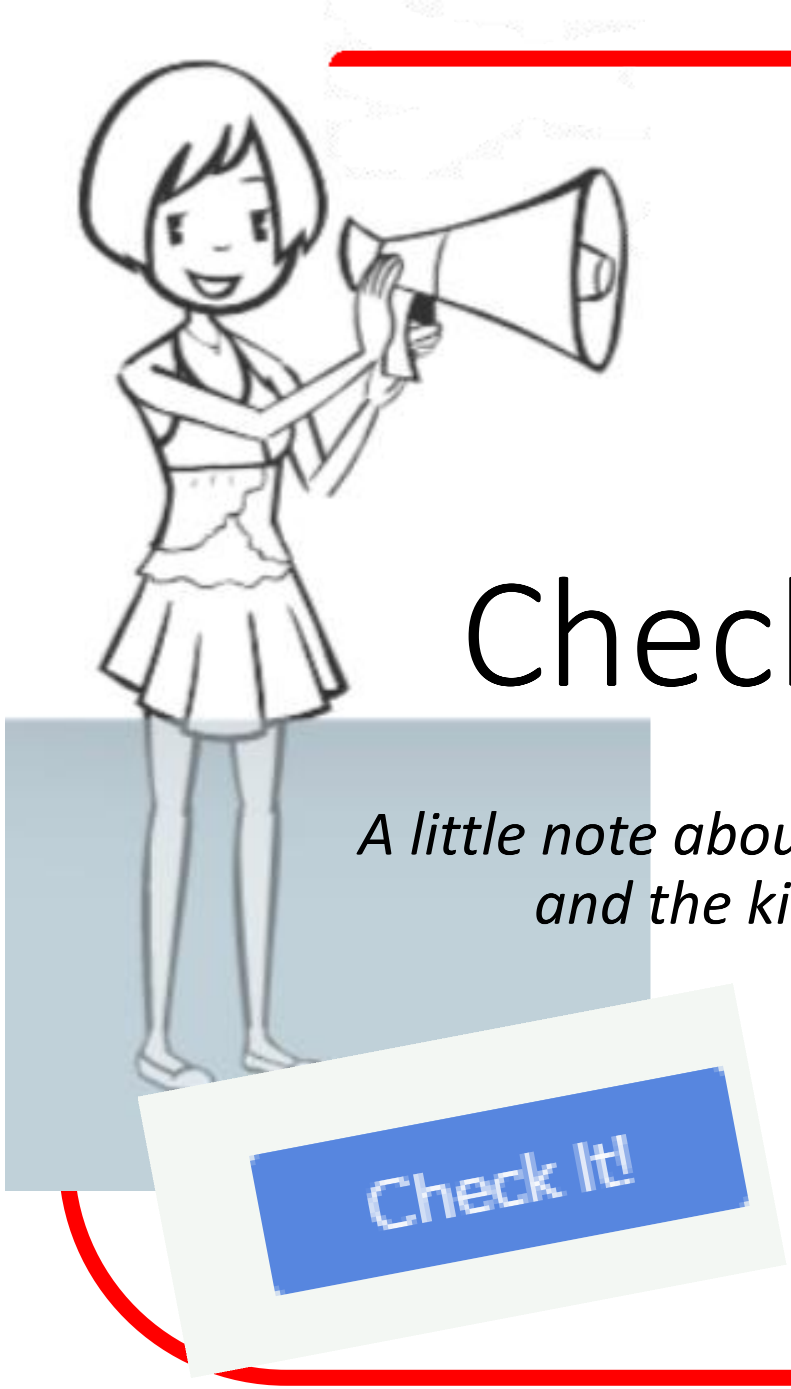


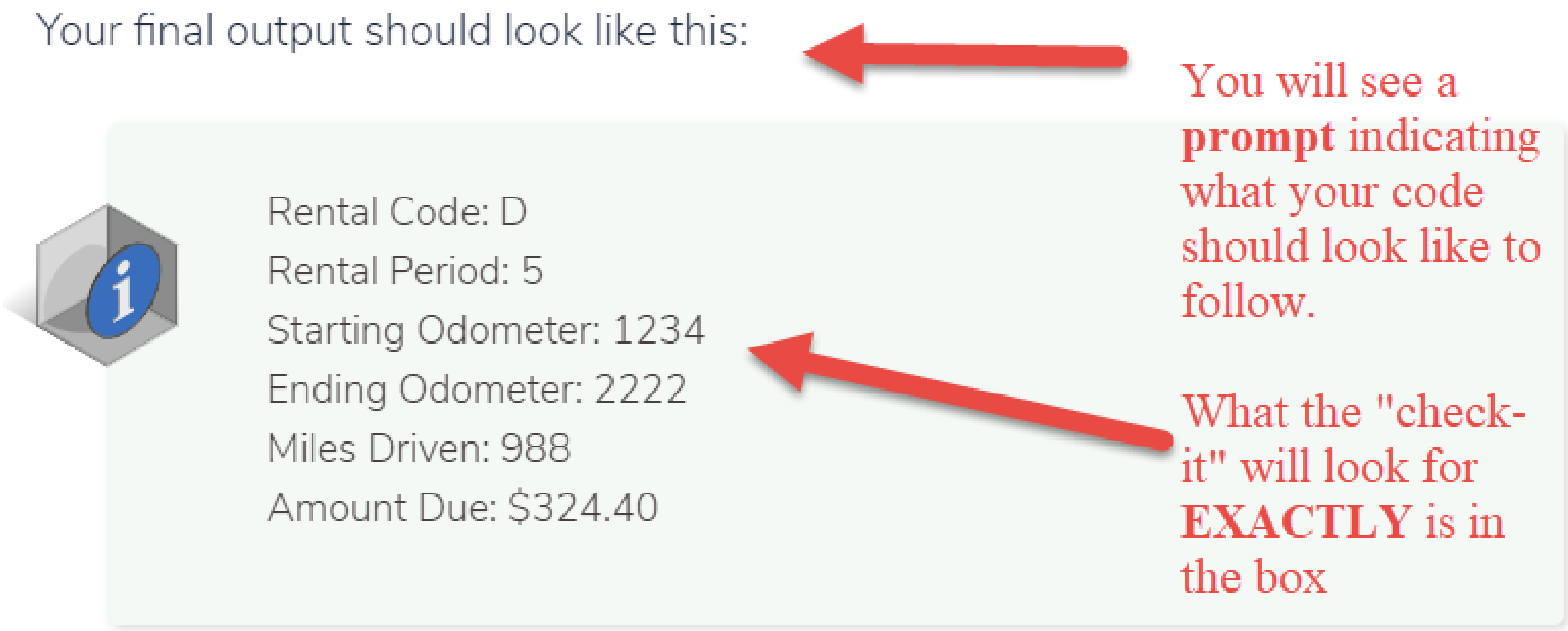
Check-its and Help-Me's!

A little note about check-its, how they work, what they are looking for and the kinds of error's and/or feedback you can expect.

Check It!

1 Let's start with Check-Its!

How it works – you will be guided on what it is you are supposed to be doing, you will then be shown what the OUTPUT of your program should produce.

