Generating random numbers in Java

Difficulty Level : Basic • Last Updated : 28 Feb, 2023

Java provides three ways to generate random numbers using some built-in methods and classes as listed below:

- **java.util.Random** class
- **Math.random** method : Can Generate Random Numbers of double type.
- **ThreadLocalRandom** class

1) java.util.Random

- For using this class to generate random numbers, we have to first create an instance of this class and then invoke methods such as nextInt(), nextDouble(), nextLong() etc using that instance.
- We can generate random numbers of types integers, float, double, long, booleans using this class.
- We can pass arguments to the methods for placing an upper bound on the range of the numbers to be generated. For example, nextInt(6) will generate numbers in the range 0 to 5 both inclusive.

Java

```
// A Java program to demonstrate random number generation
// using java.util.Random;
import java.util.Random;
```

```
        Random rand = new Random();

        // Generate random integers in range 0 to 999
        int rand_int1 = rand.nextInt(1000);
        int rand_int2 = rand.nextInt(1000);

        // Print random integers
        System.out.println("Random Integers: "+rand_int1);
        System.out.println("Random Integers: "+rand_int2);

        // Generate Random doubles
        double rand_dub1 = rand.nextDouble();
        double rand_dub2 = rand.nextDouble();

        // Print random doubles
        System.out.println("Random Doubles: "+rand_dub1);
        System.out.println("Random Doubles: "+rand_dub2);
    }
}
```

Output

```
Random Integers: 618
Random Integers: 877
Random Doubles: 0.11981638980670772
Random Doubles: 0.7288425427367139
```

2) Math.random()

The class Math contains various methods for performing various numeric operations such as, calculating exponentiation, logarithms etc. One of these methods is random(), this method returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0. The returned values are chosen pseudo randomly. This method can only generate random numbers of type Doubles. Below program explains how to use this method:

Java

```
public class generateRandom
{
    public static void main(String args[])
    {
        // Generating random doubles
        System.out.println("Random doubles: " + Math.random());
        System.out.println("Random doubles: " + Math.random());
    }
}
```

Output

```
Random doubles: 0.40748894116045375
Random doubles: 0.006683607229094002
```

3) java.util.concurrent.ThreadLocalRandom class

This class is introduced in java 1.7 to generate random numbers of type integers, doubles, booleans etc. Below program explains how to use this class to generate random numbers:

Java

```
public class generateRandom
{
    public static void main(String args[])
    {
        // Generate random integers in range 0 to 999
        int rand_int1 = ThreadLocalRandom.current().nextInt();
        int rand_int2 = ThreadLocalRandom.current().nextInt();

        // Print random integers
        System.out.println("Random Integers: " + rand_int1);
        System.out.println("Random Integers: " + rand_int2);

        // Generate Random doubles
        double rand_dub1 = ThreadLocalRandom.current().nextDouble();
        double rand_dub2 = ThreadLocalRandom.current().nextDouble();

        // Print random doubles
        System.out.println("Random Doubles: " + rand_dub1);
        System.out.println("Random Doubles: " + rand_dub2);

        // Generate random booleans
        boolean rand_bool1 = ThreadLocalRandom.current().nextBoolean();
        boolean rand_bool2 = ThreadLocalRandom.current().nextBoolean();

        // Print random Booleans
        System.out.println("Random Booleans: " + rand_bool1);
        System.out.println("Random Booleans: " + rand_bool2);
    }
}
```

Output

```
Random Integers: -2106603442
Random Integers: 1894823880
Random Doubles: 0.6161052280172054
Random Doubles: 0.8418944588752132
Random Booleans: false
Random Booleans: true
```

To generate Random numbers with specific ranges, There 2 different ways to do

1. Using Random Class

Here is formula to generate a random numbers with a specific range, where min and max are our lower and higher limit of number.

Random rand = new Random();
int randomNum = rand.nextInt(max - min + 1) + min;

Java

```
import java.io.*;
import java.util.*;

class GFG {
    public static void main (String[] args) {
        Random rand = new Random();
          int max=100,min=50;
          System.out.println("Generated numbers are within "+min+" to "+max);
          System.out.println(rand.nextInt(max - min + 1) + min);
          System.out.println(rand.nextInt(max - min + 1) + min);
          System.out.println(rand.nextInt(max - min + 1) + min);
    }
```

```
Generated numbers are within 50 to 100
58
87
55
```

Time Complexity: It has a time complexity of O(1)
Auxiliary Space: O(1) requires constant space.

2. Using Math.random() Method

Here is the formula to generate a random number with specific range, where min and max are our lower and higher limit of number:

*int randomNum = min + (int)(Math.random() * ((max – min) + 1));*

Java

```java
/*package whatever //do not write package name here */

import java.io.*;
import java.util.*;

class GFG {
    public static void main (String[] args) {
        int max=100,min=50;
          System.out.println("Generated numbers are within "+min+" to "+max);
           System.out.println(min + (int)(Math.random() * ((max - min) + 1)));
          System.out.println(min + (int)(Math.random() * ((max - min) + 1)));
          System.out.println(min + (int)(Math.random() * ((max - min) + 1)));
    }
}
```

Output

```
99
77
```

Time Complexity: It has a time complexity of O(1)

Auxiliary Space: O(1) requires constant space.

Related Articles

1. Random vs Secure Random numbers in Java
2. How to Create a Random Graph Using Random Edge Generation in Java?
3. Generating Password and OTP in Java
4. Java Program to Generate Random Numbers Using Multiply with Carry Method
5. Generate Random Numbers Using Middle Square Method in Java
6. Java.util.Random class in Java
7. Java.util.Random.nextInt() in Java
8. Image Processing in Java - Creating a Random Pixel Image
9. Java Math random() method with Examples
10. A Java Random and StringBuffer Puzzle