USN:1BM19CS216
NAME:YASHASWINI SHAH
DATE:27/11/2020

LAB 8: 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 cases both father and son’s age and throws an exception if son’s age is >=father’s age.

1BM19CS216

27/11/2020

Lab 8: WAP that demonstrates handling of expections in inheritance tree. Create a base called Father and derived class "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 cases both father and son's age and throws an expection if son's age >= father's age.

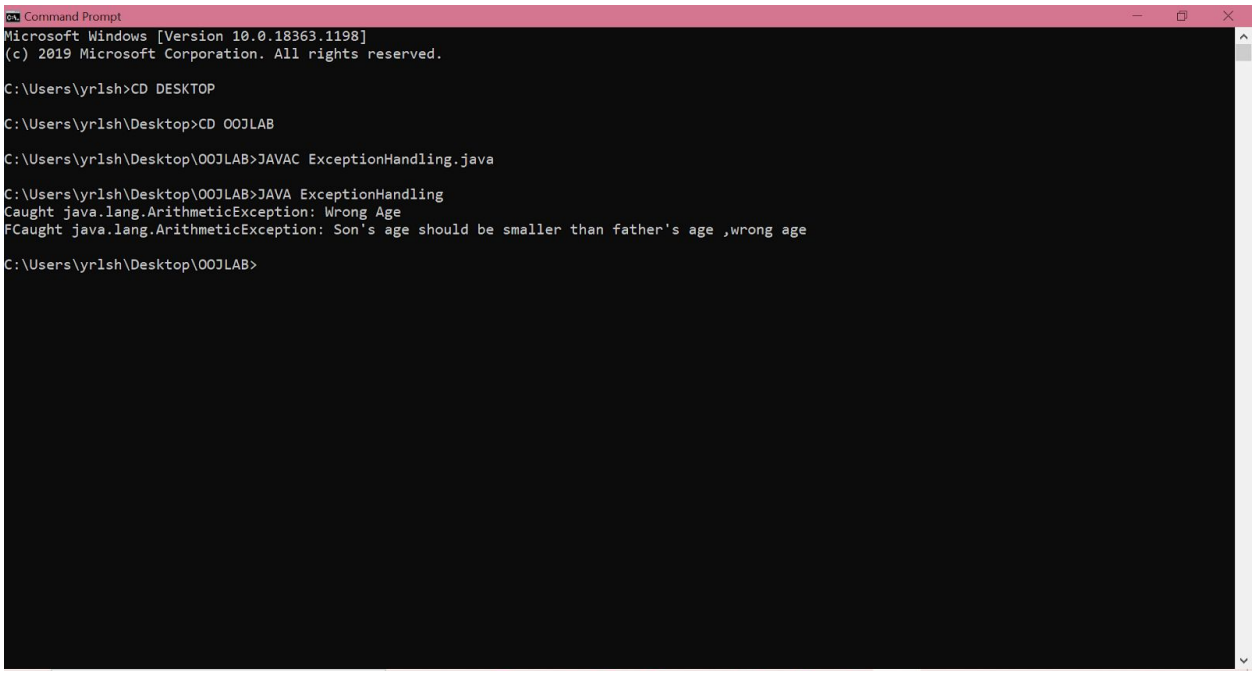
```
class Father
{
Static void acceptNameF(int inputAge) throws
    ArithmeticException.
{
try
{
if (inputAge < 0)
throw new ArithmeticException("Wrong Age");
}
catch (ArithmeticExpection e) {
System.out.println("Caught " + e);
}
}
}
class Son extends Father.
{
Static void checkSFage(int S_Age, int F_Age)
```

```
throws ArithmeticException
{
try {
if (S_Age >= F_Age)
throw new ArithmeticException ("Son's age
    should be smaller than father's age,
    wrong age");
System.out.println ("Son's age is " + S_Age +
    "Fathers age is " + F_Age);
}
catch (ArithmeticException e) {
System.out.println ("Caught " + e);
}
}
}

public class ExceptionHandling {
public static void main (String args[]) {

Father.acceptNameF(-1);
Son.checkSFage(40, 20);
}
}
```

```
class Father
{

static void acceptNameF(int inputAge) throws ArithmeticException
{
try
{
if(inputAge<0)
throw new ArithmeticException("Wrong Age");
}
catch (ArithmeticException e) {
System.out.println("Caught " + e);
}
}



}
class Son extends Father
{

static void CheckSFage(int S_Age, int F_Age) throws ArithmeticException

{
try{
if(S_Age>=F_Age)
throw new ArithmeticException("Son's age should be smaller than father's age ,wrong age");
System.out.println("Son's age is"+S_Age+"Fathers age is "+F_Age);


}
catch (ArithmeticException e) {
System.out.println("FCaught " + e);
}

}
}
public class ExceptionHandling{
public static void main(String args[]) {


Father.acceptNameF(-1);
Son.CheckSFage(40,20);


}
```

}

OUTPUT:


Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\yrlsh>CD DESKTOP
C:\Users\yrlsh\Desktop>CD OOJLAB
C:\Users\yrlsh\Desktop\OOJLAB>JAVAC ExceptionHandling.java
C:\Users\yrlsh\Desktop\OOJLAB>JAVA ExceptionHandling
Caught java.lang.ArithmeticException: Wrong Age
FCaught java.lang.ArithmeticException: Son's age should be smaller than father's age ,wrong age
C:\Users\yrlsh\Desktop\OOJLAB>