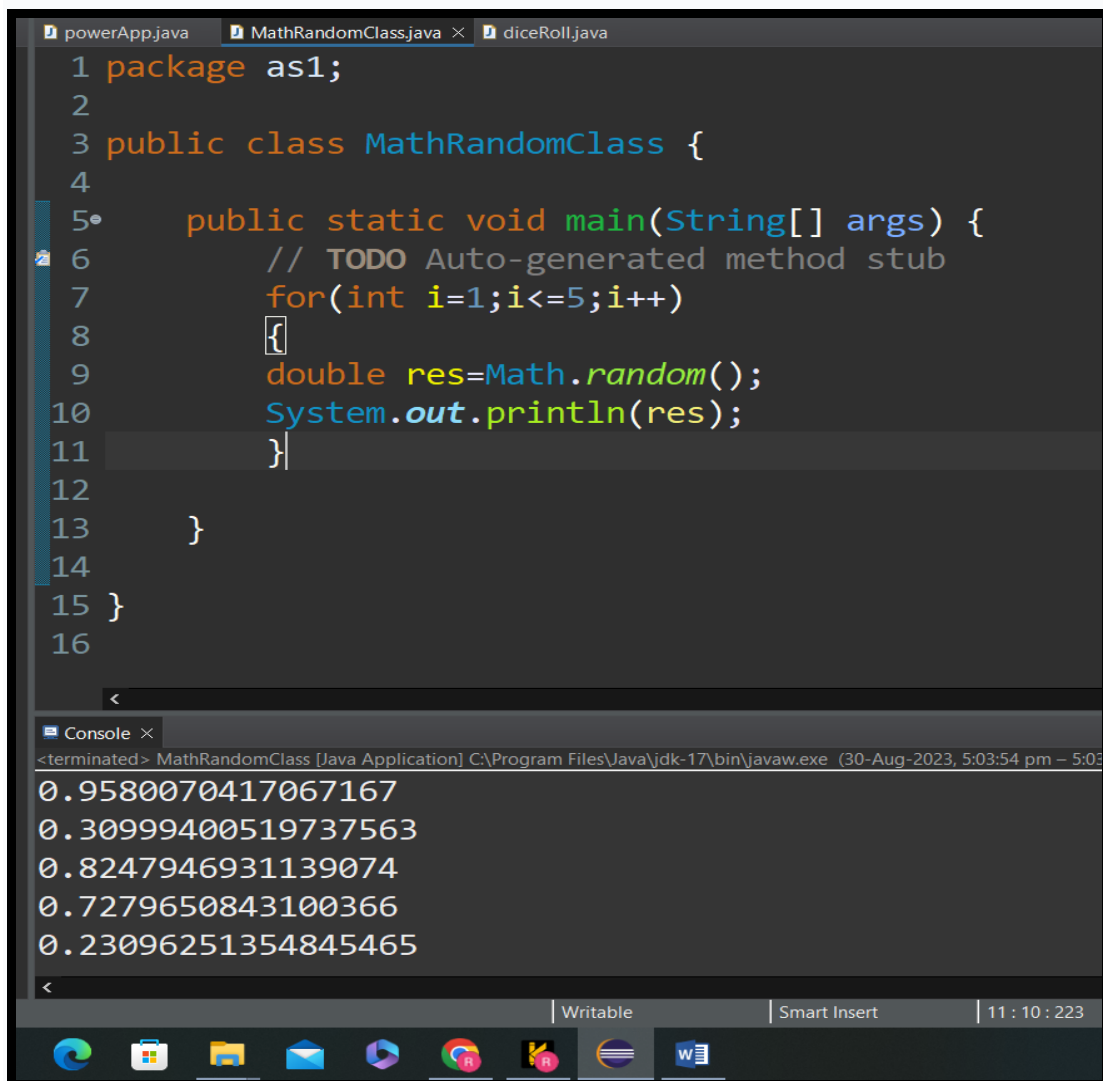


Math.random()

We frequently need to create random numbers while developing applications in Java programming. Several applications (for instance, a **dice roller** app) often need to create random numbers. This is where we use `math.random`.

