LAB-7

30/12/24

Exception Handling

Write a program that demonstrate handling of exceptions in inheritance tree. Create a new base class father & derive class son. In Father class implement a constructor which throws wrong Age when age<0. In son class implement a constructor which throws wrongAge() when age >= Father age.

```
import java.util.Scanner;

class wrongAge extends Exception{
    public wrongAge(String s){
        super(s);
    }
}
class input{
  Scanner sc = new Scanner(System.in);
}
class Father extends Input{
    int fatheAge;
    Father(){
    fatherAge = sc.nextInt();
     try{
       check();
     }catch(wrongAge e){
       System.out.println(e);
     }
    void check() throw WrongAge{
        if (fatheAge<0)
        throw new wrongAge("cannot
                be negative");
    }
}
```

```
void display (){
    System.out.println("Father Age : " + fatherAge);
  }
}
class son extends Father {
  int sonAge;
  son(){
    super();
      try {
       check();
    }
    catch (WrongAge R){
        System.out.println(e);
    }
  }
  void check() throws wrongAge {
    if (sonAge<0)
    throw new wrongAge ("cannot be negative");
      else if (sonAge > fatherAge)
        throw new wrongAge ("son Age can't be
            greater than father's Age");
        else if (sonAge == fatherAge");
    throw new wrongAge("son Age can't be
        equal to father's age");
    }
  void display(){
    System.out.println(e);
  }
}
```

```
class Main {
  public static void main (String, args[])
  {
    son s = new son();
      s.display();
  }
}
```

Output:

```
50
30
Father Age : 50
son Age : 30

-12
Age cannot be negative.

20
40
son's Age cannot be greater than Father's Age
```

—x—

Shreyas.K
1BM22CS271.