‘MATH’ AND ‘RANDOM’ CLASSES IN JAVA

Jayasimha Yalamakuru

What is a Math Class?

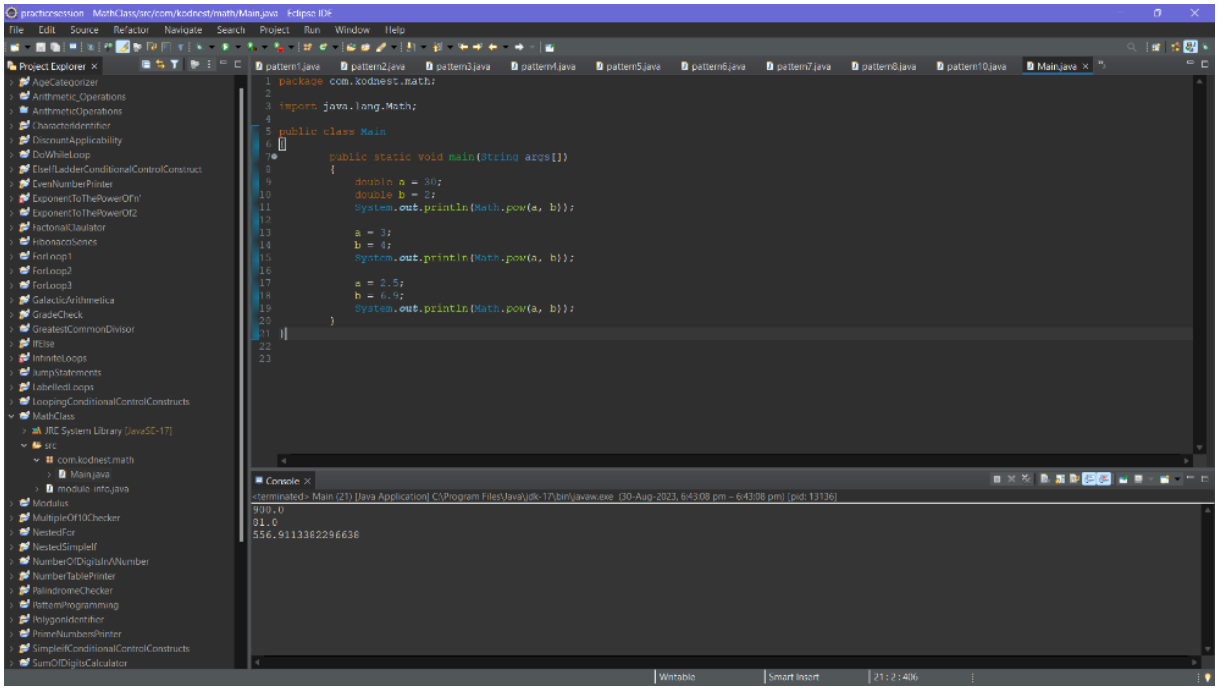
The java.lang.Math.pow() is the class used to calculate a number raise to the power of some other number. This function accepts two parameters and returns the value of first parameter raised to the second parameter. There are some special cases as listed below.

- If the second parameter is positive or negative zero then the result will be 1.0.
- If the second parameter is 1.0 then the result will be same as that of the first parameter.
- If the second parameter is NaN then the result will also be NaN.
- The function java.lang.Math.pow() always returns a double datatype.

This method is an inbuilt method of the “Math” class which is also inbuilt in nature.

Syntax:

public static double pow(double a, double b)
Parameter:
a : this parameter is the base
b : this parameter is the exponent.
Return :
This method returns a^b.



- Executable example code for math class

Random method in Math Class

The java.lang.Math.random() method returns a pseudorandom 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 pseudorandom-number generator, exactly as if by the expression new java.util.Random.

