Prelab 2

We encourage you to work together on the Pre Lab. The Pre Lab is not graded but will help you prepare for your lab session. In the Pre Lab, you may find questions to answer. We do not require you to provide us the answers, but we do recommend you to try to answer these questions. If you have any questions on the material in the pre lab, first check the book and recitation slides, if you do not find your answer please email your recitation TA or the course instructors.

Importing Libraries

We can import python libraries in 2 ways.

- **import <library name here>**. This statements imports a specific module or package from the library that we need to use. When using this import method the library name will always be appear with the package.

For the purpose of this lab we need to use the math library. An example of that can be.

```python
import math ← imports the math library
x = 25
print(math.sqrt(x))
```

The following code produces the result as

```console
>>>5.0
```

In this we see that `math` is the library name and `sqrt` is the package we are want to use. The , connects the two.

- **from <library name> import **

* in programming generally means everything. So the statement can be read as, from <library name> import everything. Using this method produces the same result but with a little difference in the code.

```python
from math import * ← imports the math library
x = 25
print(sqrt(x))
```

The following code produces the result as

```console
>>>5.0
```

As you can see in the code above, we have the same result but we did not use the word `math` in front of `sqrt`. That is because we imported all the packages in the math library and can use them directly.
Nested if-else conditions

While you program, there may be a situation when you want to check for another condition after a condition resolves to true. In such a situation, you can use the nested if construct.

Simple If-else statement

```python
def main():
    x = int(input("Enter an integer: "))
    if(x == 10):
        print("The value is 10")
    else:
        print("The value is not 10")
main()
```

Nested if-else

In the code above. If we input the value as 10, the program prints “The value is 10”. Otherwise it prints “The value is not 10”. Now suppose we want to make another decision if the value is 10. If the value is 10, the program should take another input from the user. If the value of second input is 5, it prints “The value is 5”. Otherwise it prints “The value is not 5”. This can be done in the following way.

```python
def main():
    x = int(input("Enter an integer: "))
    if(x == 10):
        y = int(input("Enter second input: "))
        if(y == 5):
            print("The value is 5.")
        else:
            print("the value is not 5.")
    else:
        print("The value is not 10 and did not enter the nested if.")
main()
```

OUTPUT if we enter value as 10 the first time.

Enter an integer: 10
Enter second input: 5
The value is 5.

OUTPUT if we enter any other value than 10 the first time.

Enter an integer: 8
The value is not 10 and did not enter the nested if.

The above code uses nested if-else. So the format for using nested if condition looks like.

```python
if expression1:
    statement(s)
    if expression2:
        statement(s)
```
statement(s)
else
  statement(s)
else:
  statement(s)