Week

Write a program 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

```
import java.util.*;
class WrongAge extends Exception {
  WrongAge(String message) {
      super(message);
  }
}
class Father { int age;
  Father(int age) throws WrongAge {
     if(age<0) {
          throw new WrongAge("Age cannot be
                                negative");
}
this.age = age;
  }
}

class Son extends Father {
   int sage;
  Son(int fatherAge, int sonAge) throws WrongAge {
   super(fatherAge);

   if(sonAge >= fatherAge) {
   throw new WrongAge("Son's age should be less than
     Father's age");
}
```

```
        this.sAge = sonAge;
    }
}

public class Error {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        try {
            System.out.println("Enter the daddy's age:");
            int a = sc.nextInt();
            Father father = new Father(a);

            System.out.println("Enter the son's age:");
            int b = sc.nextInt();
            Son son = new Son(a, b)
        } catch (WrongAge e) {
                Sys
        catch(WrongAge e) {
            System.out.println("Exception: "+e.getMessage());
        }
    }
}
```

Output: Enter the ~~dda~~ daddy's age:

50

Enter the son's age:

55

Exception: Son's age should be less than Father's age

30.01.24