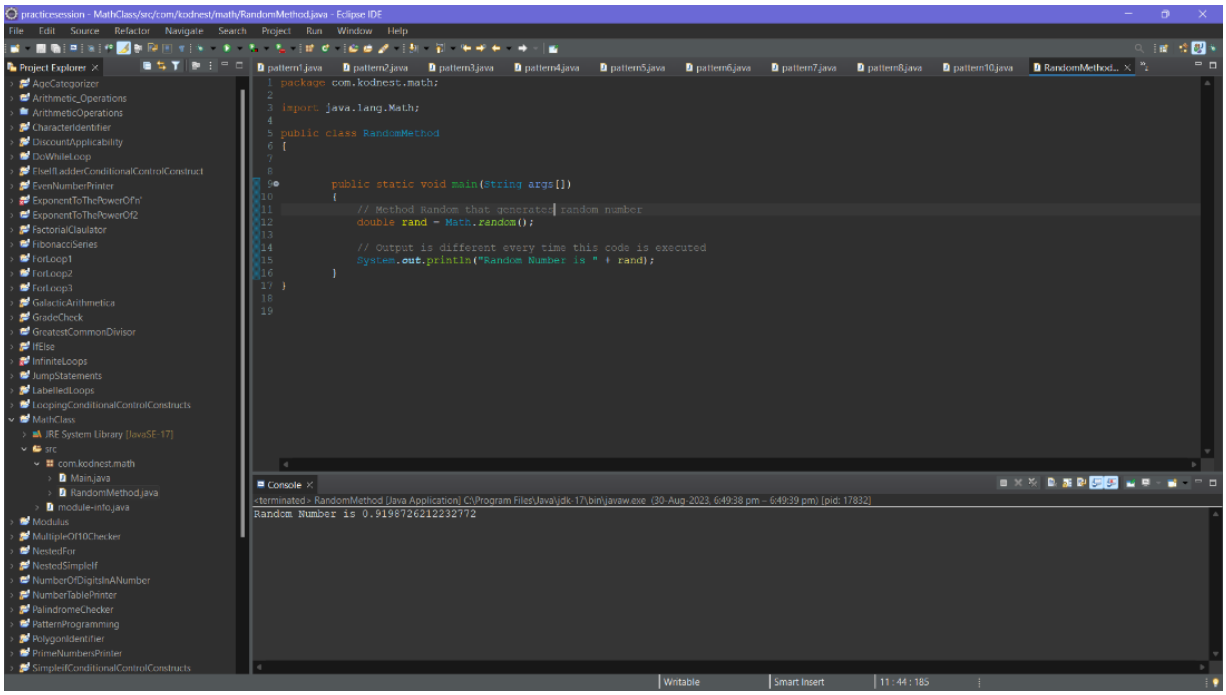
Declaration of Java Math random()

Below is the declaration of java.lang.Math.random() method is mentioned below:

public static double random()

Return Type

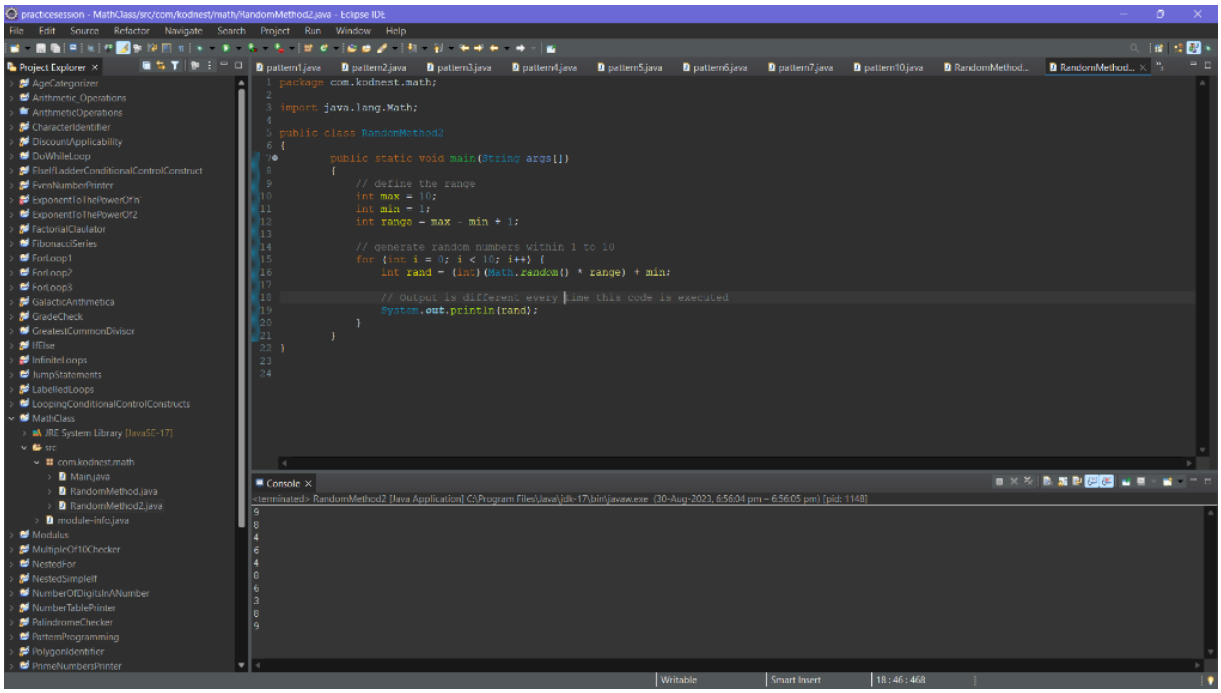
This method returns a pseudorandom double greater than or equal to 0.0 and less than 1.0.



- Executable example program for ‘random’ method in ‘Math’ class

Random method in a Range:

To get random integer numbers from a given fixed range, we take a min and max variable to define the range for our random numbers, both min and max are inclusive in the range.



- Executable example program for ‘random’ method in a range of integers

RANDOM CLASS

Random class is used to generate pseudo-random numbers in java. This class provides various method calls to generate different random data types such as float, double, int.

