Answers to Questions from TT1.2

Name: Minh An Nguyen
Student ID: 104844794

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 '$' 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		*34.00*			
	Read the value of dessert_price			*8.50*		
	Calculate the total_price				*52.80*	
	Convert to dollars					*$52.80*
	Output the total_price					*$52.80*
Second Pass	*Read the value of appetizer_price*	*12.40*				
	Read the value of main_price		*41.00*			
	Read the value of dessert_price			*9.80*		
	Calculate the total_price				*63.20*	
	Convert to dollars					*$63.20*
	Output the total_price					*$63.20*

2. Complete 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** then running the program.

3. Short Answer Questions:

Focus in the following on using the correct computing terminology.

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

1. 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)
   Because the program cannot perform calculations on variables that haven't been declared yet. Moreover, executing a program in a random sequence would lead to a wrong answer (e.g. the variables still hold the old values) or error.

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 *calculates the sum of a and b.*
Then it *assigns the sum to variable a.*

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

The value of i will be *increased by 1.*

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
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?*

- Local Variables
  - Limited scope: Only accessible within the methods, blocks, or loops they're defined in.
- Global Variables
  - Available everywhere: Accessible from any part of your program.

See the lesson materials for help with Question 6. You could also see:
https://www.tutorialspoint.com/ruby/ruby_variables.htm