Errors and Warning Messages

```
In [23]: def DoSum( Value1, Value2): return Value1 + Value2
```

```
In [24]: DoSum()

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-24-ddda916e57d2> in <module>()
----> 1 DoSum()

TypeError: DoSum() missing 2 required positional arguments: 'Value1' and 'Value2'
```

Trying DoSum() with just one argument would result in another error message. In order to use DoSum() you must provide two argument.

```
In [82]: ("Value equals 5!") if Value == 6: print("Value equals 6!") else: print("Value is something else.") print("It equals" + str(Value))

  File "<ipython-input-82-3c96d7f99010>", line 1
    def TestValue(Value): if Value == 5: print("Value equals 5!") if Value == 6: print("Value equals 6!") else: print("Value i
s something else.") print("It equals" + str(Value))
                                                 ^
SyntaxError: invalid syntax
```

```
In [84]: def TestValue(Value):
             if Value == 5:
                 print('Value equals 5!')
             elif Value == 6:
                 print('Value equals 6!')
             else:
                 print('Value is something else.')
                 print('It equals ' + str(Value))
```

The code in Python needs to have the indentations, it probably won't work if we put it in one line.

```
for Value1 in MyTuple:
    if type(Value1) == int:
        print Value1
    else:
        for Value2 in Value1:
            if type(Value2) == int:
                print "\t", Value2
            else:
                for Value3 in Value2:
                    print "\t\t", Value3
```

Missing parenthesis after “print”

```
In [27]: for Value1 in MyTuple:
             if type(Value1) == int:
                 print (Value1)
             else:
                 for Value2 in Value1:
                     if type(Value2) == int:
                         print ("\t", Value2)
                     else:
                         for Value3 in Value2:
                             print ("\t\t", Value3)
```

```
In [125]: n = 2
          with open("Colors.txt", 'r') as open_file:
              for j, observation in enumerate(open_file):
                  if j % n==0:
                      print('Reading Line:' + str(j) + 'Content' + observation)

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-125-399da6d99786> in <module>()
      3     for j, observation in enumerate(open_file):
      4         if j % n==0:
----> 5             print('Reading Line:' + int(j) + 'Content' + observation)

TypeError: can only concatenate str (not "int") to str
```

Cambie de Anaconda a Command Prompt

```
In [1]: n = 2
        with open("Colors.txt") as open_file:
            for j, observation in enumerate(open_file):
                if j % n== 0:
                    print('Reading Line:' + str(j) + ' Content: ' + observation)

Reading Line:0 Content: Color   Value

Reading Line:2 Content: Orange  2

Reading Line:4 Content: Green   4

Reading Line:6 Content: Purple  6

Reading Line:8 Content: White   8
```