Exercise 5

Task 1 – Writing a Class That Uses if/else Statements

In this task, you are going to write a class called DateTwo that uses if/else statements to display the day of the week based on the value of a variable. Follow these steps to write your class.

1. Create a new Java project called Exercise5.
2. In the newly created project, add a new Java class called DateTwo with one member variable called dayNumber. Assign a value to dayNumber between 1 and 7 where
    a. The number 1 represents Monday (beginning of the week)
    b. The number 7 represents Sunday (end of the week)
3. In the DateTwo class, create a displayDay method that uses if/else constructs to inspect the value of the dayNumber and displays the corresponding day of the week. The displayDay method should also display and error message if an invalid number is found.
4. Create a new Java class called DateTwoTest and add codes below.

```
public class DateTwoTest {

    public static void main (String args[]) {

        DateTwo myDateTwo = new DateTwo();
        myDateTwo.displayDay();
    }
}
```

5. Run the program and test it using DateTwoTest.
6. Repeat step 5 several times by assigning different values to the DateTwo member variable dayNumber in the DateTwoTest class.

Task 2 – Writing Another Class That Uses if/else Statements

In this task, you write a class called Clock that uses if/else statements to display the part of day depending on the time of day. Use the following table as a guideline.

Time of Day	Part of Day
8:01 to 12:00	Morning
12:01 to 17:00	Afternoon
17:01 to 24:00	Evening
0:01 to 8:00	Early Morning

Follow these steps to write your program.

1. Create a new Java class called Clock with a variable currentTime that contains the hour of the day.
2. In the Clock class, create a displayPartOfDay method that uses if/else constructs to display part of the day associated with the value of the currentTime variable. For example, if the value of the currentTime variable is equal to 1504, your program would display "Afternoon".
3. Create a test class called ClockTest with codes below.

```
public class ClockTest {

    public static void main (String args[]) {

        Clock myClock = new Clock();
        myClock.displayPartOfDay();
    }
}
```

4. Run the program and verify the output.
5. Repeat several times by assigning different values to the Clock member variable currentTime in the ClockTest class.

Note – A leading zero indicates an octal value. Therefore, the program does not compile if you set `currentTime` to 0800. You need to specify `currentTime` as 800 for 8:00 AM to successfully compile the program. No tests have been done for values that lie outside the range of 100 and 2400.

Task 3 – Writing a Class That Uses the switch Statement

In this task, you create a class called Month that uses switch statements to display the name of the month based on the numeric value of a variable. Complete the following steps.

1. Create a new Java class called Month.
2. In the Month class, create a variable called monthNumber. Assign a value to the variable that is between 1 and 12, where number 1 represents the month of January, and the number 12 represents the month of December.
3. In the Month class, create a displayMonth method that uses a switch construct to inspect the value of the monthNumber variable and displays the corresponding name of the month. The displayMonth method should also display an error message if an invalid number is used.
4. Create a new test class called MonthTest.

```
public class MonthTest {

    public static void main (String args[]) {

        Month myMonth = new Month();
        myMonth.displayMonth();
    }
}
```

5. Execute the program by running the MonthTest class file.
6. Repeat step 5 several times by assigning different value to the Month member variable monthNumber.

THE END.