. Как программа понимает, что нужно брать подменённый System.in? Я метод вынес вообще в другой пакет!

Пример 1. Всё работает.**Пример 2. Программа сломалась.**

Ответить

0

Beslan Bavizhev 36 уровень

1 февраля, 00:55



В System.in ты положил **однострочный** текст и пытаешься оттуда сканером считать построчно **два** раза. Поэтому программа выкидывает исключение. Оберни считывание в цикл и ошибки не будет. И для наглядного примера возьми многострочный String s = "Мама мыла раму\nПапа курил бамбук\nДед копал огород".

```
1 while (scanner.hasNext())
2     {
3         String line = scanner.nextLine();
4         System.out.println(line);
5     }
```

Ответить

+2

Roman Bogdanov 20 уровень, Днепр

11 декабря 2017, 16:33



если закомментировать

```
1 //      System.setIn(is);
```

Enter пустой строки не прекращает while. Кто знает почему? Там \n остаётся?

Ответить

0

Дмитро Карлович 21 уровень

24 октября 2017, 15:28



Пример понял только когда закинул в Идею. Только StringBuilder здесь лишний (ИМХО)

Ответить

+1

Святослав 23 уровень, Санкт-Петербург

23 сентября 2017, 10:15



Пример понял..Только не очень понял нафига это может пригодиться.Кто может подсказать?

Ответить

+3

Сергей Черник 33 уровень

5 октября 2017, 11:30

...

Можно, например, менять стандартные потоки вывода и ошибок на запись в файл! Вот тебе и логи)) (Но так делать не нужно, для логов нужно использовать либо SLF4J, либо log4j того же Apache. Но это уже совсем другая история ^-^)

Ответить

+4

Stanislav 21 уровень, Москва

16 января, 14:05

...

Ломать чужие проги

Ответить

+2

Волошин Виталий 27 уровень

16 августа 2017, 21:59

...

наглядно очень получается, если скопировать этот пример в Идею и в дебагере пошагово посмотреть на переменные, более-менее понятен процесс тогда

Ответить

+2

Sam 18 уровень, Нижний Новгород

5 июня 2017, 13:00

...

isr не надо закрывать. reader его сам закроет

Ответить

+3

Anonymous #1105523 21 уровень

9 апреля 2017, 12:17

...

Я один не знаю что за StringBuilder?

Ответить

0

Паша Шпаковский 32 уровень, Москва

10 апреля 2017, 21:25

...

я тоже не знаю

Ответить

0

Петр 22 уровень

17 апреля 2017, 20:09

...

Ну скачайте уже доки по Java API, и там ковыряйте. Всё равно без этого программистами не станете.

Ответить

+6

victor 30 уровень, Минск

19 апреля 2017, 01:38

...

здесь мы генерируем ввод данных с консоли, потом при вызове reader.readLine() метод натывается на символ перевода строки в потоке байт и возвращает строку, переходит на новую ступень цикла.

Ответить

+1

Galiya S M 19 уровень, Алматы

8 мая 2017, 09:04

...

<http://docs.oracle.com/javase/tutorial/java/data/index.html> лучше весь раздел прочитать

Ответить

0

Vikentsi 22 уровень, Минск

19 сентября 2017, 15:33

...

никакого смысла без идеального русского не вижу смысла читать.
И даже с идеальным тоже не вижу смысла читать как и целиком какую нибуть книгу по JAVA.
Читать надо по необходимости, необходимый кусок.
А необходимость это когда возникает вопрос, так что
Anonymous #1105523 ты знаешь что делать)

Ответить

+4

Orion 22 уровень

3 октября 2017, 01:29

...

Доводилось пользоваться StringBuilder-ом в некоторых предыдущих задачах, впервые повстречал на stackoverflow - так и выучил что это такое.
Stringbuilder позволяет нехило сэкономить ресурсы в определённых случаях.

Ответить

+2

fermopili 40 уровень

22 марта 2017, 17:52

...

Возникает логический вопрос - а как вернуться к стандартному вводу System.in с консоли, вопрос переключения от стандартного ввода данных к переопределенному

Ответить

0

Петр 22 уровень

17 апреля 2017, 20:06

...

залезь в исходный класс System, и посмотри значение переменной in по умолчанию. Это null.
Значит обнулить - это зафигачить туда обратно null.

Ответить

+4

kotdavinci 33 уровень, Москва

18 сентября 2017, 16:11

Как именно обнулить? Через System.setIn(null); не работает.

Ответить

0

Сергей Черник 33 уровень

5 октября 2017, 10:59

Значит так. Вопрос очень ценный. Ценный потому, что его нужно задавать ДО изменения системных потоков(Как in, так и out, err). потому, что легче всего его решить, зарезервировав переменные до изменения.

```
1 // сохраняем системное значение InputStream'a
2 InputStream in = System.in;
3 // изменяем поток!
4 System.setIn(something else);
5 // --какой-то код--
6
7 // сбрасываем
8 System.setIn(in);
```

Ответить

+21

[↺ Загрузить еще](#)

[jsh.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