FULL STACK DEVELOPMENT – WORKSHEET 3

OMKAR PANDEY

BATCH = FS2308

DATE:30/9/2023

Q1. Which one of the following is not a Java feature?

Ans. Use of pointers

Q2. Which of these cannot be used for a variable name in Java?

Ans. Keyword

Q3. Which of the following is a superclass of every class in Java?

Ans. Object class

Q4. Which one is a valid declaration of a boolean?

Ans. Boolean b3 = false;

Q5. Which is the modifier when there is none mentioned explicitly?

Ans. Default

Q6. All the variables of interface should be?

Ans. public, static and final

Q7. Which of these data types is used to store command line arguments?

Ans. String

Q8.How many arguments can be passed to main()?

Ans. Infinite

Q9.What will be the output of the following Java program, Command line execution is done as – "java Output This is a command Line"?

```
class Output
 {
 public static void main(String args[])
 {
 System.out.print(args[0]);
}
 }
```

Ans. This

Q10.What is the value of “d” in the following Java code snippet? double d = Math.round (2.5 + Math.random());

Ans. 3

Q11.Which of these methods is a rounding function of Math class?

Ans. all of the mentioned

Q12. Standard output variable ‘out’ is defined in which class?

Ans. System

Q13.What will be the output of the following Java program?

```
class main_class
 {
 public static void main(String args[])
 {
 int x = 9;
if (x == 9)
{
 int x = 8; System.out.println(x);
       }
    }
 }
```

Ans. Compilation error

Q14.Which of these is the method which is executed first before execution of any other thing takes place in a program?

Ans. static method

Q15.Which of these can be used to differentiate two or more methods having the same name?

Ans. All of the mentioned

Q16. What will be the output of the following Java program?

```
class Output
 {
static void main(String args[])
 {
 int x , y = 1;
x = 10;
```

```
 if(x != 10 && x / 0 == 0)
System.out.println(y);
 Else
 System.out.println(++y);
 }
 }
```

Ans. 2

Q17.What will be the output of the following Java program?

```
class area
 {
 int width;
 int length;
 int height;
 area()
{
width = 5;
length = 6;
 height = 1;
 }
 void volume()
 {
    volume = width * height * length;
 }
}
class cons_method
{
 public static void main(String args[])
 {
area obj = new area();
 obj.volume();
 System.out.println(obj.volume);
```

```
 }
 }
```

Ans.30

Q18. Write Syntax to create/define java methods.

Ans.

```
public class Main {
  static void myMethod() {
    System.out.println("I just got executed!");
  }

  public static void main(String[] args) {
    myMethod();
  }
}
```

Q19. Write a java program following instructions.

A. Make a class Addition

a. initialize sum as 0

b. make addTwoInt method taking two int parameters a,b. make sum = a+b. Return Sum

B. define class as Method Call. Define main method

a. Create object of class Addition

b. call method using instance of object

c. Print sum

Ans.

```
public class Addition {
  private int sum;
  public Addition() {
    sum = 0;
  }
  public int addTwoInt(int a, int b) {
    sum = a + b;
    return sum;
  }
```

```
}
public class MethodCall {
  public static void main(String[] args) {
    Addition addition = new Addition();
    int sum = addition.addTwoInt(10, 20);
    System.out.println(sum);
  }
}
```

Q20. Write a java program following instructions

A. Define a class Example

a. Define two instance variables number and name

b. Define accessor (getter) methods

c. Define mutator (setter) methods

d. define method printDetails —-> print name and number

B. Define public class Demo (Main Class)

a. Define main method

b. Make Instance/object of example class

c. set number and name using instance created as 123 and Your name.

d. call printDetails method using instance

Ans.

```
public class Example {
    private int number;
    private String name;
    public int getNumber() {
        return number;
    }
    public String getName() {
        return name;
    }
    public void setNumber(int number) {
        this.number = number;
```

```
    }
    public void setName(String name) {
        this.name = name;
    }
    public void printDetails() {
        System.out.println("Name: " + name + ", Number: " + number);
    }
}
```