

( / )

# Generating Random Numbers in a Range in Java

Last modified: December 8, 2022

Written by: baeldung (<https://www.baeldung.com/author/baeldung>)

**Java** (<https://www.baeldung.com/category/java>)

**Random** (<https://www.baeldung.com/tag/random>)

---

**Get started with Spring 5 and Spring Boot 2, through the *Learn Spring* course:**

**> CHECK OUT THE COURSE** (</ls-course-start>)

---

## 1. Overview

In this tutorial, we'll explore different ways of generating random numbers within a range.

## Further reading:

### Generating Random Numbers in Java (/java-generating-random-numbers)

Learn different ways of generating random numbers in Java.

[Read more \(/java-generating-random-numbers\)](#) →

### Java - Random Long, Float, Integer and Double (/java-generate-random-long-float-integer-double)

Learn how to generate random numbers in Java - both unbounded as well as within a given interval.

[Read more \(/java-generate-random-long-float-integer-double\)](#) →

### Java - Generate Random String (/java-random-string)

Generate Bounded and Unbounded Random Strings using plain Java and the Apache Commons Lang library.

[Read more \(/java-random-string\)](#) →

## 2. Generating Random Numbers in a Range

### 2.1. *Math.random*

*Math.random* gives a random *double* value that is greater than or equal to 0.0 and less than 1.0.

Let's use the *Math.random* method to generate a random number in a given range *[min, max)*.

```
public int getRandomNumber(int min, int max) {  
    return (int) ((Math.random() * (max - min)) + min);  
}
```



Why does that work? Let's look at what happens when *Math.random* returns 0.0, which is the lowest possible output:

```
0.0 * (max - min) + min => min
```



So, the lowest number we can get is *min*.

Since 1.0 is the exclusive upper bound of *Math.random*, this is what we get:

```
1.0 * (max - min) + min => max - min + min => max
```



Therefore, the exclusive upper bound of our method's return is *max*.

In the next section, we'll see this same pattern repeated with *Random#nextInt*.

## 2.2. *java.util.Random.nextInt*

We can also use an instance of *java.util.Random* to do the same.

Let's make use of the *java.util.Random.nextInt* method to get a random number:

```
public int getRandomNumberUsingNextInt(int min, int max) {  
    Random random = new Random();  
    return random.nextInt(max - min) + min;  
}
```



The *min* parameter (the origin) is inclusive, whereas the upper bound *max* is exclusive.

## 2.3. *java.util.Random.ints*

The *java.util.Random.ints* method returns an *IntStream* of random integers.

So, we can utilize the `java.util.Random.ints` method and return a random number:

```
public int getRandomNumberUsingInts(int min, int max) {  
    Random random = new Random();  
    return random.ints(min, max)  
        .findFirst()  
        .getAsInt();  
}
```



Here as well, the specified origin *min* is inclusive, and *max* is exclusive.

### 3. Conclusion

In this article, we saw alternative ways of generating random numbers within a range.

Code snippets, as always, can be found over on GitHub

(<https://github.com/eugenp/tutorials/tree/master/core-java-modules/core-java-numbers-3>).

**Get started with Spring 5 and Spring Boot 2,  
through the *Learn Spring* course:**

**>> CHECK OUT THE COURSE (/ls-course-end)**



## Learning to build your API **with Spring?**

**Download the E-book** (</rest-api-spring-guide>)

---

Comments are closed on this article!

### COURSES

[ALL COURSES \(/ALL-COURSES\)](#)

[ALL BULK COURSES \(/ALL-BULK-COURSES\)](#)

[ALL BULK TEAM COURSES \(/ALL-BULK-TEAM-COURSES\)](#)

[THE COURSES PLATFORM \(HTTPS://COURSES.BAELDUNG.COM\)](https://courses.baeldung.com)

### SERIES

[JAVA "BACK TO BASICS" TUTORIAL \(/JAVA-TUTORIAL\)](#)

[JACKSON JSON TUTORIAL \(/JACKSON\)](#)

[APACHE HTTPCLIENT TUTORIAL \(/HTTPCLIENT-GUIDE\)](#)

[REST WITH SPRING TUTORIAL \(/REST-WITH-SPRING-SERIES\)](#)

[SPRING PERSISTENCE TUTORIAL \(/PERSISTENCE-WITH-SPRING-SERIES\)](#)

[SECURITY WITH SPRING \(/SECURITY-SPRING\)](#)

[SPRING REACTIVE TUTORIALS \(/SPRING-REACTIVE-GUIDE\)](#)

## ABOUT

[ABOUT BAELDUNG \(/ABOUT\)](#)

[THE FULL ARCHIVE \(/FULL\\_ARCHIVE\)](#)

[EDITORS \(/EDITORS\)](#)

[JOBS \(/TAG/ACTIVE-JOB/\)](#)

[OUR PARTNERS \(/PARTNERS\)](#)

[PARTNER WITH BAELDUNG \(/ADVERTISE\)](#)

[TERMS OF SERVICE \(/TERMS-OF-SERVICE\)](#)

[PRIVACY POLICY \(/PRIVACY-POLICY\)](#)

[COMPANY INFO \(/BAELDUNG-COMPANY-INFO\)](#)

[CONTACT \(/CONTACT\)](#)