

Assignment 3

Question 1 (Medium):

There are 5 problems in the following code which is trying to search for all occurrences of the target in a list of numbers. Find all 5 problems and create a good version that works.

```
numbers = [6, 12, 17, 23, 38, 45, 77, 84, 90]
target = 23
index = 1
found = False
for number in numbers:
    if number == target:
        found = False
        print("Found, target, "at index", index)
if not found:
    print("Your target was not found...")
```

The correct output should look like this:

```
Found 23 at index 3
```

Hint: You can obviously just copy this code and paste it into google CoLab (you might have to fix the indentation a little bit, but that's not part of the 5 problems). If you're stuck, don't forget to go through the session 5 of [notes](#).

Question 2 (Medium):

Take an input from the user, convert the input to an integer, and then print the cube of that number. Take an input again, convert it to an integer again, and then print its cube again. Continuously do this until the input given is -1.

The output should look something like this:

```
Give me a number to cube(-1 to exit): 2
The cube of 2 is 8
Give me a number to cube(-1 to exit): 5
The cube of 5 is 125
Give me a number to cube(-1 to exit): 10
The cube of 10 is 1000
Give me a number to cube(-1 to exit): -1

You entered -1. Loop ended.
```

Hint: Look at session 6 of this CoLab: [notes](#)

Question 3 (Hard):

Same Backstory: Lets say that you are a programmer for the Govt. of Delhi, India. The Govt. of Delhi has recently filed a law about even and odd number plates on cars in Delhi. Even and odd cars are allowed to stay on road alternatively to reduce pollution. They have already placed cameras to figure out the number on the plate of all cars. And you are given this number as an integer (n).

Question: Given an integer n , and a string day ('even' or 'odd') as parameters of a function, return `True` if a person with number plate n is allowed to drive today and return `False` if the number plate is not eligible to drive today.

Basically, create a function with the following details:

Parameters: n (an integer), day (a string: 'odd' or 'even')

Return: Boolean (*True* if number plate n is eligible to drive on the day , *False* otherwise)

Hint: Look at session 6 of this CoLab: [notes](#)

Question 4 (Hard):

Type this in your python shell or in google CoLab:

```
matrix = [[0, 1, 2, 3],  
          [4, 5, 6, 7],  
          [8, 9, 10, 11],  
          [12, 13, 14, 15]]
```

Use nested loops(do **not** use the in-built python function) to find the sum of each row in this matrix.

So the output should look like:

```
Row 0 sum: 6  
Row 1 sum: 22  
Row 2 sum: 38  
Row 3 sum: 54
```

Hint: Look at session 6 of this CoLab: [notes](#)