



Java Lab

if Statements and Input using **Scanner** – Bonus Lab

IMPORTANT! Save all your work to a safe location such as oneDrive.

Create a folder for SDPD into which you will save all your work for this module, arranged how you wish. Ideally you should create a folder each week for your lab exercises. Note that you should create a separate file for each exercise.

Exercise 1

Goal: Create a program in Java convert from Fahrenheit to Celsius and vice-versa.

Create a new file called *Temperature* for this exercise.

Create a program that prompt the user to enter a temperature. The program will then ask the user if this value is Farenheit or Celsius, and will then convert to the other. Your putput should be similar to as shown below.

To convert from Fahrenheit to Celsius, the following formula can be used:

$$(\text{temp} - 32) * (5.0 / 9.0)$$

To convert from Celsius to Fahrenheit, the following formula can be used:

$$(\text{temp} * (9.0 / 5.0)) + 32$$

```
C:\Windows\system32\cmd.exe
-- Temperature Converter --
Please enter a temperature: 32
Is this temperature in Fahrenheit or Celsius? (enter f or c): f

The temperature in Celsius is: 0.0
Press any key to continue . . .
```

```
C:\Windows\system32\cmd.exe
-- Temperature Converter --
Please enter a temperature: 100
Is this temperature in Fahrenheit or Celsius? (enter f or c): c

The temperature in Fahrenheit is: 212.0
Press any key to continue . . .
```

Exercise 2

Goal: Create a program in Java to convert from Fahrenheit to Celsius and vice-versa.

Create a new file called *Temperature* for this exercise.

Your program should calculate and print the Electricity bill of a given customer. The customer ID, name and number of units consumed by the user should be taken from the keyboard and the total amount to pay to the customer displayed on the console. The charges are as follows:

up to 199 units cost €1.20 per unit
up to 400 units cost €1.50 per unit

```
C:\Windows\system32\cmd.exe
Enter customer ID: 445
Enter customer name: John
Enter number of units consumed: 344

- - - - - CUSTOMER BILL - - - - -
Customer ID: 445
Customer name: John
Units consumed: 344
Total amount to pay: 516.0
Press any key to continue . . .
```

```
C:\Windows\system32\cmd.exe
Enter customer ID: 123
Enter customer name: Bob
Enter number of units consumed: 122

- - - - - CUSTOMER BILL - - - - -
Customer ID: 123
Customer name: Bob
Units consumed: 122
Total amount to pay: 146.4
Press any key to continue . . .
```

Exercise 3

Goal: Create a program in Java about a lab policy for a college.

Create a new file called *LabPolicy* for the fictitious *ABC college* for this exercise.

Your program will prompt the user for three values: whether or not they attended lab, whether or not they turned in the program (first program if they attended lab, third program if they did not), and whether or not the program was submitted on time (only prompt if they did submit a program). The program will then display output appropriate for the user's responses: if everything was done correctly, the program will display an affirmative message (the same message will be displayed for those who attended lab and submitted program 1 or did not attend and submitted program 3). If something was done incorrectly, the program will display a negative message and then offer a line of help specific to the reason for not completing the lab.

Below are six example runs of the program with input **Highlighted in green**. Make sure that your output matches the examples below (except for the sections determined by user input).

Example 1

```
Did you attend lab this week? [y / n] y
Did you submit program 1 to D2L? [y / n] y
Did you submit the program by the deadline? [y / n] y

Congratulations! You fulfilled your lab obligations for the week.
```

Example 2

```
Did you attend lab this week? [y / n] Y
Did you submit program 1 to D2L? [y / n] y
Did you submit the program by the deadline? [y / n] N

Sorry, you did not fulfill your lab obligations for the week.
Please try to get your program submitted on time in the future!
```

Example 3

```
Did you attend lab this week? [y / n] y
Did you submit program 1 to D2L? [y / n] n

Sorry, you did not fulfill your lab obligations for the week.
You need to submit your program in order to get credit.
```

Example 4

```
Did you attend lab this week? [y / n] n
Did you submit program 3 to D2L? [y / n] y
Did you submit the program by the deadline? [y / n] y

Congratulations! You fulfilled your lab obligations for the week.
```

Example 5

```
Did you attend lab this week? [y / n] N
Did you submit program 3 to D2L? [y / n] Y
Did you submit the program by the deadline? [y / n] N

Sorry, you did not fulfill your lab obligations for the week.
Please try to get your program submitted on time in the future!
```

