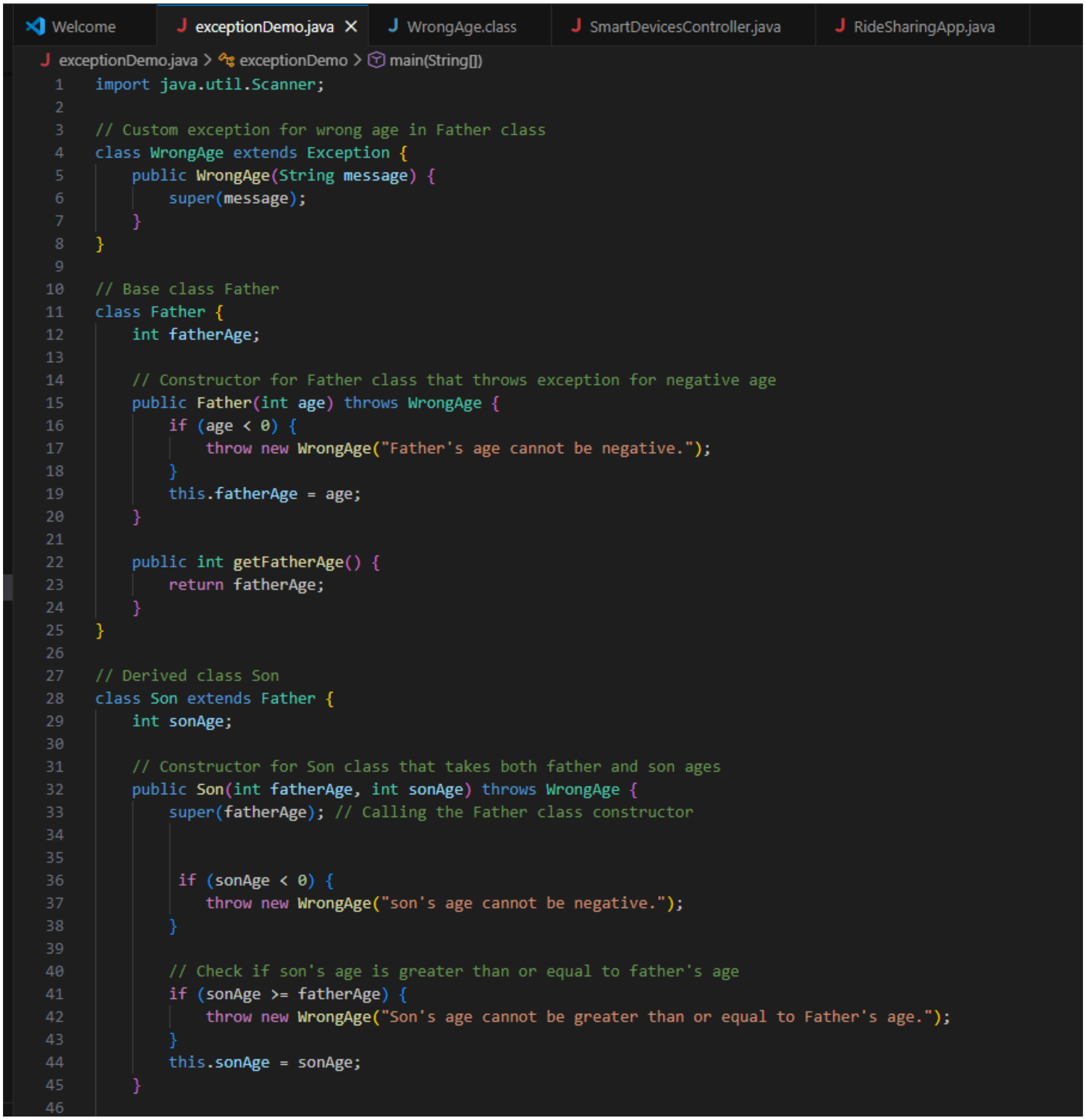
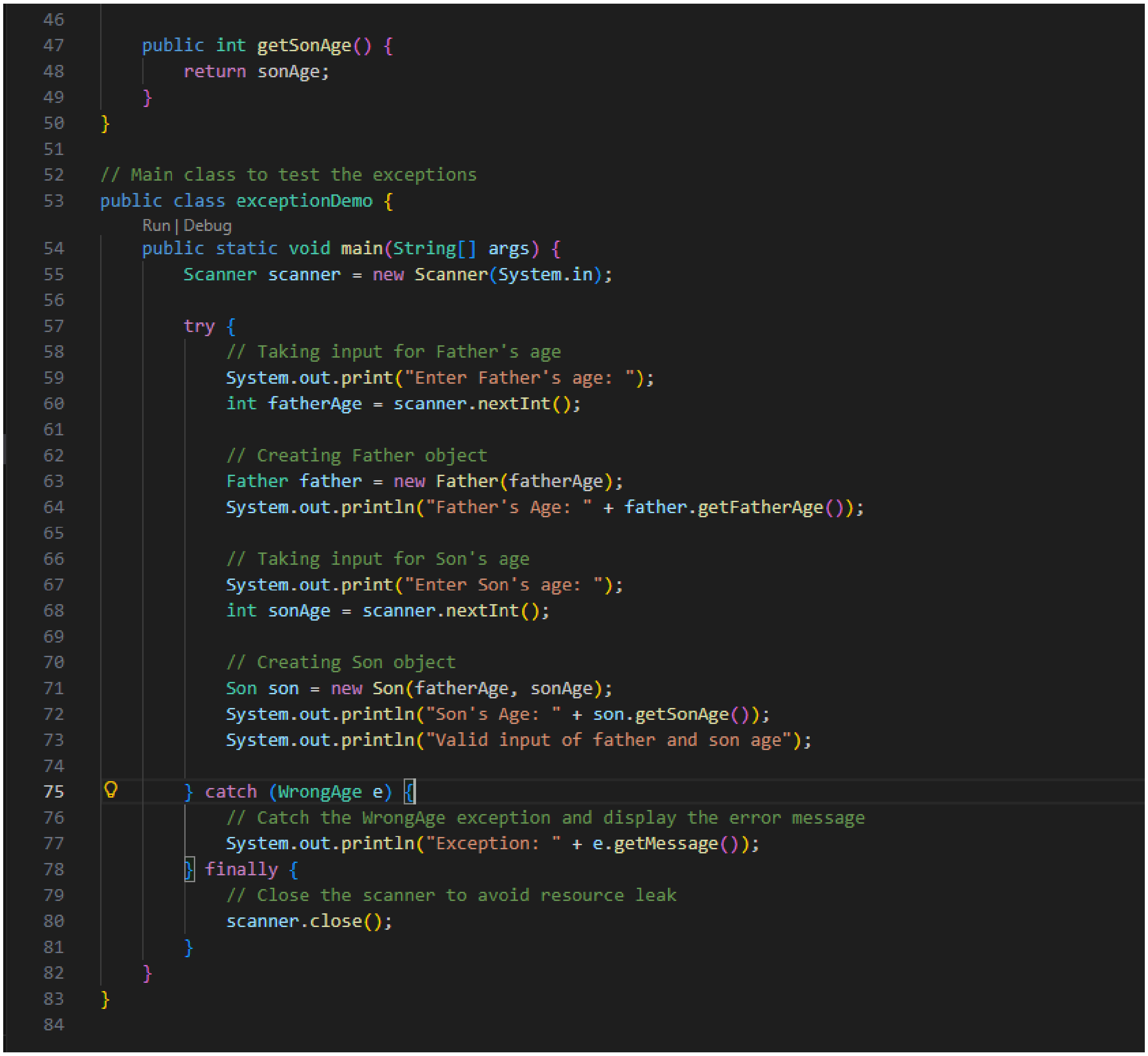
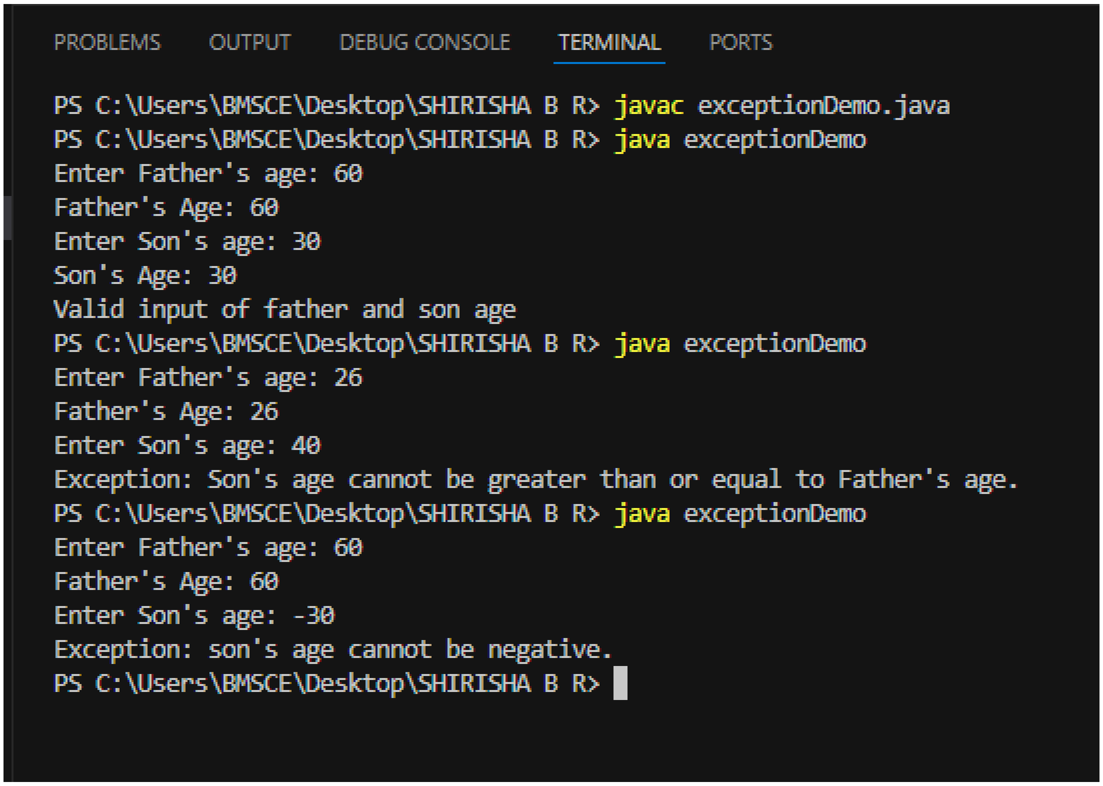
JAVA WEEK-07


Welcome
exceptionDemo.java
WrongAge.class
SmartDevicesController.java
RideSharingApp.java
exceptionDemo.java > exceptionDemo > main(String[])

```java
import java.util.Scanner;

// Custom exception for wrong age in Father class
class WrongAge extends Exception {
    public WrongAge(String message) {
        super(message);
    }
}

// Base class Father
class Father {
    int fatherAge;

    // Constructor for Father class that throws exception for negative age
    public Father(int age) throws WrongAge {
        if (age < 0) {
            throw new WrongAge("Father's age cannot be negative.");
        }
        this.fatherAge = age;
    }

    public int getFatherAge() {
        return fatherAge;
    }
}

// Derived class Son
class Son extends Father {
    int sonAge;

    // Constructor for Son class that takes both father and son ages
    public Son(int fatherAge, int sonAge) throws WrongAge {
        super(fatherAge); // Calling the Father class constructor