Constructors:

- Random(): Creates a new random number generator
- Random(long seed): Creates a new random number generator using a single long seed

Declaration:

public class Random
extends Object
implements Serializable

Methods present in "Random" Class: -

1. **java.util.Random.doubles():** Returns an effectively unlimited stream of pseudo random double values, each between zero (inclusive) and one (exclusive)

   Syntax:
   public DoubleStream doubles()

   Returns:
   a stream of pseudorandom double values

2. **java.util.Random.ints():** Returns an effectively unlimited stream of pseudo random int values

   Syntax:
   public IntStream ints()

   Returns:

a stream of pseudorandom int values

3. **java.util.Random.longs():** Returns an effectively unlimited stream of pseudo random long values

Syntax:
public LongStream longs()

Returns:
a stream of pseudorandom long values

4.next(int bits): java.util.Random.next(int bits) Generates the next pseudo random number

Syntax:
protected int next(int bits)

Parameters:
bits - random bits

Returns:
the next pseudo random value from this
random number generator's sequence

5.java.util.Random.nextBoolean(): Returns the next pseudo random, uniformly distributed boolean value from this random number generator’s sequence

Syntax:
public boolean nextBoolean()

Returns:
the next pseudorandom, uniformly distributed boolean value from this random number generator's sequence

6.java.util.Random.nextBytes(byte[] bytes) : Generates random bytes and places them into a user-supplied byte array

Syntax:

public void nextBytes(byte[] bytes)

Parameters:

bytes - the byte array to fill with random bytes

Throws:

NullPointerException - if the byte array is null

7.java.util.Random.nextDouble(): Returns the next pseudo random, uniformly distributed double value between 0.0 and 1.0 from this random number generator's sequence

Syntax:

public double nextDouble()

Returns:

the next pseudo random, uniformly distributed double value between 0.0 and 1.0 from this random number generator's sequence

8.java.util.Random.nextFloat(): Returns the next pseudo random, uniformly distributed float value between 0.0 and 1.0 from this random number generator's sequence

Syntax:

public float nextFloat()

Returns:

the next pseudorandom, uniformly distributed float value between 0.0 and 1.0 from this random number generator's sequence

9.java.util.Random.nextInt(): Returns the next pseudorandom, uniformly distributed int value from this random number generator's sequence

Syntax:

public int nextInt()

Returns:
the next pseudorandom, uniformly distributed int value from this random number generator's sequence

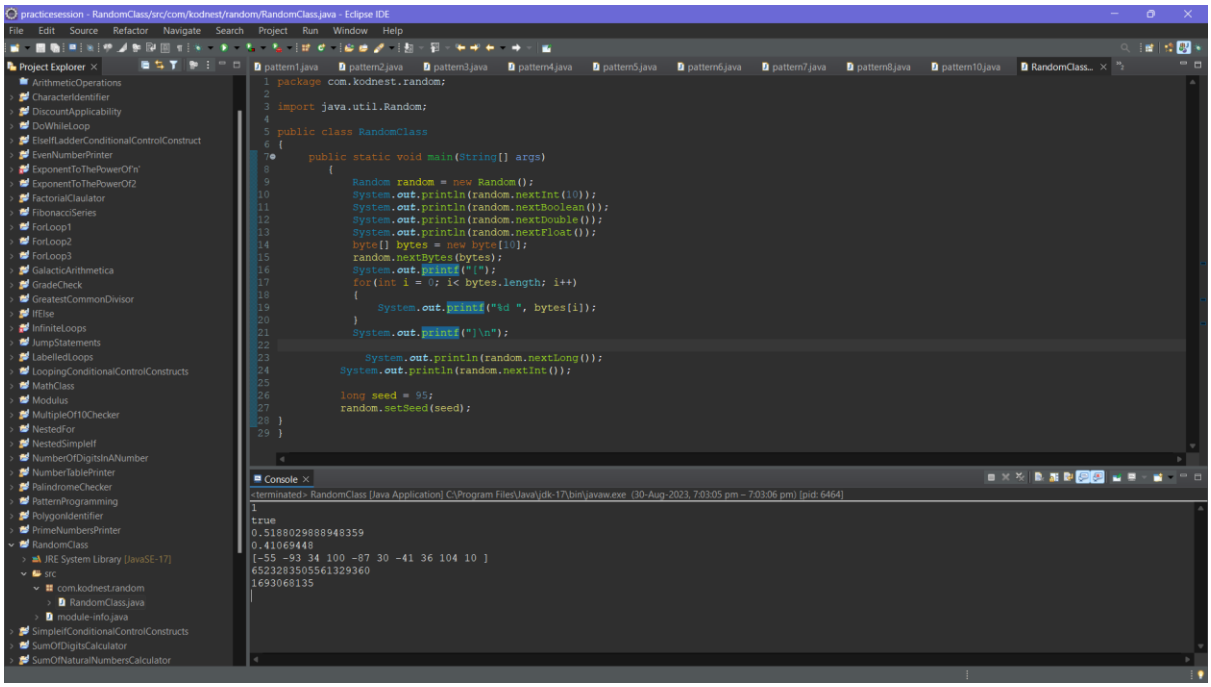
10.java.util.Random.nextLong(): Returns the next pseudorandom, uniformly distributed long value from this random number generator's sequence

Syntax:
public long nextLong()

Returns:
the next pseudorandom, uniformly distributed long value from this random number
generator's sequence

Above are the some of the methods present in the 'Random' class.

Example:



- Executable example program of random class