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=father’s age.

```
import java.util.Scanner;

class fatherAgeException extends Exception
{
   public String toString()
   {
      return("Father's age is less than 0");
   }
}
class sonAgeException extends Exception
{    int a;
   sonAgeException(int age){        a=age;
   }
   public String toString()
   {        if(a<0)
         return ("Son's age is less than 0");        else            return ("Son's age is more
than Father's age");    }
}  class Father{
int age;
   Scanner in=new Scanner(System.in);
   Father()
   {
      System.out.println("Enter the father's age:");        age=in.nextInt();
   }
   void ex1() throws fatherAgeException
   {
      if(age<0)
         throw new fatherAgeException();
   }
}  class Son extends Father
{
   int age;
   Son()
   {
      System.out.println("Enter the son's age:");        age=in.nextInt();
   }     void ex2() throws sonAgeException{
if(age<0||age>super.age){
         throw new sonAgeException(age);        }
```

```
   }
}
class Except
{
   public static void main(String[]args){
      Son s=new Son();        try{
         s.ex1();
      }

      catch(fatherAgeException e){
         System.out.println(e);
      }

      try{
         s.ex2();        }
      catch(sonAgeException e){
         System.out.println(e);
      }
   }
}
```

Output

```
Enter the father's age:
45
Enter the son's age:
50
Son's age is more than Father's age
```

```
Enter the father's age:
30
Enter the son's age:
-23
Son's age is less than 0
```