365 DataScience Linear search in Python

Step 1 Create a function that performs a linear search on a list

```
# Create a function that checks whether a certain item ('item') is cont
ained in a list ('my_list').
# The time it takes to go through a list grows linearly with respect to
 the size of the list.
# Therefore, the complexity of the search is O(n).
# This means that the method is not particularly efficient and is good
for small datasets.
def linear_search(item, my_list):
    i = 0
    # Create a boolean variable which initially assumes that the item i
s not found.
    found = False

    # While the iterator 'i' is strictly smaller than the lenght of the
 list and the item is still not found,
    # perform the operations in the body of the loop.
    while i < len(my_list) and found == False:

        # Check if the i^th item in the list is equal to the item we ar
e searching for.
        if my_list[i] == item:

            # If it is, then we set the boolean variable to True.
            # This will break out of the loop as the second condition w
on't be met.
            found = True

        # If the i^th item is not equal to 'item', then we increase the
 value of 'i' by one
        # and continue our search.
        # Once we reach the end of the list without success, 'i' will b
e set to the lenght of the list.
        # This will break out of the loop as the first condition won't
be met.
        else:
            i = i + 1

    # Return True if the item is in the list.
    # Return False if the item is not in the list.
    return found
```

Step 2 Check if the function performs as expected

```
# Create a list of numbers
test = [6, 5, 8, 2, 3, 45, 87, 24, 70]

# Check if the number 4 is inside this list.
print(linear_search(4,test))

# Check if the number 87 is inside this list.
print(linear_search(87,test))
```