The `java.lang.Math.random()` method returns a random double type number greater than or equal to 0.0 and less than 1.0. When this method is first called, it creates a single new random-number generator, exactly as if by the expression `new java.util.Random`.

Example:



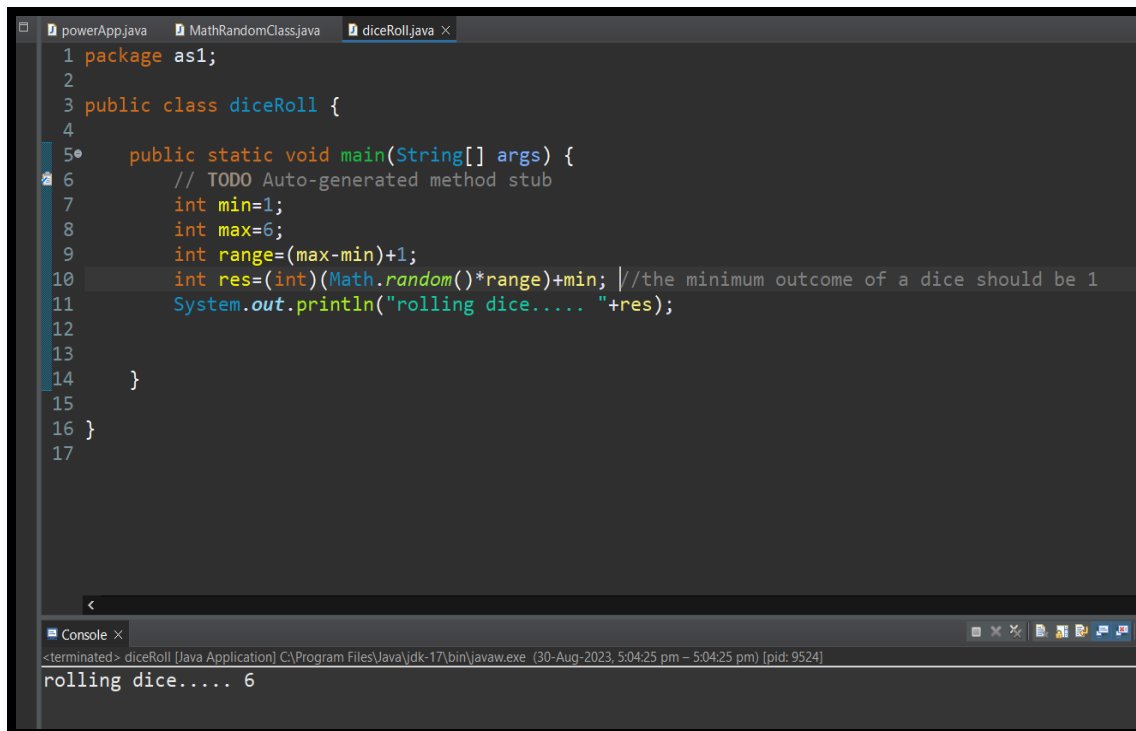
```
1 package as1;
2
3 public class MathRandomClass {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         for(int i=1;i<=5;i++)
8         {
9             double res=Math.random();
10            System.out.println(res);
11        }
12    }
13 }
14
15 }
16
```

Console

```
<terminated> MathRandomClass [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (30-Aug-2023, 5:03:54 pm - 5:03:54 pm)
0.9580070417067167
0.30999400519737563
0.8247946931139074
0.7279650843100366
0.23096251354845465
```

`Math.random()` method in `Math` class has been simply used in a loop to print random Double numbers ranging between 0.0 and 1.0.

Use case: Writing a program to make a dice which would yield numbers from 1 to 6 randomly in each throw.

