LAB - 7
Exception Handling

Write a Program that demonstrates handling of exceptions in Inheritance tree. Create a base class Father & derive class Son. In Father class implement a constructor which throws WrongAge 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 asc = new Scanner(System.in);
}
class Father extends Input{
      int fatherAge;
      Father(){
        fatherAge = asc.nextInt();
        try{
            check();
        } catch (WrongAge e){
            System.out.println(e);
        }

      void check() throws WrongAge{
          if (fatherAge < 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 e){
            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 ob = new Son();
        ob.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

1/2/2024

Shamanth K Murthy 1BM22CS251