

Lab-7

Page No.

Date

Write a program that demonstrate handling of exception in inheritance tree. Create base class Father and derived class son who extends father. In father class, implement constructor that takes both father and son's age and throws exception if son's age > father's age.

```
import java.util.Scanner;
```

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

```
class InputScanner {
```

```
    Scanner s = new Scanner(System.in);  
    public int nextInt() {  
        return s.nextInt();  
    }  
}
```

```
class Father extends InputScanner {
```

```
    int fatherAge;
```

```
    public Father() throws WrongAge {
```

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

```
        fatherAge = nextInt();
```

```
        if (fatherAge < 0) {
```

```
            throw new WrongAge("Age can't be negative");  
        }  
    }  
}
```

```
    }  
}
```

```
    public void display() {
```

```
        System.out.println("Father's age: " + fatherAge);  
    }  
}
```



```

}
class Son extends Father {
    int sonAge;
    public Son() throws WrongAge {
        super();
        System.out.println("Enter son's age:");
        sonAge = nextInt();
        if (sonAge >= FatherAge) {
            throw new WrongAge("son's Age can't be greater than father's Age");
        }
        else if (sonAge < 0) {
            throw new WrongAge("Age can't be negative");
        }
    }
}

```

```

    public void display() {
        super.display();
        System.out.println("son's age: " + sonAge);
    }
}

```

```

public class Main {
    public static void main (String[] args) {
        try {
            Son son = new Son();
            son.display();
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

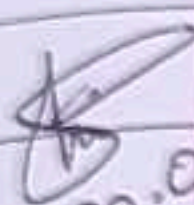



Output

Enter father's age : 23

Enter son's age : 29

~~Error~~: son's age can't be greater than father's age


30.01.24