

Hands-on Experiment # 3 : Worksheet

Section _____ 1 _____ Date _____ 3 February 2020 _____

No more than 3 students per one submission of this worksheet.

Student ID _____ Name _____

Student ID _____ Name _____

Student ID _____ Name _____

This time, you are provided with a new “Java101.class” as well as its source code.
The file “Java101.class” provides a new method called *Java101.typeOf()* which can be used to determine the data type of its input value.

Part A: Indicate Resulting Data Types of Expressions with Multiple Data Types

1. Determine the data type as well as the value of each expression in the table below.
2. Write a Java program to verify that your answers (both the data types and the values) are correct.
3. Capture a screenshot showing the output of your program.

Expression	Data Type	Value
20.0/2.5	double	8.0
“3”+2+1	String	321
5+“6”+7	String	567
4/6*3	Int	0
3.0/4*4	Double	3.0
(short)1.5	Short	1
3+2.0F+4.0	Double	9.0
4-6==7-9	Boolean	True
(int)(5-6.5)==1-2	Boolean	True

List the source code of your program below.

```
public class Part_A {  
    public static void main(String[] args) {  
        Java101 j = new Java101();  
        System.out.println(j.typeOf("3" + 2 + 1));  
    }  
}
```

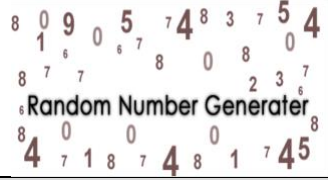
```
System.out.println("3" + 2 + 1);
System.out.println(j.typeOf(5 + "6" + 7));
System.out.println(5 + "6" + 7);
System.out.println(j.typeOf(4 / 6 * 3));
System.out.println(4 / 6 * 3);
System.out.println(j.typeOf(3.0 / 4 * 4));
System.out.println(3.0 / 4 * 4);
System.out.println(j.typeOf((short) 1.5));
System.out.println((short) 1.5);
System.out.println(j.typeOf((3 + 2.0F + 4.0)));
System.out.println(3 + 2.0F + 4.0);
System.out.println(j.typeOf((int) (5 - 6.5) == 1 - 2));
System.out.println((int) (5 - 6.5));
}
}
```

Insert the screenshot below.

```
String
321
String
567
int
0
double
3.0
short
1
double
9.0
boolean
-1
```

Part B: Random Number Generator

In this part, you will write a “random generator” program.



A random number generator that generates integer number from 1 to 10 and normally distributed with an average of 5 and standard deviation of 2.

An execution of the program generate an integer number.

The file “Java101.class” also provides a method called `Java101.showRomanNumber()` which takes an `int` value as its input. When invoked, the method shows a Roman numeric associated with the input value.

1. Study the following two methods: `java.util.Random()` and `java.lang.Math.ceil()` from <http://docs.oracle.com/javase/8/docs/api/index.html>
2. Come up with a Java expression using `Random()`, `Math.ceil()`, and an appropriate cast operator so that the expression produces a random `int` value in the range of 1 to 10 that is normally distributed with an average of 5 and standard deviation of 2.
3. Write a Java program that performs the number generation described. Name the program appropriately.
4. List the source code as well as screenshots of the program.

Hint: `nextGaussian()` returns the next pseudorandom, Gaussian ("normally") distributed double value with mean 0.0 and standard deviation 1.0 from this random number generator's sequence.

Show your Java expression in step 2 here.

```
Random rn = new Random();
double randomNum = Math.ceil(rn.nextGaussian() * 2 + 5)
```

Also, list the source code of the program you wrote below.

```
import java.util.Random;
import java.lang.Math;

public class Num_Gen {
    public static void main(String[] args) {
        Random rn = new Random();
        for (int i = 0; i < 30; i++) {
            double randomNum = Math.ceil(rn.nextGaussian() * 2 + 5);
            System.out.print(Math.max(1, Math.min(randomNum, 10)) + ",");
        }
    }
}
```

Insert the screenshots below.

```
~/Downloads/L03 ➔ cd "/Users/boss_wt/Downloads/L03/" && javac Num_Gen.java && java Num_Gen
6.0,6.0,8.0,6.0,5.0,5.0,3.0,6.0,5.0,7.0,7.0,9.0,7.0,8.0,5.0,6.0,3.0,4.0,4.0,4.0,4.0,6.0,10.0,4.0,6.0,3.0,6.0,5.0,4.0,3.0,
```

Part C: Test the Random Number Generator

1. Run the program you wrote in Part B 30 times. Note the result of each generation in the table below.

Trial #	Result	Trial #	Result	Trial #	Result
1	8.0	11	9.0	21	7.0
2	10.0	12	6.0	22	7.0
3	4.0	13	3.0	23	1.0
4	5.0	14	8.0	24	6.0
5	5.0	15	2.0	25	6.0
6	4.0	16	9.0	26	4.0
7	7.0	17	5.0	27	5.0
8	7.0	18	8.0	28	7.0
9	5.0	19	9.0	29	5.0
10	4.0	20	5.0	30	5.0

2. Calculate mean and standard deviation of generated numbers.

Mean (Average)	= 5.86666667
Standard Deviation (SD)	= 2.109238936