



## Остановить бушующий поток: неофициальная версия

Java Core (/quests/QUEST\_JAVA\_CORE)  
6 уровень (/quests/lectures/?quest=QUEST\_JAVA\_CORE&level=6), 8 лекция (/quests/lectures/questcore.level06.lecture08)

ОТКРЫТА

— Привет, Амиго! Вот интересный вопрос, с которым ты уже столкнулся или столкнёшься в ближайшее время. А **как остановить запущенную нить?**

Допустим, пользователь отдал программе команду «загрузить файл из интернета». Главная нить создала для этого задания отдельную дочернюю нить, и передала ей объект, метод `run` которого содержит все необходимые действия для скачивания файла.

А тут пользователь – раз и передумал. Не хочет он качать этот файл. **Как отменить задание и остановить нить?**

— Да, как?

— **Никак.** Это и есть самый общий и самый правильный ответ. **Нить остановить нельзя, она может остановиться только сама.**

Но можно дать нити сигнал, сообщить ей каким-нибудь образом, что работу больше выполнять не нужно, и ей нужно завершиться. Так же, как главная нить завершается вместе с выходом из метода `main`, чтобы завершить дочернюю нить, она должна закончить выполнение метода `run`.

— И как это лучше всего сделать?

— Можно завести какую-нибудь переменную, например типа `boolean`. Если она `true` – нить работает. Если же она стала `false` – нить должна завершиться. Например, так:

### Код

### Описание

```
1 class Clock implements Runnable
2 {
3     public void run()
4     {
5         while (true)
6         {
7             Thread.sleep(1000);
8             System.out.println("Tik");
9         }
10    if (!ClockManager.isClockRun)
11        return;
12    }
13 }
14 }
```

Класс `Clock` (часы) будет вечно писать в консоль раз в секунду слово «Tik»

Если переменная `ClockManager.isClockRun` равна `false` – метод `run` завершится.

```
1 class ClockManager
2 {
3     public static boolean isClockRun = true;
4     public static void main(String[] args)
5     {
6         Clock clock = new Clock();
7         Thread clockThread = new Thread(clock);
8         clockThread.start();
9
10        Thread.sleep(10000);
11        isClockRun = false;
12    }
13
14 }
```

Главная нить, запускает дочернюю нить – часы, которая должна работать вечно.

Ждет 10 секунд и подает часам сигнал на завершение.

Главная нить завершает свою работу.

Нить часов завершает свою работу.

— А если у нас несколько нитей, что тогда?

— Тогда лучше завести такую переменную для каждой нити. Удобнее всего будет добавить ее прямо в класс. Можно добавить туда переменную boolean isRun. Хотя лучше добавить переменную boolean isCancel, которая будет принимать значение true, если задание отменено.

#### Код

#### Описание

```
1 class Clock implements Runnable
2 {
3     private boolean isCancel = false;
4
5     public void cancel()
6     {
7         this.isCancel = true;
8     }
9
10    public void run()
11    {
12        while (!isCancel)
13        {
14            Thread.sleep(1000);
15            System.out.println("Tik");
16        }
17    }
18 }
```

Класс Clock (часы) будет писать в консоль раз в секунду слово «Тик», пока переменная isCancel равна false.

Когда переменная isCancel станет равной true, метод run завершится.

```
1 public static void main(String[] args)
2 {
3     Clock clock = new Clock();
4     Thread clockThread = new Thread(clock);
5     clockThread.start();
6
7     Thread.sleep(10000);
8     clock.cancel();
9 }
```

Главная нить, запускает дочернюю нить – часы, которая должна работать вечно.

Ждет 10 секунд и отменяет задание, вызовом метода `cancel`.

Главная нить завершает свою работу.

Нить часов завершает свою работу.

— Буду знать, спасибо, Элли.

< (/quests/lectures/questcore.level06.lecture07)

×16 > (/quests/lectures/questcore.level06.lecture09)



0



0



0



0



0

+27

Комментарии (27)

популярные

новые

старые

Никита

...

[javarush.ru/](https://javarush.ru/). [G+ \(https://plus.google.com/114772402300089087607\)](https://plus.google.com/114772402300089087607). [Twitter \(https://twitter.com/javarush\\_ru\)](https://twitter.com/javarush_ru). [Facebook \(https://www.facebook.com/javarush\)](https://www.facebook.com/javarush)



Программистами не рождаются  
© 2018