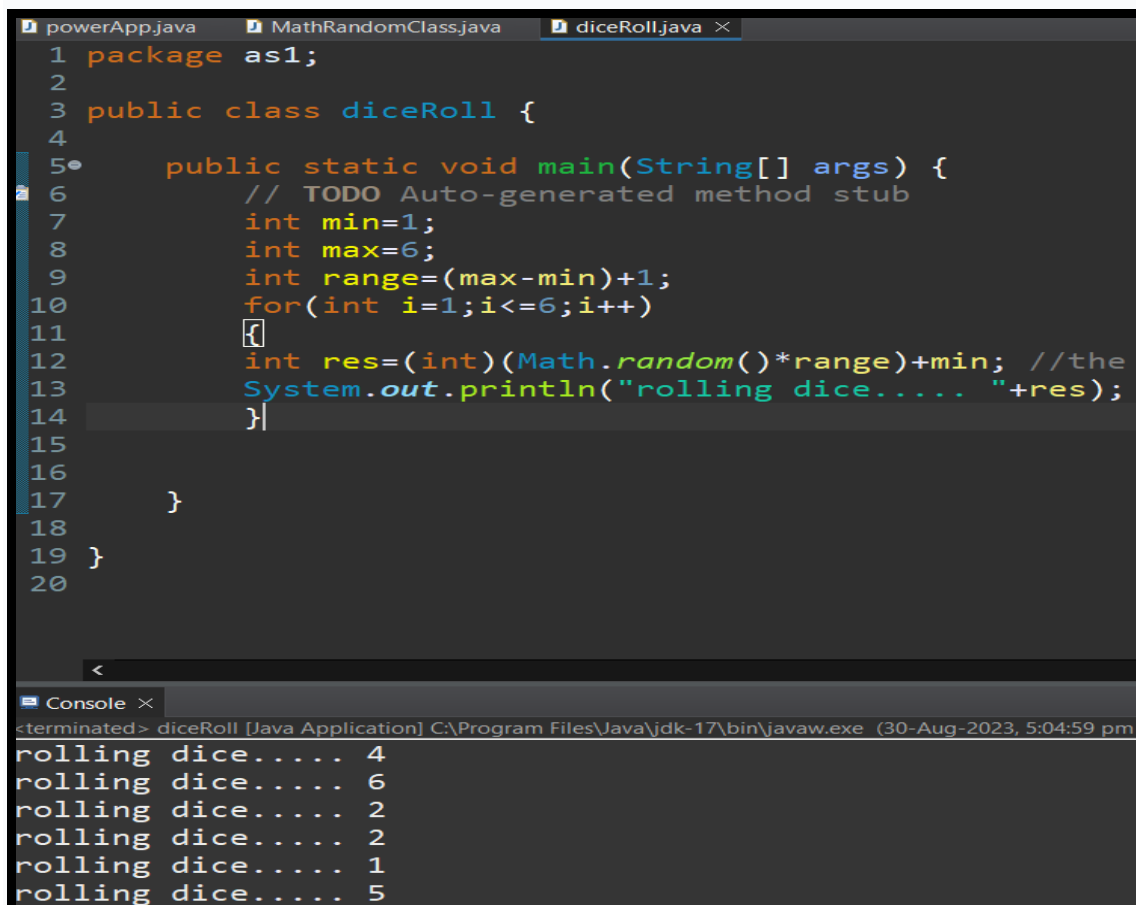


The screenshot shows an IDE with three tabs: `powerApp.java`, `MathRandomClass.java`, and `diceRoll.java`. The `diceRoll.java` file contains the following code:

```
1 package as1;
2
3 public class diceRoll {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int min=1;
8         int max=6;
9         int range=(max-min)+1;
10        int res=(int)(Math.random()*range)+min; //the minimum outcome of a dice should be 1
11        System.out.println("rolling dice..... "+res);
12
13    }
14 }
15
16 }
17
```

The console output at the bottom shows: `rolling dice..... 6`.

Putting the code block in a loop to yield random numbers (1 to 6) 6 times at once:



The screenshot shows the same IDE with the `diceRoll.java` file updated to include a loop for six rolls. The code is as follows:

```
1 package as1;
2
3 public class diceRoll {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int min=1;
8         int max=6;
9         int range=(max-min)+1;
10        for(int i=1;i<=6;i++)
11        {
12            int res=(int)(Math.random()*range)+min; //the
13            System.out.println("rolling dice..... "+res);
14        }
15    }
16 }
17
18 }
19 }
20
```

The console output at the bottom shows six lines of results: `rolling dice..... 4`, `rolling dice..... 6`, `rolling dice..... 2`, `rolling dice..... 2`, `rolling dice..... 1`, and `rolling dice..... 5`.