

## LAB PROGRAM – 6

Q. 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 WrongAgeException extends Exception
{
    String msg = new String();
    WrongAgeException(String x)
    {
        msg=x;
    }
    public String toString()
    {
        return msg;
    }
}

class Father
{
    int f_age;
    Father ()
    throws WrongAgeException
    {
        Scanner s = new Scanner(System.in);
```

```

System.out.println("Enter father's age:");

f_age = s.nextInt();

if (f_age < 0)
{
throw new WrongAgeException("Father age < 0");

}

}

void display()
{
System.out.println("Father age: "+f_age);

}

}

class Son extends Father{ int s_age;
Son() throws WrongAgeException
{
Scanner s = new Scanner(System.in);
System.out.println("Enter son's age:");
s_age = s.nextInt();

if (s_age < 0)
{
throw new WrongAgeException("Son age < 0");

}

else if (s_age > f_age)
{
throw new WrongAgeException("Son age is > that father's age!");

}

}

void display()
{
System.out.println("Father age: "+f_age);

```

```
System.out.println("Son age: "+s_age);  
}  
}  
class excep  
{  
    public static void main(String[] args)  
    {  
        try  
        {  
            Father f = new Father();  
            f.display();  
            Son s = new Son();  
            s.display();  
        }  
        catch (WrongAgeException wae)  
        {  
            System.out.println(wae);  
        }  
    }  
}
```

Lab - Program-6.

- Q. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class "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  $\leq 0$ . In son class, implement a constructor that calls base father and son's age and throws an exception if son's age is  $\geq$  father's age.

```
→ import java.util.*;
class WrongAgeException extends Exception
{
    String msg = new String();
    WrongAgeException(String x)
    {
        msg = x;
    }
    public String toString()
    {
        return msg;
    }
}
class father
{
    int f_age;
    father()
    throws WrongAgeException
    {
        Scanner s = new Scanner(System.in);
```

```

System.out.println("Enter Father's age:");
f-age = s.nextInt();
if (f-age < 0)
{
    throw new WrongAgeException("Father age < 0");
}
}

```

```

void display()
{

```

```

    System.out.println("Father age: " + f-age);
}
}

```

```

class Son extends Father
{

```

```

    int s-age;
    Son() throws WrongAgeException
    {

```

```

        Scanner s = new Scanner(System.in)
        System.out.println("Enter son's age:");
        s-age = s.nextInt();
        if (s-age < 0)
        {

```

```

            throw new WrongAgeException("Son age < 0");
        }

```

```

        else if (s-age > f-age)
        {

```

```

            throw new WrongAgeException("Son age is > than father's age!");
        }
    }
}

```

```

void display()
{

```

```

    System.out.println("Father age: " + f-age);
}
}

```



```
System.out.println("Son age : " + s.age);
```

```
}
```

```
}
```

```
class excep
```

```
{
```

```
public static void main (String args[])
```

```
{
```

```
try
```

```
{
```

```
Father f = new Father();
```

```
f.display();
```

```
Son s = new Son();
```

```
s.display();
```

```
}
```

```
catch (Wrong AgeException wae)
```

```
{
```

```
System.out.println(wae);
```

```
}
```

```
}
```

```
}
```

Output:

Enter father's age:  
50

Ⓢ father age : 50

Enter father's age  
50

Enter son's age  
20

Father age: 50

Son age: 20

Enter ~~father~~ father's age  
10

Father age: 10

Enter father's age:  
10

Enter son's age:  
20

Son age is > that father's age.

~~35~~  
35

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\STUDENT>cd C:\Users\STUDENT\Desktop\1bm1cs034
C:\Users\STUDENT\Desktop\1bm1cs034>javac exception.java
C:\Users\STUDENT\Desktop\1bm1cs034>java excep
Enter father's age:
50
Father age: 50
Enter father's age:
50
Enter son's age:
20
Father age: 50
Son age: 20

C:\Users\STUDENT\Desktop\1bm1cs034>java excep
Enter father's age:
10
Father age: 10
Enter father's age:
10
Enter son's age:
20
Son age is > that father's age!

C:\Users\STUDENT\Desktop\1bm1cs034>
```