

## Program 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  $\geq$  father's age.

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 uses both father and son's age and throws an exception if son's age is  $\geq$  father's age.

```
Class WrongAgeException extends Exception {  
    public WrongAgeException (String message) {  
        super (message);  
    }  
}  
  
Class Father {  
    int age;  
    public Father (int age) throws WrongAgeException {  
        if (age < 0) {  
            throw new WrongAgeException ("Father's age cannot be  
negative.");  
        }  
        this.age = age;  
    }  
}
```

```
Class Son extends Father {  
    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;
    }

```

```

}

```

```

public class ExceptionInheritance Demo

```

```

    public static void main (String [] args) {
        try {

```

```

            Father father = new Father (40);

```

```

            Son son = new Son (40, 20);

```

```

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

```

```

            System.out.println (" Son's age: " + son.sonAge);

```

```

        } catch (WrongAgeException e) {

```

```

            System.out.println ("Exception caught: " + e.getMessage());

```

```

        }

```

```

    }

```

```

}

```

## Output

Case 1: Valid Input

Father father = new Father(40);

Father's age : 40

Son Son = new Son(40, 20);

Son's age : 20

✓  
22/11/24

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

```
class Father {  
    int age;  
  
    public Father(int age) throws WrongAge {  
        if (age < 0) {  
            throw new WrongAge("Age cannot be negative for Father.");  
        }  
        this.age = age;  
        System.out.println("Father's age is " + this.age);  
    }  
}
```

```
class Son extends Father {  
    int sonAge;  
  
    public Son(int fatherAge, int sonAge) throws WrongAge {  
  
        super(fatherAge);
```



```

        if (sonAge >= fatherAge) {
            throw new WrongAge("Son's age cannot be greater than or equal to Father's age.");
        }
        this.sonAge = sonAge;
        System.out.println("Son's age is " + this.sonAge);
    }
}

```

```

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

        try {
            Father father1 = new Father(-5);
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }

        try {
            Son son1 = new Son(40, 45);
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }

        try {
            Son son2 = new Son(40, 30);
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

}

```
D:\24BMSCE>javac Age.java
```

```
D:\24BMSCE>java Age
```

```
Error: Age cannot be negative for Father.
```

```
Father's age is 40
```

```
Error: Son's age cannot be greater than or equal to Father's age.
```

```
Father's age is 40
```

```
Son's age is 30
```