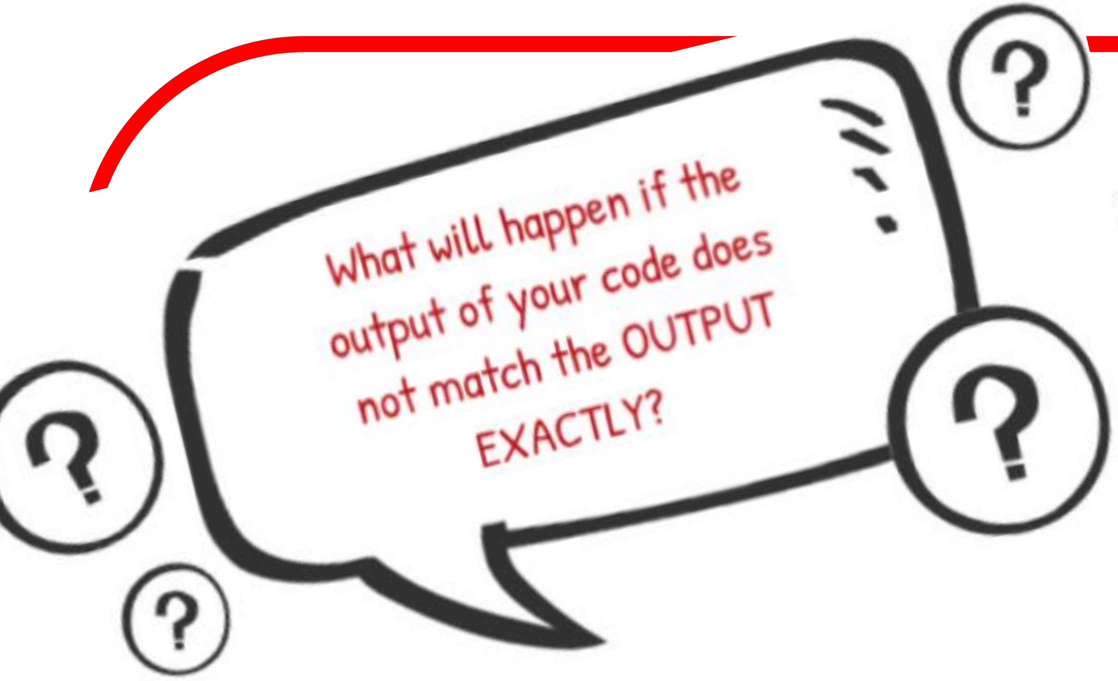


Your final output should look like this:

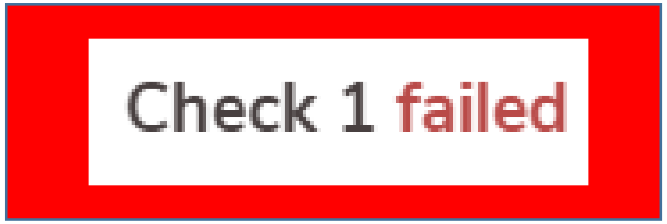
```
Rental Code: D
Rental Period: 5
Starting Odometer: 1234
Ending Odometer: 2222
Miles Driven: 988
Amount Due: $324.40
```

You will see a **prompt** indicating what your code should look like to follow.

What the "check-it" will look for **EXACTLY** is in the box


What will happen if the output of your code does not match the OUTPUT EXACTLY?
?
?
?
?

You will get this message>>>>>



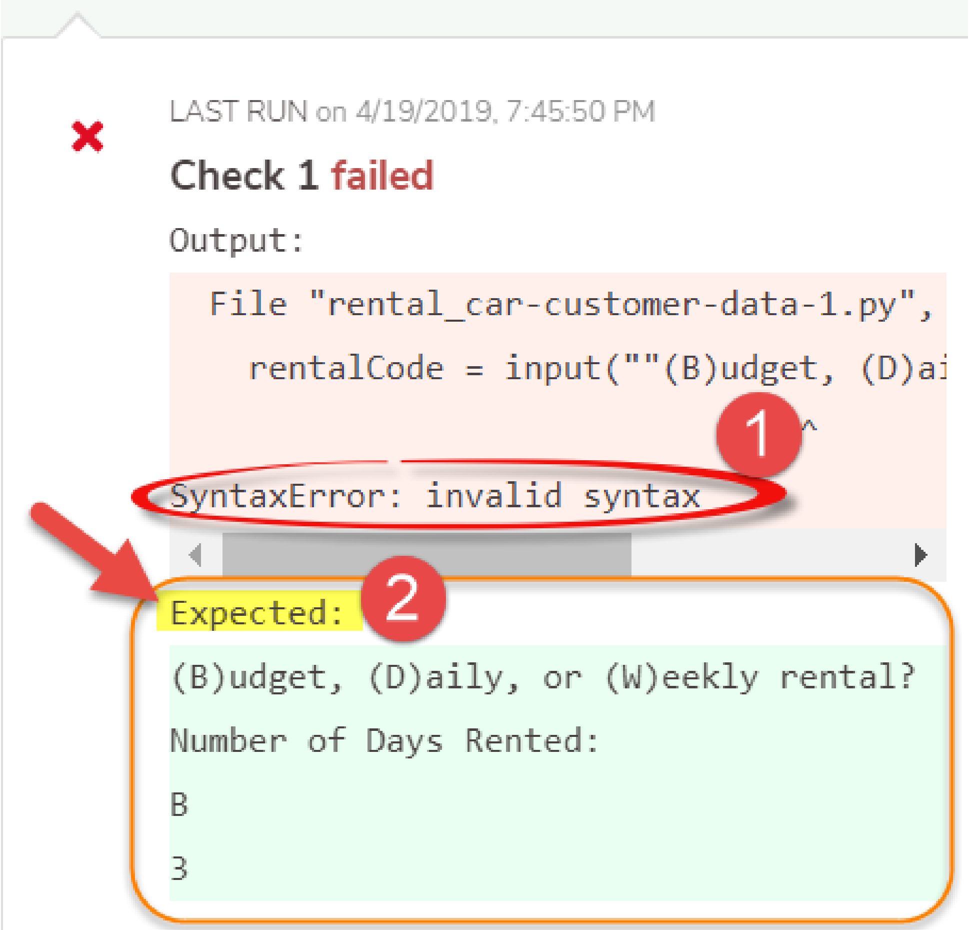
Check 1 failed

Not a fun message to get, especially when you are first starting to learn how to code, right? So what exactly COULD it mean?

