So far, we know quite a bit about constructing logical statements with Pythin, such as if/else/elif statements, for and while loops, checking if an item is in a list or not in a list.

Now lets see how we can perform these operations within a function.

Recall the mod operator % which returns the remainder after division, if a number is even then mod 2 (%2) should be == to zero

In [1]:
```
2 % 2
```

Out[1]:
```
0
```

In [2]:
```
20 % 2
```

Out[2]:
```
0
```

In [3]:
```
3 % 2
```

Out[3]:
```
1
```

In [4]:
```
41 % 40
```

Out[4]:
```
1
```

In [5]:
```
20 % 2 == 0
```

Out[5]:
```
True
```

In [6]:
```
41 % 40 == 0
```

Out[6]:
```
False
```

Now, lets construct this to a function.

In [10]:
```
def even_check(number):
    result =  number % 2 == 0
    return result
```

In [11]:
```
even_check(2)
```

Out[11]:
```
True
```

In [13]:
```
even_check(21)
```

Out[13]:
```
False
```

We can also achieve what is pictured above by performing the following:

In [14]:
```
def even_check(number):
    er % 2 == 0
```

In [15]:
```
even_check(2)
```

Out[15]:
```
True
```

In [16]:
```
even_check(21)
```

Out[16]:
```
False
```

Return true if any number is even inside a list

Now, lets check if any number in a list is even: Let's return a boolean indicating if any number in a list is even. Notice here how return breaks out oif the loop and exits the function:

In [17]:
```
def check_even_list(num_list):
    #Go through each number
    for number in num_list:
        #Once we get a hit on an even number, we return True
        if number % 2 == 0:
            return True
        # Otherwise we do not do anything
        else:
            pass
```

In [20]:
```
check_even_list([1,3,5])
```

In [21]:
```
check_even_list([2,4,5])
```

Out[21]:
```
True
```

In [22]:
```
check_even_list([2,1,1])
```

Out[22]:
```
True
```

In [23]:
```
check_even_list([1,3,6])
```

Out[23]:
```
True
```

Notice, if there is not an even number, it returns nothing.
If we wanted it to return false, we can perform the following:

In [25]:
```
def check_even_list(num_list):
    #Go through each number
    for number in num_list:
        #Once we get a hit on an even number, we return True
        if number % 2 == 0:
            return True
        # Otherwise we do not do anything
        else:
            pass
    return False
```

In [26]:
```
check_even_list([1,3,5])
```

Out[26]:
```
False
```

In [27]:
```
check_even_list([2,4,6])
```

Out[27]:
```
True
```

NOTICE HOW THE 'RETURN FALSE' lines up with the first FOR statement, indention is important if we want to check the whole list as return will break out of the function once a condition is met.

Now, lets return all even numbers in a list, if there are no even numbers, return an empyty list.

In [32]:
```
def check_even_list(num_list):
    #return all the even numbers in a list
    even_numbers = []
    #even numbers is a placeholder, that we will call later on in the code
    #GO through each number
    for number in num_list:
        #Once we a hit on an even number, we append the even number
        if number % 2 == 0:
            even_numbers.append(number)
        #Dont do anything if it is not even
        else:
            pass
        #Notice the indention, this ensures we run through the entire for loop
    return even_numbers
```

In [29]:
```
check_even_list([1,2,3,4,5,6])
```

Out[29]:
```
[2, 4, 6]
```

In [ ]:
```

```