PAGE :
DATE : 30/01/24

LAB-07

1) WAP that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge() when the input age < 0. In Son class, implement a constructor that takes both father and son's age and throws an exception if son's age is >= father's age.

```
class WrongAge extends Exception
{
    public WrongAge (String c)
    {
        Super (c);
    }
}

class InputScanner {
    Scanner S = new Scanner (System.in);
}

class Father extends InputScanner
{
    int FatherAge;
```

```
public Father() throws WrongAge
{
    System.out.println("Enter
        Father's age:");
    fatherAge = s.nextInt();
    if (fatherAge < 0)
    {
        throw new WrongAge("Age cannot
            be negative");
    }
}

public void display(){
    System.out.println("Father's age:
        " + fatherAge);
    }
}

class Son extends Father
{
    int SonAge;
    public Son() throws WrongAge{
        super();
        System.out.println("Enter Son's
            age:")
        SonAge = s.nextInt();
    if (SonAge >= FatherAge)
    {
        throw new WrongAge("Son's
```

```
        gt cannot be greater than father's
            age");

    else if ( SonAge <0)
    {
        throw new WrongAge ("Age cannot
            be negative");
    }
}

public void display()
{
    Super.display();
    System.out.println("Son's age:" +
    SonAge = S.nextInt() + SonAge);

}
}

public class Age
{
    public static void main(String[] args)
    {
        try{
            Son Son = new Son();
            son.display();
        }
        catch (WrongAge e)
        {
            System.out.println("Error:" + e.get
                Message());
        }
}
```

⇒ OUTPUT :-

Enter Father's Age 94
Enter Son's Age : 34

Father's Age : 94
Son's Age : 34.

30.01.24