FUNCTIONS

"She can't do that ... shoot her or something!" -Nute Gunray (Trade Federation Viceroy)

WHAT IS A FUNCTION?

A functions is a resuable piece of code that is only run when it is called or executed.

```
#functions
yoda_quotes = ["wars not make one great.","You will be..You wi
print(yoda_quotes[0])

len(yoda_quotes)


type(yoda_quotes)
```

You call or excecute a function by writing the name and the the () after the name.

The coma seperated items inside the () are called the arguments.

```python
#simple function
def droid_count(start_count, destroyed_count):
    return start_count - destroyed_count

print(droid_count(100,23))

reamining = droid_count(2322,122)
print(reamining)


print(droid_count(12344))#errors because of no 2nd argument
```

Create Functions by adding def (define) and the name of the function, then any parameters needed for the function then a colon (:).

The return statement "sends" the value of the folloing expression out to the rest of the code. It is not required, many functions that 'just do something' do not have a return.

Functions that have a return value are able to have that return value assigned to a varaible or used right then as a literal.

```
yoda_quotes = ["wars not make one great.","You will be..You wi

#using external items
def count_quotes():
    print(len(yoda_quotes)) #kinda useless
    #no return?

count_quotes()

def better_count_quotes(quote_list):
    return len(quote_list)

    print('The quote list is complete.')

#accepts the argument instead
print(better_count_quotes(yoda_quotes))
```

- After the return statement, the function will exit ignoring ny other code.
- It's possible to use external information in a function. But it is normally un-advisble.
- Scope is the code that can be accessed. Functions have their own scope called "local scope". Variables not in a function are called "globally scoped"
- With python local scoped variables will not modify global scoped