

## Homework: Shell Scripting (Graduate)

### hello.sh

Create a bash script called “hello.sh” with the following behavior:

- If there are no arguments, it should print a “Usage” and exit *with an error code*
- Your program will expect to receive a “greeting” in \$1 and possibly a name in \$2; if there is no second argument, use “Human” as the default
- Print the greeting, a comma and space, the name, and an exclamation point

Here is how it should look:

```
$ ./hello.sh
Usage: hello.sh GREETING [NAME]
$ ./hello.sh "Top o' the morning"
Top o' the morning, Human!
$ ./hello.sh "Greetings" "Earthling"
Greetings, Earthling!
```

### gap.sh

Write a bash script that will print out the files in the “biosys-analytics/data/gapminder” directory. Note that to be portable for testing purposes, you will need to use a **relative** path from the directory where the script lives (hint: start with \$PWD). Your program will do the following:

- If there are no arguments, print out all the *basenames* of the files in sorted order
- If there is an argument, treat it like a regular expression and find files where the basename matches at the beginning of the string in a case-insensitive manner and print them in sorted order
- If no files are found, print a message telling the user

```
$ ./gap.sh | head -5
1  Afghanistan
2  Albania
3  Algeria
4  Angola
5  Argentina
$ ./gap.sh 1
1  Lebanon
2  Lesotho
3  Liberia
4  Libya
```

```
$ ./gap.sh [y-z]
    1 Yemen_Rep
    2 Zambia
    3 Zimbabwe
$ ./gap.sh x
There are no countries starting with "x"
```

## Testing

You have been provided a **Makefile** that will run a test suite. This is what it should look like when all tests are passing:

```
$ make test
python3 -m pytest -v test.py
===== test session starts =====
platform darwin -- Python 3.7.0, pytest-3.8.0, py-1.6.0, pluggy-0.7.1 -- /anaconda3/bin/python
cachedir: .pytest_cache
rootdir: /Users/kyclark/work/biosys-analytics/assignments/02-bash-scripting-grad, inifile:
plugins: remotedata-0.3.0, openfiles-0.3.0, doctestplus-0.1.3, arraydiff-0.2
collected 5 items

test.py::test_exists PASSED [ 20%]
test.py::test_usage PASSED [ 40%]
test.py::test_hello_too_many PASSED [ 60%]
test.py::test_hello PASSED [ 80%]
test.py::test_gap PASSED [100%]

===== 5 passed in 0.46 seconds =====
```

## Commit

Remember that I can't pull your work until it's been pushed it to GitHub.

```
$ git add hello.sh gap.sh
$ git commit -m 'homework 2 grad' hello.sh gap.sh
$ git push
```