LAB 7 Program: 24/11/20

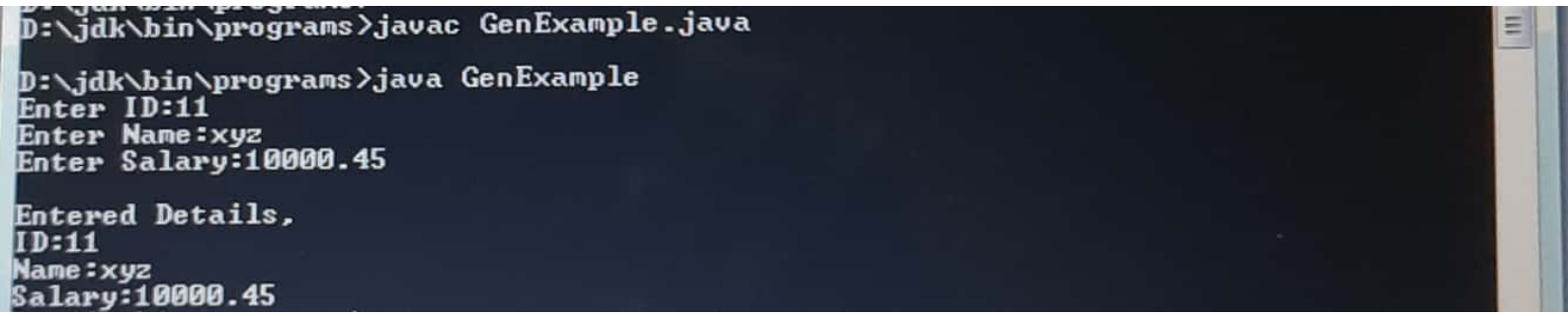
1BM19CS090
RAHIL

a) Write a program to demonstrate generics with multiple object parameters.

code:

```
import java.util.Scanner;
class Gen<I, S, F>{
    I id;
    S name;
    F sal;
    void display(){
        System.out.print("\nEntered Details,\nID: "+id);
        System.out.print("\nName: "+name+"\nSalary: "+sal);
    }
class GenExample{
public static void main(String args[]){
    Scanner ss = new Scanner(System.in);
    Gen<Integer, String, Float> ob1 = new Gen<Integer,
                                          String,
                                          Float>();
    System.out.print("Enter ID:");
    ob1.id = ss.nextInt();
    System.out.print("Enter Name:");
    ob1.name = ss.next();
    System.out.print("Enter Salary:");
    ob1.sal = ss.nextFloat();
    ob1.display();
}
}
```

```
1   import java.util.Scanner;
2   class Gen<I,S,F>{
3       I id;
4       S name;
5       F sal;
6       void display(){
7           System.out.print("\nEntered Details,\nID:"+id);
8           System.out.print("\nName:"+name+"\nSalary:"+sal);
9       }
10  }
11  class GenExample{
12      public static void main(String args[]){
13          Scanner ss=new Scanner(System.in);
14          Gen<Integer,String,Float> ob1=new Gen<Integer,String,Float>();
15          System.out.print("Enter ID:");
16          ob1.id=ss.nextInt();
17          System.out.print("Enter Name:");
18          ob1.name=ss.next();
19          System.out.print("Enter Salary:");
20          ob1.sal=ss.nextFloat();
21          ob1.display();
22      }
23  }
```



24/11/20

1BM19CS090
RAHIL

Lab 8 Program:

WAP that demonstrates the exceptions in inheritance tree. Create a base class called "Father" and derived which takes the age and throws the exception WrongAge() when the input age = father's age.

Code:

```
import java.util.Scanner;
class Father {
    int fage;
    Scanner ss = new Scanner(System.in);
    Father(){
        System.out.print("Enter Father's Age:");
        fage = ss.nextInt();
    }
}
class FatherSonAge {
    public static void main(String args[]){
        Scanner ss = new Scanner(System.in);
        Son ob2 = new Son();
        System.out.print("Enter Son's Age:");
        ob2.sage = ss.nextInt();
        try {
            if(ob2.fage <= ob2.sage)
                throw new Exception("Not Valid");
        }catch(Exception e) {
            System.out.println("Father's age cannot be
                Lesser or Equal to Son's Age\n"+e);
        }
        System.out.print("Details,\n");
        System.out.print("Father's Age:" + ob2.fage);
        System.out.print("\nSon's Age:" + ob2.sage);
    }
}
```

```java
import java.util.Scanner;
class Father{
    int fage;
    Scanner ss=new Scanner(System.in);
        Father(){
        System.out.print("Enter Father's Age:");
        fage=ss.nextInt();
        }
}
class Son extends Father{
    int sage;
}
class FatherSonAge{
    public static void main(String args[]){
        Scanner ss=new Scanner(System.in);
        Son ob2=new Son();
        System.out.print("Enter Son's Age:");
        ob2.sage=ss.nextInt();
        try{
        if(ob2.fage<=ob2.sage)
            throw new Exception("Not Valid");
        }catch(Exception e){
            System.out.println("Father's age cannot be Lesser or Equal to Son's Age\n"+e);
        }
        System.out.print("Details,\n");
        System.out.print("Father's Age:"+ob2.fage);
        System.out.print("\nSon's Age:"+ob2.sage);
    }
}
```