There are usually two reasons a check-it fails:

1) The code didn't run because you had some sort of syntax error in your code.
2) The OUTPUT of the code you have written does not match EXACTLY what the check-it is looking for

(1) What IS a syntax error anyway?



LAST RUN on 4/19/2019, 7:45:50 PM

✖ Check 1 failed

```
Output:
  File "rental_car-customer-data-1.py",
    rentalCode = input(""(B)udget, (D)ai
                                       ^
SyntaxError: invalid syntax
```

```
Expected:
(B)udget, (D)aily, or (W)eekly rental?
Number of Days Rented:
B
3
```

This is an example of a syntax error. A syntax error means there is something in the code that is confusing the system so it won't run. A syntax error is like making a spelling mistake, or forgetting to put the period at the end of a sentence. Code works much the same way.

1 The "check-it" will show SyntaxError: telling **WHAT** the error is and **WHERE** the error is located.

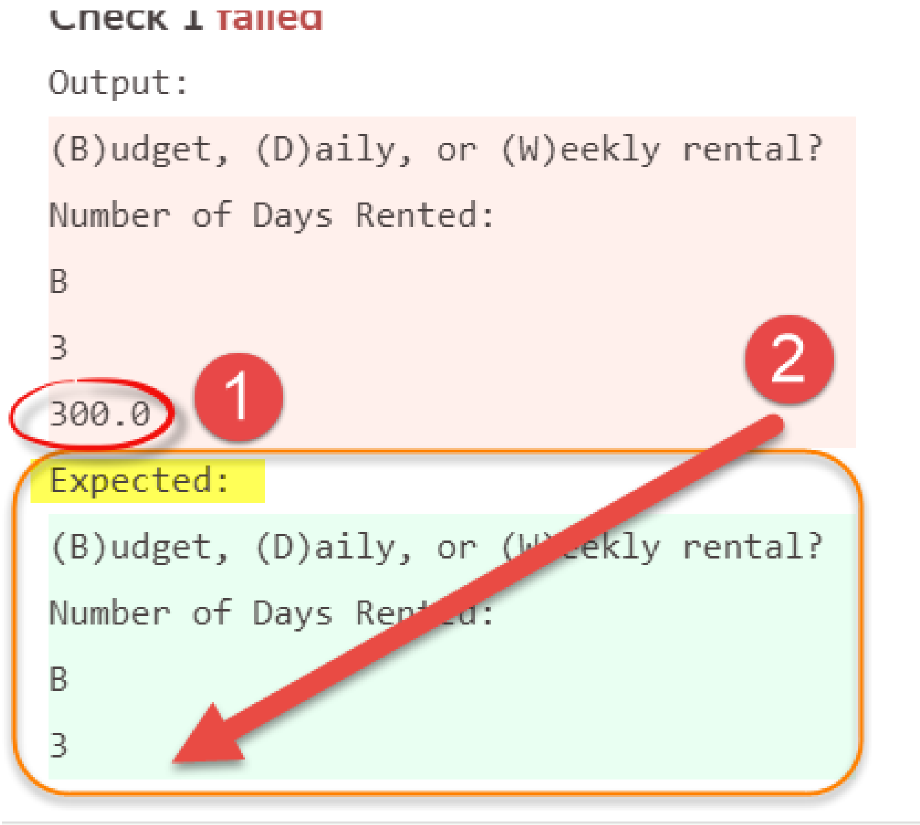
```
Output:
File "rental_car-customer-data-1.py", line 10
     rentalCode = input(""(B)udget, (D)aily, or (W)eekly rental?') ^

SyntaxError: invalid syntax
```

In this particular example, the input statement has a (") at the beginning of the statement, and a (') at the end. That is considered a syntax statement because the two don't match. String literals in python are surrounded by either single quotation marks, or double quotation marks.

'hello' is the same as "hello". As long as they match, you can almost always use either.

(2) Whaddya mean my OUTPUT doesn't match?