Example 6

```
Did you attend lab this week? [y / n] n
Did you submit program 3 to D2L? [y / n] n

Sorry, you did not fulfill your lab obligations for the week.
Perhaps you should attend lab in the future.
```

Some prompts and questions to consider *before* starting to write the program:

- How will you determine whether or not to display particular prompts (e.g., asking if the program was submitted by the deadline).
- How will you print the same negative message if the user did not complete the lab requirements for the week, but provide a different line of help depending on the situation?
- Consider how you might use variables to your advantage in this program, particularly in how you name your variables, such that your conditional statements read almost like English (e.g., "if the student did not attend lab and submitted program 3 and submitted the program on time...").
- Note that in the examples, users can provide their answer with either an uppercase or lowercase letter. How might you accommodate both kinds of answers? You can check for both types of answers in your conditionals.

The following is a high level checklist of requirements for your program:

- ☐ Your class is named LabPolicy and is placed in the project created for this assignment
- ☐ Your code is commented outlining what function each line of code performs
- ☐ Your code is formatted with proper indentation and appropriate variable names
- ☐ Your code fulfills the functionality outlined above
- ☐ You have all the prompts listed in the examples in the exact same order
- ☐ You have blank lines as indicated in the examples
- ☐ Your strings (both prompts and output) are formatted **exactly** as shown in the examples

Exercise 4

Create a program that will determine whether or not a given integer is divisible by 3. Print the result (e.g., "Divisible by 3" or "Not divisible by 3").

Exercise 5

Write code that, for a given number, prints **Fizz** if the number is divisible by 3, **Buzz** if the number is divisible by 5, and **Fizz Buzz** if the number is divisible by both.

Exercise 6

Write a java program called *XCheck* that assigns 1 to x if y is greater than 0.

Exercise 7

Write a java program called *SameNumbers* that checks whether two numbers entered by the user are the exact same.

Exercise 8

Write a java program called *SameCharacters* that checks whether two characters entered by the user are the exact same.

Exercise 9

Write a java program called *SameString* that checks whether two strings by the user are the exact same.

Exercise 10

Write a java program called *NumberCheck* that checks whether an integer entered by the user is positive and divisible by 2.

Exercise 11

Write a java program to find the eligibility of admission for a professional course based on the following criteria:

Eligibility Criteria :

Marks in Maths ≥ 65

Marks in Phy ≥ 55

Marks in Chem ≥ 50

Total in all three subject must be ≥ 190

Expected Output :

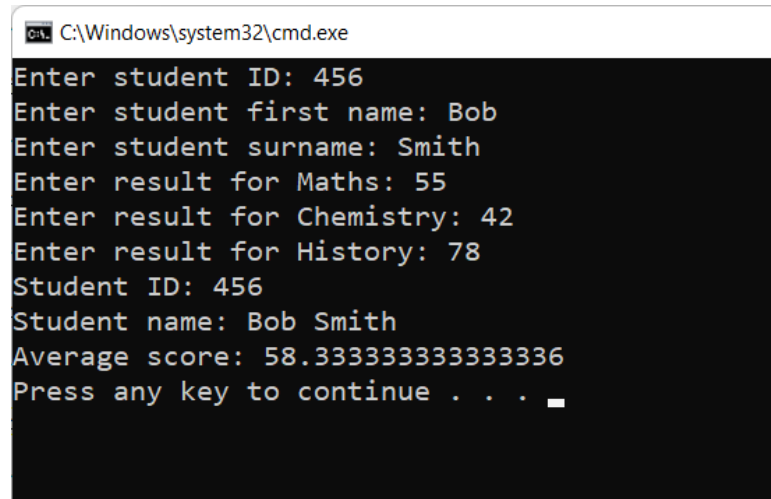
The candidate is not eligible for admission

OR

The candidate is eligible for admission

Exercise 12

Write a java program that prompts the user to enter a student ID, student first name, student surname, and results for 3 college subjects. The program should then calculate the average score that the student has, and produce output similar to as shown below:



```
C:\Windows\system32\cmd.exe
Enter student ID: 456
Enter student first name: Bob
Enter student surname: Smith
Enter result for Maths: 55
Enter result for Chemistry: 42
Enter result for History: 78
Student ID: 456
Student name: Bob Smith
Average score: 58.333333333333336
Press any key to continue . . .
```