

Mathematical Operations in Java – Random Numbers

Most programming languages, including Java, have some sort of *random number generator*. Some languages provide multiple options for generating random numbers, but most provide a single option that must be manipulated by the programmer to produce the desired result.

In Java, the random number generation is part of the `Math` class, and it is invoked using the `Math.random` method, which has the following signature:

```
public double random()
```

This method returns a double value x where $0 \leq x < 1$ (i.e., the numbers from 0.00000 to 0.99999, but does not include 1).

Example 1 – One common use of random numbers in programming is to simulate the throwing of a *fair die*, which is a six-sided die with an equal probability of getting a 1, 2, 3, 4, 5, or 6. We can do so by performing the following operations:

| Operation | Result | |
|--|-----------|------------|
| | Low Value | High Value |
| (a) Call the <code>Math.random</code> method to generate a random value x , where $0 \leq x < 1$ | 0.000000 | 0.999999 |
| (b) Multiply this value by 6 (since there are six possible sides on the die) to get the interval $0 \leq x < 6$. | 0.000000 | 5.999999 |
| (c) Cast this result as an integer, which chops off any fractional (decimal) part. Our value is now 0, 1, 2, 3, 4, or 5. | 0 | 5 |
| (d) Adding 1 will give us the desired range of values, from 1 to 6. | 1 | 6 |

All of these operations can be performed in one step with the statement

```
int dieRoll = (int)( 6 * Math.random() ) + 1;
```

 ↑ ↑ ↑ ↑
(c) (b) (a) (d)

All random number generation follows these basic steps, where we:

- (a) generate a random number between 0.00000 and 0.99999,
- (b) multiply by the number of possible results we desire,
- (c) optional: cast the result as an integer (if we want integer values),
- (d) add (or subtract) some number to shift our random numbers to the desired range.