

• WAP of inheritance tree if son's age's \geq father's age.

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

```
class WrongAge extends Exception {
```

```
    public WrongAge(String s) {
```

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

```
    }
```

```
}
```

```
class Father {
```

```
    int age;
```

```
    public Father(int age) {
```

```
        this.age = age;
```

```
    }
```

```
    public void checkAge() throws WrongAge {
```

```
        if (age < 0)
```

```
        {
```

```
            throw new WrongAge("Age cannot be negative");
```

```
        }
```

```
    }
```

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

```
        System.out.println("Father's age = " + age);
```

```
    }
```

```
}
```

```
class Son extends Father {
```

```
    public Son(int age) {
```

```
        super(age);
```

```
    }
```

```
int sage = age;
```

```
public void checkAge() throws WrongAge {
```

```
    if (sage < 0)
```

```
    {
```

```
        throw new WrongAge("Age cannot  
        be negative");
```

```
    }
```

```
    else if (sage > 0)
```

```
    {
```

```
        throw new WrongAge("son cannot  
        older than father");
```

```
    }
```

```
    else if (sage == age)
```

```
    {
```

```
        throw new WrongAge("son cannot  
        be same age as Father");
```

```
    }
```

```
}
```

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

```
    System.out.println("son's age = " + sage);
```

```
}
```

```
}
```

```
public class AgeException {
```

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

```
        Scanner sc = new Scanner (System.in)
```

```
        System.out.println ("Enter age of father");
```

```
        Father f = new Father (sc.nextInt());
```

```
        System.out.println ("Enter age of son");
```


son s = new son (sc.nextInt());

sc.close();

s.age = f.age;

try {

f.checkage();

s.checkage();

s.display();

d.display();

}

Catch (InterruptedException e)

{

System.out.println(e);

}

}

}

Output →

Enter age of Father

50

Enter the age of son

-10

Age cannot be negative

Wrong Age

Java Exception Age

Enter age of Father

20

Enter age of son

30

son cannot be older than father

Wrong Age

Java Exception Age

Enter age of Father
20

Enter age of son
20

Don cannot be same age as Father
Wrong Age

Just Enrichment Age

Enter age of Father

30

Enter age of son

5

Don's Age = 5

Father's Age = 30