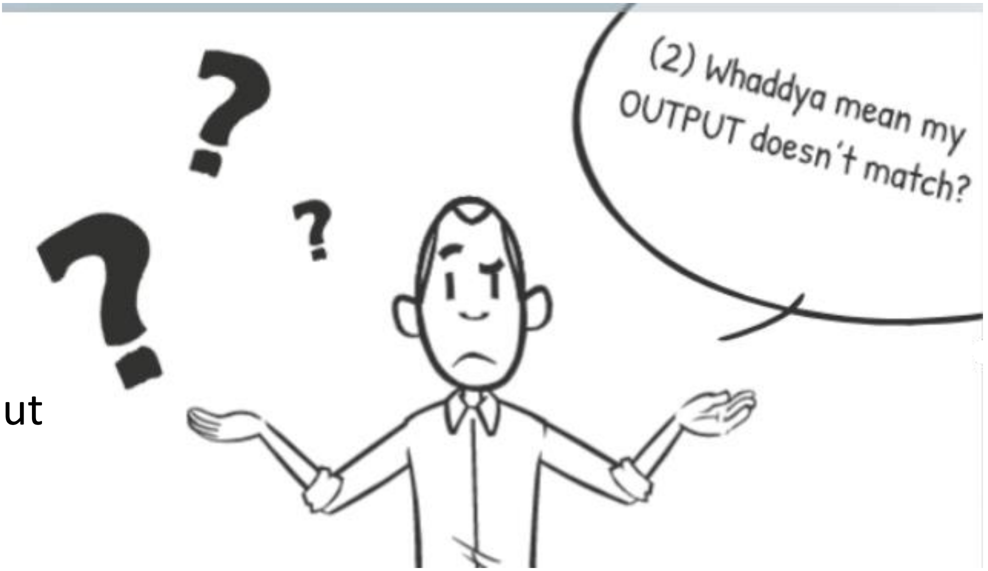


Check 1 failed

```
Output:
(B)udget, (D)aily, or (W)eekly rental?
Number of Days Rented:
B
3
300.0
Expected:
(B)udget, (D)aily, or (W)eekly rental?
Number of Days Rented:
B
3
```

This is an example when the OUTPUT your code generates does not match.

1 In this example, the code prints out an additional line of information: **300.0**



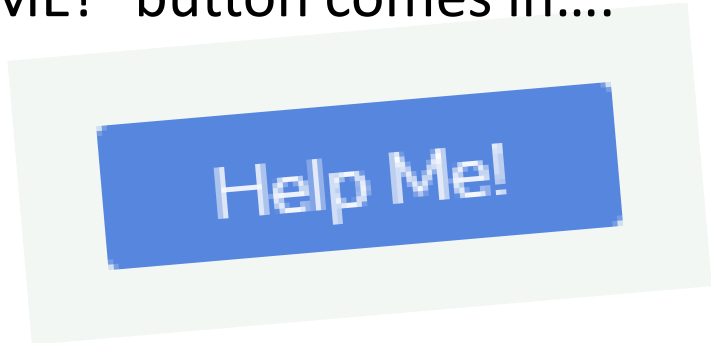
2 If the code you wrote doesn't produce the EXACT output asked for, the check-it will fail! In this case, the code written produces an extra piece of information – the **300.0**. That was NOT was expected.

Ok – so my check-it failed, now what?

IF you have a syntax error, you should review your code to make sure you spelled everything correctly, the names of variables are the same throughout, you matched (“) or (‘) etc.

If you discover that your output doesn’t match, see if you can figure out why.

….either way, how do I figure out how to fix it? That’s where the “HELP ME!” button comes in….



WHERE do these “Help Me!’s” Live?

