

```
import java.util.*;
class WrongAge extends Exception{
int f,s;
WrongAge(int fage,int sage){
f=fage;
s=sage;
}
public String toString(){
return "Please enter the correct ages as father's age can't be less than or equal to son's age";
}}
class NegativeAge extends Exception{
int a;
NegativeAge(int fage){
a=fage;
}
public String toString(){
return "Age can't be a negative value.";
}}
class Father{
int fage;
Scanner in=new Scanner(System.in);
Father() throws NegativeAge{
System.out.println("Enter the father's age: ");
fage=in.nextInt();
if(fage<0){
throw new NegativeAge(fage);
}}}
class Son extends Father{
int sage;
Scanner in=new Scanner(System.in);
```



```
System.out.println("Enter the father's age: ");
fage=in.nextInt();
if(fage<0){
throw new NegativeAge(fage);
}}}
class Son extends Father{
int sage;
Scanner in=new Scanner(System.in);
Son() throws NegativeAge,WrongAge{
super();
System.out.print(" Enter the son's age: ");
sage=in.nextInt();
if(sage<0){
throw new NegativeAge(sage);
}
if(sage>=fage){
throw new WrongAge(fage,sage);
}}}
class Main{
public static void main(String args[]){
try{
Son s=new Son();
}
catch(NegativeAge n){
System.out.println("Exception: "+n);
}
catch(WrongAge w){
System.out.println("Exception: "+w);
}}}

```



Search

14:15
06-12-2020

Q7 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 ~~the~~ ~~the~~ Wrong Age() when the input age < 0. In Son class, implement a constructor that calls both father and son's age and throws an exception if son's age is \geq father's age.

```
Ans import java.util.*;
class WrongAge extends Exception {
    int f, s;
    WrongAge(int fage, int sage) {
        f = fage;
        s = sage;
    }
    public String toString() {
        return "Please enter the correct ages as father's age can't be less than or equal to son's age. ";
    }
}
class NegativeAge extends Exception {
    int a;
    NegativeAge(int fage) {
        a = fage;
    }
    public String toString() {
        return "Age can't be a negative value. ";
    }
}
class Father {
    int fage;
    Scanner in = new Scanner(System.in);
    Father() throws NegativeAge {
        System.out.print("Enter the father's age: ");
        fage = in.nextInt();
    }
}
```


Date _____
Page _____

```

    if (fage < 0) {
        throw new NegativeAge (fage);
    }
}

```

```

class Son extends Father
{
    int sage;
    Scanner in = new Scanner(System.in);
    Son() throws NegativeAge, WrongAge {
        sapher();
        System.out.print("Enter the son's age:");
        sage = in.nextInt();
        if (sage < 0)
        {
            throw new NegativeAge (sage);
        }
        if (sage >= fage) {
            throw new WrongAge (fage, sage);
        }
    }
}

```

```

class Main {
    public static void main(String args[]) {
        try {
            Son s = new Son();
        }
        catch (NegativeAge n) {
            System.out.println("Exception : " + n);
        }
        catch (WrongAge w) {
            System.out.println("Exception : " + w);
        }
    }
}

```


Command Prompt

Microsoft Windows [Version 10.0.18363.1198]

(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\sohan>cd C:\Users\sohan\Desktop\Java Programs\lp8

C:\Users\sohan\Desktop\Java Programs\lp8>javac Main.java

C:\Users\sohan\Desktop\Java Programs\lp8>java Main

Enter the father's age:

55

Enter the son's age: 23

C:\Users\sohan\Desktop\Java Programs\lp8>java Main

Enter the father's age:

3

Enter the son's age: 77

Exception: Please enter the correct ages as father's age can't be less than or equal to son's age

C:\Users\sohan\Desktop\Java Programs\lp8>