Practical 1: OOPs Concept in Java-1

a. Write a program to create a class and implement default, overloaded and copy constructor.

Default Constructor

Bike.java

```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 */

package com.mycompany.bike;

/**
 *
 * @author 
 */
public class Bike {
    Bike(){
        System.out.println("Bike is created");
    }

    public static void main(String[] args) {
        Bike b=new Bike();
    }
}
```

Output - Run (Bike)

```
Compiling 1 source file with javac [debug target 21] to target\classes

--- exec:3.1.0:exec (default-cli) @ Bike ---
Bike is created
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  0.967 s
Finished at: 2024-01-11T18:09:25+05:30
```

9:12 INS Unix (LF)

Overloaded Constructor

Add.java

```java
/**
 *
 * @author
 */
public class Add {
    Add(){
        System.out.println("Hiiii");
    }
    Add(int a){
        System.out.println("a="+a);
    }
    Add(char c){
        System.out.println("c="+c);
    }
    public static void main(String[] args) {
        Add obj1=new Add();
        Add obj2=new Add(10);
        Add obj3=new Add('@');
    }
}
```

Output - Run (add)

```
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes

--- exec:3.1.0:exec (default-cli) @ add ---
Hiiii
a=10
c=@
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.091 s
```

Copy Constructor

Student1.java

```java
public class Student1 {
    int id;
    String name;
    Student1(int i,String n){
        id=i;
        name=n;
    }
Student1(Student1 s){
        id=s.id;
        name=s.name;
    }
void display(){
    System.out.println(id+""+name);
}
    public static void main(String[] args) {
        Student1 s1=new Student1(111,"Karan");
        Student1 s2=new Student1(s1);
        s1.display();
        s2.display();
    }
}
```

com.mycompany.student1.Student1 > main

Output - Run (Student1)

```
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes

--- exec:3.1.0:exec (default-cli) @ Student1 ---
111Karan
111Karan
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
```

b. **Write a program to create a class and implement the concept of Method Overloading.**

Addition.java

```java
*/
public class Addition {
void add(){
    System.out.println("Zero parameter");
}
void add(char a){
    System.out.println("character="+a);
}
void add(int a,int b){
    System.out.println("a="+a+"b="+b);
}
void add(int e,float d){
    System.out.println("e="+e+"d="+d);
}
    public static void main(String[] args) {
        Addition a=new Addition();
        a.add();
        a.add('@');
        a.add(10,20);
        a.add(50,60.5F);
    }
}
```

Output - Run (Addition)

```
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes

--- exec:3.1.0:exec (default-cli) @ Addition ---
Zero parameter
character=@
a=10b=20
e=50d=60.5
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
```

c. Write a program to create a class in implement the method of Static methods.

Start Page × Number.java ×

Source History

```java
package com.mycompany.number;

/**
 *
 * @author
 */
public class Number {
    static void greatest(){
        int x=10;
        int y=20;
        if(x>y){
            System.out.println("X is greatest");}
        else{
            System.out.println("Y is greatest");}
        return;
        }


    public static void main(String[] args) {
        Number.greatest();
    }
}
```

com.mycompany.number.Number

Output - Run (Number) ×

```
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes

--- exec:3.1.0:exec (default-cli) @ Number ---
Y is greatest
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.102 s
Finished at: 2024-01-14T15:16:10+05:30
------------------------------------------------------------------------
```