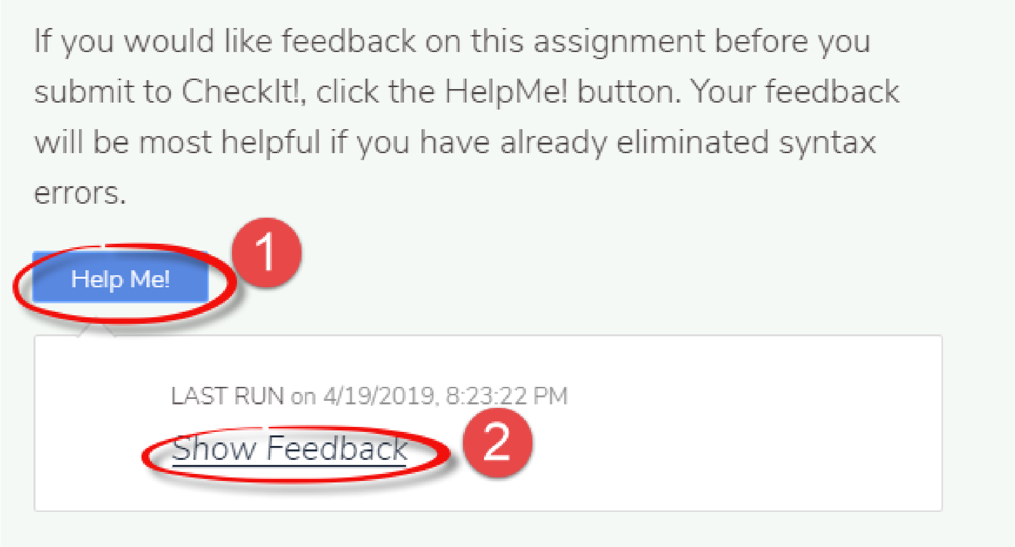
You will find a “Help Me!” toward the end of the assignment within Codio. You should do the check-it’s BEFORE clicking on the “Help-Me!” button!

How it works – IF you click on the “Help Me!” button, your code will be analyzed by our friend ***Jitt***– she’s what we call a Just In Time Tutor (thus the spelling of her name.) ***Jitt*** will match what you have written against what lots of other students have written, and will provide input and advice about your code. That feedback will include a review of the concepts that have been covered for that particular assignment along with some common mistakes others make with that concept, some examples, and some additional advice for how to fix code that isn’t working the way you want it to, and where to go for additional help and advice.

Can I see an example of how Help Me! Works?

This is an example of what the Help Me! button looks like.

Of course!



If you would like feedback on this assignment before you submit to CheckIt!, click the HelpMe! button. Your feedback will be most helpful if you have already eliminated syntax errors.

