



Бесплатная электронная книга

УЧУСЬ

random

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#random

.....	1
<b>1:</b> .....	<b>2</b>
.....	2
Examples.....	2
.....	2
- .....	2
.....	4

---

# Около

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [random](#)

It is an unofficial and free random ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official random.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

# глава 1: Начало работы со случайными

## замечания

В этом разделе приведен обзор того, что такое случайный случай, и почему разработчик может захотеть его использовать.

Следует также упомянуть любые крупные предметы в случайном порядке и ссылки на связанные темы. Поскольку документация для случайного является новой, вам может потребоваться создать начальные версии этих связанных тем.

## Examples

### Установка или настройка

Подробные инструкции по произвольной настройке или установке.

### Фишер-Йейтс перетасовывает

Также известен как «Перетасовка Кнута» и «Дурстенфельд-Фишер-Йейтс». Эта тасовка принимает массив из  $n$  элементов и перемешивает их. Алгоритм поистине случайен в том, что после перетасовки каждая перестановка массива одинаково вероятна.

В java:

```
public static void shuffle(E[] deck) {  
  
    //From the end, swap each card with a random card from the unswapped portion.  
    for(int i = deck.length - 1; i > 0; i--)  
    {  
        //Pick an element from [0,i], inclusive.  
        int chosenCard = (int) (Math.random() * (i + 1));  
  
        E temp = deck[i];  
        deck[i] = deck[chosenCard];  
        deck[chosenCard] = temp;  
    }  
}
```

Обратите внимание: необходимо, чтобы элемент замены включался из  $[0, i]$  включительно, а не  $[0, i)$  исключителен: в противном случае перестановки массива, где элементы остаются на месте, невозможны, что не является действительно случайным.

Предполагая, что предполагаемые случайные числа принимают  $O(1)$  для генерации, алгоритм работает на месте и принимает  $O(n)$  время и пространство. Массив, перетасованный таким образом, может использоваться для извлечения не повторяющихся

элементов в  $O(1)$  амортизированном времени на элемент.

```
E[] deck;
int drawIndex;

//Elements are taken from an index that advances.
public E drawUniqueCard()
{
    //Once all cards have been drawn, reshuffle the deck and draw from the top.
    if(drawIndex == deck.length)
    {
        shuffle(deck);
        drawIndex = 0;
    }
    //Pull the next card off the deck.
    return deck[drawIndex++];
}
```

Прочитайте Начало работы со случайными онлайн:

<https://riptutorial.com/ru/random/topic/9484/начало-работы-со-случайными>

---

## кредиты

S. No	Главы	Contributors
1	Начало работы со случайными	<a href="#">Community</a> , <a href="#">Mauve Ranger</a>