Answers to Questions from TT1.2

Name: Dylan Rodwell
Student ID: 105341089

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	*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*

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)

It is important that code executes in sequence. It allows programmers to design processes of steps in a particular order for the computer to run so that it does what is being asked as efficiently as possible. It would be almost impossible to write code if it wasn't executed in sequence because it would be very hard to predict what statement would go next and designing complicated software would not be possible.

2: The code main_price = 10 is an example of which kind of programming statement?

This is a declaration statement.

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

The computer first reads the variable 'a'
Then it reads the variable 'b'

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

The value of i will be 1 digit greater than it was (i+1) for example: if 'i' was 2, then the new 'i' would be 3

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	Bool

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

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

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

https://www.tutorialspoint.com/ruby/ruby_variables.htm