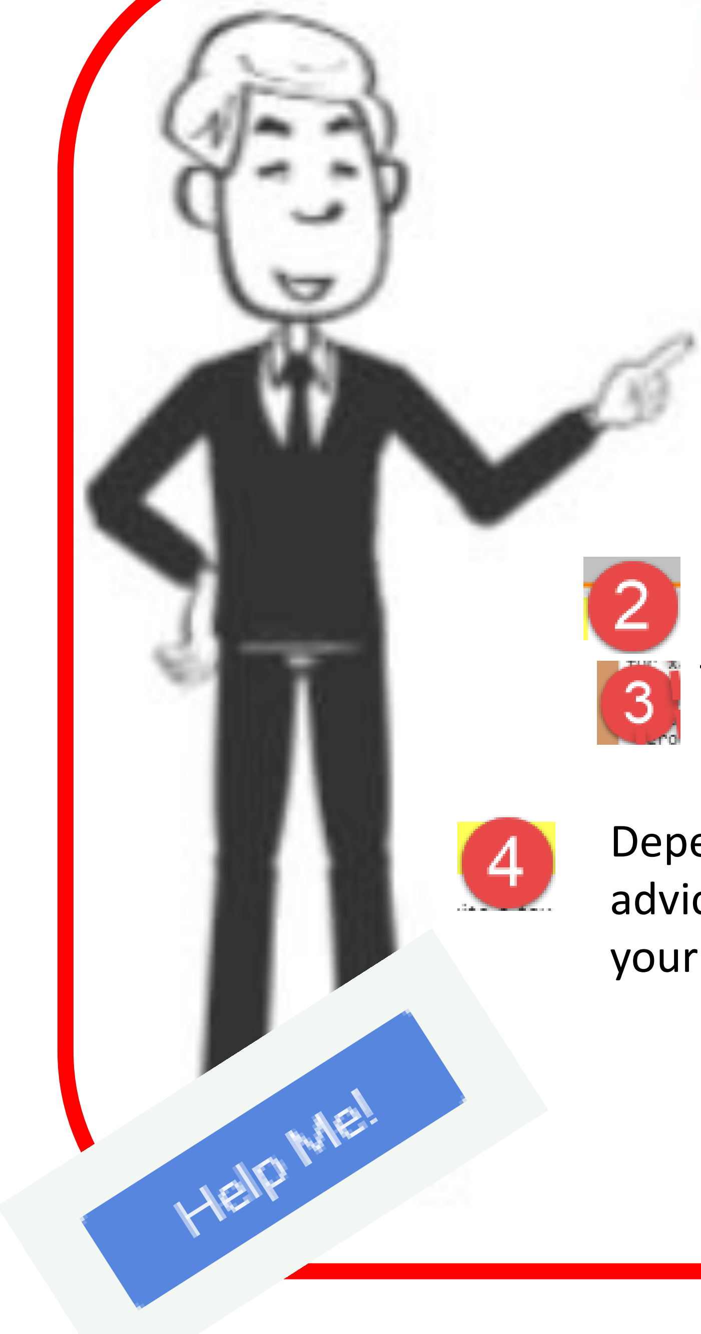
LAST RUN on 4/19/2019, 8:23:22 PM

1

Notice the blue "Help Me!" button – it looks almost the same as the "Check-it" button – the difference? It says "Help Me!" instead of Check-It! Lol!

After you have clicked the "Help Me!" button, it will take a few seconds to process, then a link will appear that says "Show Feedback"

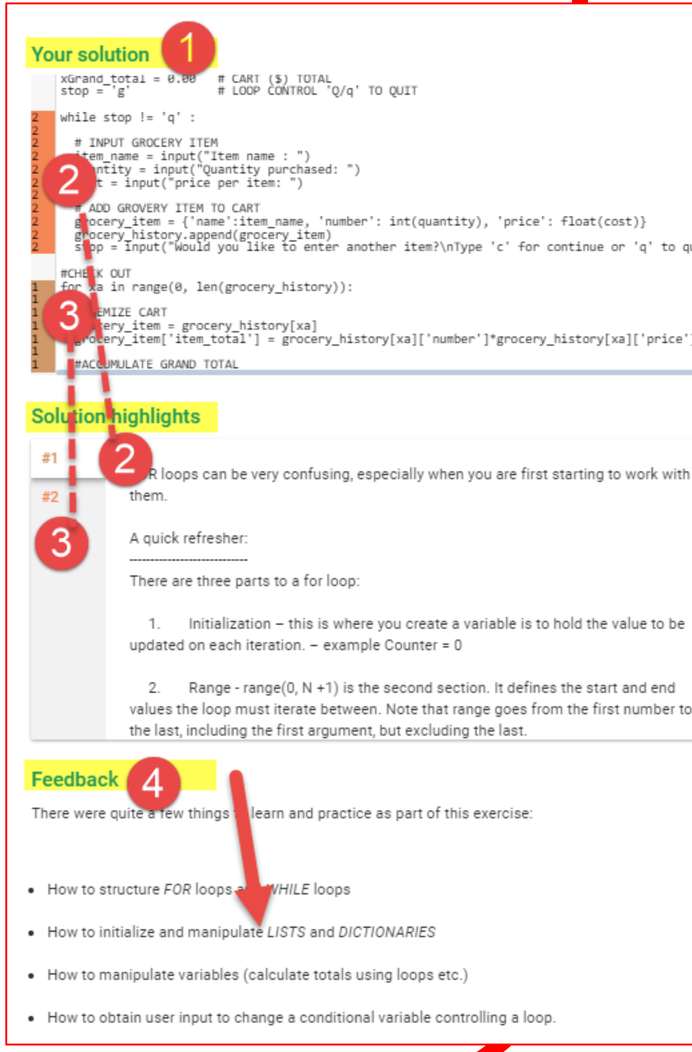
What will I see if I click "Show Feedback"?



1 A copy of the code you have written will show up at the top and be labeled: "**Your Solution**"

2 3 "**Your solution**" (*code*) will have different color highlights – in this example, there are highlights for **two** areas of code. The TWO and the THREE within the code are highlighted under the "solution highlights" further down.

4 Depending upon "**Your Solution,**" you will be given different advice and recommendations for fixing your code, or making your code more robust.



Your solution 1

```
  xGrand_total = 0.00   # CART ($) TOTAL
  stop = 'g'            # LOOP CONTROL 'Q/q' TO QUIT

2 while stop != 'q' :
2
2   # INPUT GROCERY ITEM
2   item_name = input("Item name : ")
2   quantity = input("Quantity purchased: ")
2   cost = input("price per item: ")
2
2   # ADD GROVERY ITEM TO CART
2   grocery_item = {'name':item_name, 'number': int(quantity), 'price': float(cost)}
2   grocery_history.append(grocery_item)
2   stop = input("Would you like to enter another item?\nType 'c' for continue or 'q' to qu

  #CHECK OUT
1 for xa in range(0, len(grocery_history)):
1
1   # ITEMIZE CART
1   grocery_item = grocery_history[xa]
1   grocery_item['item_total'] = grocery_history[xa]['number']*grocery_history[xa]['price']
1
1   #ACCUMULATE GRAND TOTAL
```

Solution highlights

FOR loops can be very confusing, especially when you are first starting to work with them.

