## **Lab 7:**

7. 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 uses both father and son's age and throws an exception if son's age is >=father's age.

```
Lab 7 !-
 with a pragram that dimensione houseling of exceptions in
ish wittons the custs a low day called "father" and divived
class collect non , which willout the bost class. In Father class,
 impliment a constructor which today the age and through the exception
 emongage () when the ilp age to . In non day, implement a
Constructor that cores both fother and non's age & theres on exception it
son's age is z= jothers age.
 class wrong Age Exception votends Exceptions
        public wrong traption (iting munge)?
               super (murage);
  class Father &
        predicted int ag;
         public Fother (int age) thrown wrong Age Exception ?
               if (age 20) &
                   throw new wrong for Exception ("Fother's age care he
                            negative "):
                this age = age ;
   class Sion intende Father &
          private int Sonage;
          public Son (int fother Age, int sontage) through throughout extension &
               super ( father Age );
               it (range <0) }
                     Throw new wrong Age Exception ("Son age count be ve");
               if (ton Age > = father Age) }
                    throw new wrong agreemation (" Son's agreement be greater
                      than or equal to father's eage ");
               this worky = somple;
```

```
public String to String(1) & rates "Restand Age": "+ copy of copy; "+ yorky ]

public class Ecophic Durof of copy; "End of copy; "+ yorky ]

public class Ecophic Durof & class copy; "End of copy; "+ politic copy; "the copy of copy; "End of copy; "End of copy; "the copy of copy; "End of copy; "the copy of copy; "End of copy; "the copy of copy; "the copy; "the copy of copy; "the copy; "the copy of copy; "the co
```

## **Code:**

```
class WrongAgeException extends Exception {
   public WrongAgeException(String message) {
       super(message);
   }
}

class Father {
   protected int age;

public Father(int age) throws WrongAgeException {
      if (age < 0) {
            throw new WrongAgeException("Father's age cannot be negative.");
      }
      this.age = age;</pre>
```

```
}
}
class Son extends Father {
  private int sonAge;
  public Son(int fatherAge, int sonAge) throws WrongAgeException {
     super(fatherAge);
    if (sonAge < 0) {
       throw new WrongAgeException("Son's age cannot be negative.");
     }
    if (sonAge >= fatherAge) {
       throw new WrongAgeException("Son's age cannot be greater than or equal to Father's
age.");
    this.sonAge = sonAge;
  }
  @Override
  public String toString() {
    return "Father's Age: " + age + ", Son's Age: " + sonAge;
  }
}
public class ExceptionInheritanceDemo {
  public static void main(String[] args) {
    try {
       Father father = new Father (45);
       System.out.println("Father created with age: " + father.age);
       Son son = new Son(45, 20);
```

```
System.out.println(son);
} catch (WrongAgeException e) {
System.err.println("Exception occurred: " + e.getMessage());
}

try {
Son invalidSon = new Son(40, 40);
} catch (WrongAgeException e) {
System.err.println("Exception occurred: " + e.getMessage());
}

try {
Father invalidFather = new Father(-5);
} catch (WrongAgeException e) {
System.err.println("Exception occurred: " + e.getMessage());
}

System.err.println("Exception occurred: " + e.getMessage());
}
```

## **Output:**

```
D:\Java_Lab_Programs>javac ExceptionInheritanceDemo.java

D:\Java_Lab_Programs>java ExceptionInheritanceDemo
Father created with age: 45
Father's Age: 45, Son's Age: 20
Exception occurred: Son's age cannot be greater than or equal to Father's age.
Exception occurred: Father's age cannot be negative.
```