JAVA LAB1
PGP - COMP1039

JAVA INSTALL

- Check if Java is already installed on your laptop.
  - 1. Open the Command Prompt (cmd.exe) or Terminal (in Mac OS).
  - 2. Use the following command to check if Java is already installed

    java -version

  - 3. If Java is already installed, version information will display
  - *java version "11.0.1" 2018-10-16 LTS*
    Java(TM) SE Runtime Environment 18.9 (build 11.0.1+13-LTS)
    Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.1+13-LTS, mixed mode)
  - *4*. If not, you can download JDK from
  - *https://www.oracle.com/java/technologies/java-se-glance.html*
  - Please choose the standard edition (SE) version 8 or later.

FIRST PROGRAM

```
class HelloWorld {
	public static void main(String[] args) {
		System.out.println(”Hello World”);
	}
}
```

- 1. TEST1 try deleting each of ‘public’, ‘static’, ‘void’, ‘main’ and ‘args’ keywords and see if it still compiles and runs. What do you learn from the error messages?
- 2. TEST2 what happens if we use the print method instead?

EXAMPLE 2

```
class Example2{
	public static void main(String[] args){
		int var1;
		int var2;
		var1 = 1024;
		System.out.println("var1 contains " + var1);
		var2 = var1 / 3;
		System.out.println("var2 contains var1 / 3:  " + var2);
	}
}
```

// 1. Try different data types, double, float and long.

// 2. Try var2 = var2 / 3.

EXAMPLE 3

```
class Example3{
	public static void main(String[] args){
		int w = 10;
		double x = 10.0;
		System.out.println("Original value of w: " + w);
		System.out.println("Original value of x: " + x);
		System.out.println();
		w = w / 4;
		x = x / 4;
		System.out.println("w after division: " + w);
		System.out.println("x after division: " + x);

	}
}
// 1. Try w = 11, w = 0.
// 2. Try x = 0.0, x = -0.1
// 3. Test if it is ok to use 4.0 rather than 4, i.e., w / 4.0
```

FORDEMO

```
class ForDemo{
	public static void main(String[] args){
		int count;
		for(count = 0; count < 5; count = count+1)
			System.out.println("This is count:" + count);
		System.out.println("Done!");
	}
}
```

// 1. Test if it is ok to use double value, i.e., double count.
// 2. Try use While and Do While to produce the same result.

IFDEMO2

Debug the following code:

```
class IfDemo2{
	public static void main(String[] args){
		Double a, b;
		a = 3.0;
		b = 3.0;
		if ( a < b )
			System.out.println(“a is less than b”);
		else if ( a == b )
			System.out.println(“a is the same as b”);
		else
			System.out.println(“b is less than a”);
	}
}
```

FLOATPOUNDS

Debug the following code:

```
class FloatPounds{
	public static void main(String[] args){
		float twentyPence = 0.2f;
		System.out.println("20 pence = " + twentyPence);
		for(int i = 0; i < 1000; i++){
			twentyPence += 0.2f;
		}
		System.out.println("total = " + twentyPence);
	}
}
```

// 1. Test what is the result of 0.2f * 1000

JAVATEST

Debug the following code:

```
class JavaTest{
	public static void main(String[] args){
		short s = 0;
		int x = 7;
		int y = 8;
		int z = 50000;
		s +=z;
		System.out.println(x + y + s);
		System.out.println(x + "" + y + s);
	}
}
```

1. *Try s = s + z instead of s+=z*
2. *Delete s+=z, and then try* System.out.println(x + y + (s+z));

IF-ELSE

Task:

Given an integer n, perform the following conditional actions:

If n is odd, print Weird

If n is even and in the inclusive range of 2 to 5 (i.e., [2-5]), print Not Weird

If n is even and in the inclusive range of 6 to 20 (i.e., [6-20]), print Weird

If n is even and greater than 20, print Not Weird

- **Constraints**
- 1< n < 100
- **Output Format**
- Print Weird if the number is weird; otherwise, print Not Weird.

FOR LOOP

- **Task**
  Given an integer n , print its first 10 multiples. Each multiple n x i (where $1 \leq i \leq 10$) should be printed on a new line in the form: n x i = result.
- **Input Format**
- A single integer, .
- **Constraints**
- $2 \leq n \leq 20$
- **Output Format**
- Print 10 lines of output; each line i contains the result of n x i in the form:
  n x i = result.

LOOPS

- **Task**
  We use the integers a, b and n to create the following series:
- $(a + 2^0 \times b), (a + 2^0 \times b + 2^1 \times b), \ldots, (a + 2^0 \times b + 2^1 \times b + \ \ldots \ + 2^{n-1} \times b)$
- **Input Format**
- Three integers: a, b and n
- **Constraints**
- $0 \leq a, b \leq 50, 1 \leq n \leq 15$
- **Output Format**
- Print the corresponding series. For example, if a = 0, b = 2, n = 10, the output should be:
- 2 6 14 30 62 126 254 510 1022 2046