A quick refresher:

There are three parts to a for loop:

1. Initialization – this is where you create a variable is to hold the value to be updated on each iteration. – example Counter = 0

2. Range - range(0, N +1) is the second section. It defines the start and end values the loop must iterate between. Note that range goes from the first number to the last, including the first argument, but excluding the last.

Feedback 4

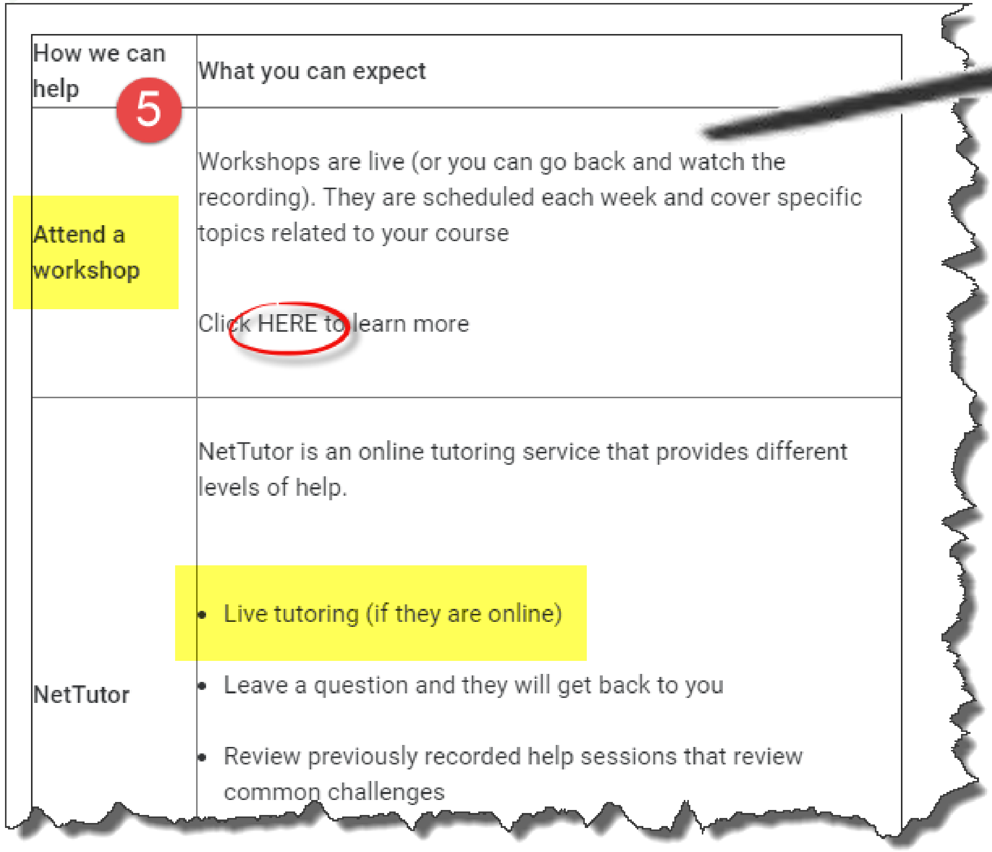
There were quite a few things to learn and practice as part of this exercise:

- How to structure *FOR* loops and *WHILE* loops
- How to initialize and manipulate *LISTS* and *DICTIONARIES*
- How to manipulate variables (calculate totals using loops etc.)
- How to obtain user input to change a conditional variable controlling a loop.

Help Me!

Keep scrolling – there is MORE HELP available!

5



How we can help	What you can expect
Attend a workshop	Workshops are live (or you can go back and watch the recording). They are scheduled each week and cover specific topics related to your course Click HERE to learn more
NetTutor	NetTutor is an online tutoring service that provides different levels of help. • Live tutoring (if they are online) • Leave a question and they will get back to you • Review previously recorded help sessions that review common challenges

NOTE: Always check in with your instructor first! They are your best source of help! HOWEVER – if they aren't available or if their recommendations STILL don't help, we also have an additional support team at the ready to lend a hand!