

W4.2 Applied

Correct the code 1

Identify what is wrong with the given `for` loop, and rewrite the loop to reflect the intended behaviour (i.e., provided as a comment in first line of the Python file).

Correct the code 2

Identify what is wrong with the given `for` loop, and rewrite the loop to reflect the intended behaviour (i.e., provided as a comment in first line of the Python file). Please do not use `replace()` in your code.

Predict the outcome of the following computations by hand

In this activity, please attempt to predict the outcome of the following code *without using the Python interpreter*.

Question 1 Submitted Mar 21st 2022 at 8:31:49 pm

```
my_list = ['1045', '1008', '2004', '2099']  
x = my_list[1:3][-1]  
print(x)
```

☐ 1045

☐ 1008

☒ 2004

☐ 2009

Question 2 Submitted Mar 21st 2022 at 8:36:18 pm

```
x = ['monkey', 'tiger', 'lion', 'mouse']  
animal = x  
animal[3] = 'meerkat'  
print(x)
```

☐ ['monkey', 'tiger', 'meerkat', 'mouse']

☒ ['monkey', 'tiger', 'lion', 'meerkat']

☐ ['monkey', 'tiger', 'lion', 'mouse', 'meerkat']

☐ ['monkey', 'tiger', 'lion', 'meerkat', 'mouse']

☐ ['monkey', 'tiger', 'lion', 'mouse']

Question 3 Submitted Mar 21st 2022 at 8:39:20 pm

```
my_list = ['purple', 'blue', 'grey']  
x = my_list  
my_list.append('red')  
print(x)
```

- ☐ ['purple', 'blue', 'grey']
- ☐ ['red']
- ☐ ['purple', 'blue', 'red']
- ☒ ['purple', 'blue', 'grey', 'red']

Question 4 Submitted Mar 21st 2022 at 8:40:37 pm

```
x = ['breakfast', 'lunch', 'dinner']  
my_list = x  
print(my_list.count('a'))
```

- ☒ 0
- ☐ 1
- ☐ 2
- ☐ 3

Question 5 Submitted Mar 21st 2022 at 8:47:16 pm

```
my_list = ['3', 'a', '5', '2', 'b', 'd']  
count = 0  
for i in range(len(my_list)):  
    if my_list[i].isalpha(): # isalpha() method returns True if all the characters are alphabet let  
        count += 1  
        my_list[i] = False  
print(my_list)
```

- ☐ ['3', 'a', '5', '2', 'b', 'd']
- ☐ ['3', 'a', '5', '2', 'b', 'd']
- ☐ [False, 'a', False, False, 'b', 'd']
- ☒ ['3', False, '5', '2', False, False]

Count Binary

Write a program in the Python file named `count_binary.py` that takes a string representation of a binary number as input, and outputs the number of 1s in the number (i.e., inputting '1001011' would output 4). Please do not use `count()` method for this exercise.

When collecting the input, please prompt the user with the following message:

"Please enter a binary string: "

When providing the output `out`, please use the following message:

"Number of 1s in the input string is: out"

Find the Index of the Minimum Number

Write a program in the Python file named `min_index.py` that takes a list of numbers as input and outputs the index of the minimum number. Please do not use `min` function for this exercise. If there is more than one number with the minimum value, output the smallest index. Please assume the input string from the user has the format `[num_1,num_2,...,num_k]` for k numbers.

When collecting the input, please prompt the user with the following message:

"Please enter a list of numbers: "

When providing the output `out`, please use the following message:

"The min index is: out"

Find the Maximum Absolute Difference

Write a program in the Python file named `max_abs_difference.py` that takes a list of numbers as input, and outputs the maximum absolute difference among all pairs of numbers from the input list. Please assume the input string from the user has the format `[num_1,num_2,...,num_k]` for k numbers.

When collecting the input, please prompt the user with the following message:

"Please enter a list of numbers: "

When providing the output `out`, please use the following message:

"The maximum absolute difference is: out"

For example, for the input `[10.1,35.7,3.3,4.2,14.1,-15.9]`, the output should be `51.6` because $|35.7 - (-15.9)| = 51.6$.

Find the Nearest Cities

Write a program in the Python file named `nearest_cities.py` that uses the given lists of x-coordinates (list of numbers), y-coordinates (list of numbers) and names of cities (list of strings) as input, and outputs the names of the two cities that are closest to each other based on their [Euclidean distance](#). As a reminder, the Euclidean distance of two points (x_1, y_1) and (x_2, y_2) is calculated based on the following formula:

$$dist = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

When providing the outputs `out1` and `out2`, please use the following message:

"The two cities that are closest to each other are: out1 and out2"

For example, given:

- x-coordinates `[10.1,0.4,6.9,4.2,14.1,15.9]`
- y-coordinates `[17.7,3.1,9.0,2.7,41.5,33.3]`
- (imaginary!) city names `['Elabro','Nutunyu','Barimba','Duduth','Tromsu','Pranho']`

the output should be `The two cities that are closest to each other are: Nutunyu and Duduth`

Feedback

Question 1

Feedback

What worked best in this lesson?

No response

Question 2

Feedback

What needs improvement most?

No response