New York University
School of Continuing and Professional Studies
Division of Programs in Information Technology

Introduction to Python
Exercises, Session 9

Ex. 9.1 Write a function callme() that prints "function called" every time you call it. Call the function with the following code:

```
callme()
callme()
callme()
```

Expected Output:

```
function called...
function called...
function called...
```

Ex. 9.2 Write a function printupper that takes one argument, a string, a prints that string uppercased. Call it with the following code:

```
printupper('hello')
printupper('my you are loud')
printupper('I am not loud.  You are.')
```

Expected Output:

```
HELLO
MY YOU ARE LOUD
I AM NOT LOUD.  YOU ARE.
```

Ex. 9.3 Write a function addme that takes two arguments, adds them together and returns the two arguments added / concatenated. Call it thusly:

```
x = addme(4, 5)
print(x)

y = addme('hey', 'you')
print(y)
```

Expected Output:

```
9
heyyou
```

Ex. 9.4 Write a function printlist that takes a list and loops through and prints each element of the list. Call it thusly:

```
printlist([1, 2, 'a', 'b'])
```

Expected Output:

```
1
2
a
b
```

Ex. 9.5 Create a module file named yourname.py where yourname is your first name. Do not put a shebang (!#) line at the top.

Create a def hello: function (that prints hello, world!) inside the yourname.py module.

Now in the same directory where you saved the module, create a python script. In the script have an import yourname statement, and then call the function through the module: yourname.hello()

Save the file and then run it.

Expected calls and output:

```
import yourname

yourname.hello()           # hello, world!
```

Ex. 9.6 Modify the above function to include an optional argument. If name=[something], print hello, [something]! instead of hello, world! But if the name= parameter is not passed, revert to saying hello, world!

So your def hello function code will be modified to accept the name=text argument (i.e., def hello(name=False)), and then test to see if text has a value -- if it is True (i.e., if name: is True). If it is True, print it after 'hello, '. If it doesn't, print 'hello, world!'

Expected calls and output:

```
import yourname

yourname.hello()                # hello, world!
yourname.hello(name='Python')   # hello, Python!
```

Ex. 9.7 Create a function getlines(filename) that takes a filename, opens the file for that filename, copies the lines of the file (i.e., from readlines()) to a list variable, and then returns the list. In the calling code, call the function with a known filename, and assign the return value of the call to a variable. Loop through the variable (of course it is a list) and print out each line in the file.

Expected calls and output:

```
lines = getlines('../python_data/student_db.txt')

for line in lines:
  print(line)             # prints each line in file
```