         if (sonAge < 0) {
            throw new WrongAge("son's age cannot be negative.");
        }

        // Check if son's age is greater than or equal to father's age
        if (sonAge >= fatherAge) {
            throw new WrongAge("Son's age cannot be greater than or equal to Father's age.");
        }
        this.sonAge = sonAge;
    }

```



```java

    public int getSonAge() {
        return sonAge;
    }
}

// Main class to test the exceptions
public class exceptionDemo {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        try {
            // Taking input for Father's age
            System.out.print("Enter Father's age: ");
            int fatherAge = scanner.nextInt();

            // Creating Father object
            Father father = new Father(fatherAge);
            System.out.println("Father's Age: " + father.getFatherAge());

            // Taking input for Son's age
            System.out.print("Enter Son's age: ");
            int sonAge = scanner.nextInt();

            // Creating Son object
            Son son = new Son(fatherAge, sonAge);
            System.out.println("Son's Age: " + son.getSonAge());
            System.out.println("Valid input of father and son age");

        } catch (WrongAge e) {
            // Catch the WrongAge exception and display the error message
            System.out.println("Exception: " + e.getMessage());
        } finally {
            // Close the scanner to avoid resource leak
            scanner.close();
        }
    }
}

```



```
PS C:\Users\BMSCE\Desktop\SHIRISHA B R> javac exceptionDemo.java
PS C:\Users\BMSCE\Desktop\SHIRISHA B R> java exceptionDemo
Enter Father's age: 60
Father's Age: 60
Enter Son's age: 30
Son's Age: 30
Valid input of father and son age
PS C:\Users\BMSCE\Desktop\SHIRISHA B R> java exceptionDemo
Enter Father's age: 26
Father's Age: 26
Enter Son's age: 40
Exception: Son's age cannot be greater than or equal to Father's age.
PS C:\Users\BMSCE\Desktop\SHIRISHA B R> java exceptionDemo
Enter Father's age: 60
Father's Age: 60
Enter Son's age: -30
Exception: son's age cannot be negative.
PS C:\Users\BMSCE\Desktop\SHIRISHA B R>
```