

# Task Sheet

## General Instructions

You will have the afternoon to complete these tasks.

Create a new project for all of the following tasks. As a suggestion, you can call your main project “tasks” but you can call it whatever suits you.

Each task will go into its own separate package and will be run individually. Each task will require its own `main` method. This would mean the following structure with each `App.java` file having its own `main`.

```
| - com.sparta
|   |
|   | - day1
|   |   |
|   |   | - debug
|   |   |   |
|   |   |   | - DebugApp.java
|   |   |
|   |   | - calculator
|   |   |   |
|   |   |   | - CalculatorApp.java
|   |
|   | - day2
```

For all tasks you should be able to explain and justify the solution that you arrive at.

# Tasks

## Understanding Conditions

---

You only need to make a note of your answers to the following (for example in a text document), no other software is required.

Assuming  $a = 5$ ,  $b = 4$ ,  $c = 3$ , indicate whether each of the following conditions resolve to true or false:

- a)  $(a < b)$
- b)  $(a != b)$
- c)  $(a > b)$  and  $(a < c)$
- d)  $(a > b)$  or  $(a < c)$
- e)  $(a - 1 == b)$
- f)  $\text{not}(b > c)$

## Number Series

---

Package: `com.sparta.day4.series`

Class: `SeriesApp`

Write a program to prompt the user for 5 integers either positive or negative (assume zero will not be an input). Sum the total of positive integers and the total of negative integers and then output the totals.

For example, if the inputs were: -5 12 -19 -3 6 the program would output:

```
Sum of positive integers: 18
Sum of negative integers: -27
```

Test your program with a suitable range of positive and negative numbers.

## Nothing for a Pair (not in this game)

---

Package: `com.sparta.day4.highlow`

Class: `HighLowApp`

## Core Java

Write a program that will deal a playing card randomly, allow the user to guess whether the next card dealt is higher or lower, and then deal another card (checking to see whether the user was correct).

If the user guessed correctly the program must output “You win!”. If they didn’t guess correctly (including if the generated values were the same), the program must output “You lose!”.

The program will need to display the ‘card’ value associated with the random number as follows:

Number generated	Value displayed
1	Ace
2 – 10	Face value, so if a 2 display a 2 etc.
11	Jack
12	Queen
13	King

Think about how you would test that your code works. Try to structure the code to maximise its testability.

### Under Age

---

Package: com.sparta.day4.agevalidation

Class: AgeValidationApp

Write a program that gets the user to input their name and their date of birth.

Your program should then check that the user’s name was entered, and that they are over eighteen. Output the results.

### Password Checker

---

Package: com.sparta.day4.password

Class: PasswordCheckerApp

Write a program with the following hard-coded password as a string constant

## Core Java

```
static void main(String[] args) {  
  
    final String PWD = "Rocket";  
  
}
```

Prompt the user to enter a password (as a string).

If the password entered by the user matches the password you have stored in the program, regardless of case, output a welcome message.

## Thunderbirds Are Go

---

Package:      com.sparta.day4.thunderbirds

Class:        ThunderbirdApp

Write a program to prompt the user to enter a number between 1 and 4 inclusive.

Your program should then output the following in response:

- If user enters 1, output "Thunderbird 1 pilot is Scott Tracy"
- If user enters 2, output "Thunderbird 2 pilot is Virgil Tracy"
- If user enters 3, output "Thunderbird 3 pilot is Alan Tracy"
- If user enters 4, output "Thunderbird 4 pilot is Gordon Tracy"
- If the user enters any other number, output "Have you never watched Thunderbirds!"

Test your program with each permitted input and with one number outside the permitted input range.