# Answers to Questions from TT1.2

Name: Minh Nguyen

Student ID: 104309099

1. **Desk Check Task: Calculate Bill Total**

## Required Variables:

## ***Real (floating point):***

## ***appetizer\_price, main\_price, dessert\_price***

## ***total\_price***

## Pseudocode:

## ***Read the value of*** *appetizer\_price*

## ***Read the value of*** *main\_price*

## ***Read the value of*** *dessert\_price*

## *total\_price = appetizer\_price + main\_price + dessert\_price*

## ***Print ‘$’ and then the value of*** *total\_price* ***to the terminal showing two decimal places.***

## Test Data:

|  |  |  |
| --- | --- | --- |
|  | First data set | Second data set |
| *appetizer\_price* | 10.30 | 12.40 |
| *main\_price* | 34.00 | 41.00 |
| *dessert\_price* | 8.50 | 9.80 |

## Expected Result:

|  |  |  |
| --- | --- | --- |
|  | First data set | Second data set |
| *Output:* | $52.80 | $63.20 |

## **Desk check** - fill this in by hand-tracing/hand-executing the pseudocode provided with the test data above:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Statement | *appetizer**\_price* | *main**\_price* | *dessert**\_price* | *total**\_price* | *output* |
| ***First Pass*** | ***Read the value of*** *appetizer\_price* | ***10.30*** |  |  |  |  |
| ***Read the value of*** *main\_price* | ***10.30*** | ***34.00*** |  |  |  |
| ***Read the value of*** *dessert\_price* | ***10.30*** | ***34.00*** | ***8.50*** |  |  |
| ***Calculate the*** *total\_price* | ***10.30*** | ***34.00*** | ***8.50*** | ***52.80*** |  |
| ***Convert to dollars*** | ***10.30*** | ***34.00*** | ***8.50*** | ***$52.80*** |  |
| ***Output the*** *total\_price* | ***10.30*** | ***34.00*** | ***8.50*** | ***$52.80*** | ***$52.80*** |
| ***Second Pass*** | ***Read the value of*** *appetizer\_price* | ***12.40*** |  |  |  |  |
| ***Read the value of*** *main\_price* | ***12.40*** | ***41.00*** |  |  |  |
| ***Read the value of*** *dessert\_price* | ***12.40*** | ***41.00*** | ***9.80*** |  |  |
| ***Calculate the*** *total\_price* | ***12.40*** | ***41.00*** | ***9.80*** | ***63.20*** |  |
| ***Convert to dollars*** | ***12.40*** | ***41.00*** | ***9.80*** | ***$63.20*** |  |
| ***Output the*** *total\_price* | ***12.40*** | ***41.00*** | ***9.80*** | ***$63.20*** | ***$63.20*** |

1. **Complete the Program Calculate Bill Total**

Now check the actual code produces the output you expected

Do this by completing the missing code in **bill\_total.rb** in **Task 1.3** and then running the program.

1. **Short Answer Questions:**

**Focus on the following on using the correct computing terminology.**

Here are some terms that may help you: Assignment, evaluate, increment,

## Using a few sentences explain why it may be important to execute statements in the correct sequence. (eg: what might happen if the last statement in Program 2 was executed earlier)

Following the correct sequence while writing code for a program is mandatory for successfully compiling the program without errors. We also cannot get the expected output if not following the correct sequence.

## 2: The code **main\_price = 10** is an example of which kind of programming statement?

This is an assignment statement.

## 3: What **actions** does the computer perform when it executes **a = a + b**?

The computer first adds the two numbers then it assigns the sum to the variable a.

## 4: How would the value of variable i change in statement **i = i + 1**?

The value of i will be incremented by 1 and assigned to the variable i.

## 5: ***What sort of types*** ***will Ruby use to store the following variables*** (given the associated variable values)?

|  |  |  |
| --- | --- | --- |
|  | **Data** | **Type** |
|  | A person's name e.g: “Fred Smith” | String |
|  | Number of students in a class e.g: 23 | Integer |
|  | The average age of a group of people e.g: 23.5 | Float |
|  | A temperature in Celsius e.g: 45.7 | Float |
|  | True or false e.g: 1 == 2 | Boolean |

Note: possible types include: Integer, String, Float, Boolean

## 6: ***Variables have a scope – what are two different scopes variables can have in Ruby?***

There are two different scope variables comprised in Ruby:

Local- refers to the variables that possess narrow scope. This is declared inside a method and no longer exists after the method terminates or returns its value.

Global- refers to those variables in a program that possesses wider scope than local and can be accessed from anywhere in the program.

*See the lesson materials for help with Question 6. You could also see:*

[*https://www.tutorialspoint.com/ruby/ruby\_variables.htm*](https://www.tutorialspoint.com/ruby/ruby_variables.htm)