UF UNIVERSITY of FLORIDA
Herbert Wertheim College of Engineering

COMPUTING FOR LIFE
COP3504C

Lab 03: The Cow Says…

Overview

This lab is designed to introduce students to the Bash Command Line Interface (CLI) and the concept of CLI arguments and give them practice writing classes. For (only) this lab, you are not allowed to use any windowed editor (such as IntelliJ, Eclipse, or Notepad++). Instead, you will use a Unix-based command line editor.

The **`cowsay`** utility is a popular Unix program from the 20th century (see https://en.wikipedia.org/wiki/Cowsay). You will write a slightly simplified **`cowsay`** program that takes in several arguments and prints out different text depending on the arguments. You must log in to the school computers using your CISE account to do this lab.

Tools

Please note that **you must use a text editor and the terminal to edit and run your program and its directories.** It is advised students learn/review basic Unix shell commands before beginning; a good run-through can be found here: https://linuxjourney.com/lesson/the-shell.

Follow these steps to get started on the lab:

1) Open a terminal and enter the **`pwd`** command to identify the path to the working (current) directory (folder)
2) Enter **`ls`** to list the contents of the current directory
3) Use the **`mkdir`** command to make a new directory called Cow*Lab*.
4) Use **`ls`** to see the change, then **`cd`** to change to the directory Cow*Lab*.
5) Do your lab work in that folder. Use your googlefu skills to find more commands.

Recommended text editors include **`nano`** and **`joe`**. (If you hate yourself, you can use **`vi`** or **`emacs`** too.) You'll also need to use the **`python3`** program to execute Python programs. You can also use its interactive shell for testing.

You can read more information about some of these commands here:
https://www.howtogeek.com/howto/42980/the-beginners-guide-to-nano-the-linux-command-line-text-editor/
https://introcs.cs.princeton.edu/java/15inout/linux-cmd.html

History of Command Line Names (Just for Fun)

In case you are wondering… **`nano`** is a free-software version of a program called **`pico`**, which is the stand-alone editor for Pine at the command line. The **`pine`** email program was an alternative version of Elm – you see, Pine Is Not Elm. The elm program was an earlier email manager. (If you think my jokes are bad, you cannot fathom the depths of terrible humor present in computer science and engineering! This is but the beginning of a terrible humor journey.)

Specification

Students will construct one module which (**`cowsay`**) which will contain a **`main()`** method as the entry point (for command line execution) and a data class (**`Cow`**). Note: **`heiferfactory.py`** is provided for you; your code must use this module to create cow objects. All attributes / methods must be private unless noted in the specification!

Provided For Students - heiferfactory

`get_cows``() -> *list[Cow]*`
Returns an array of cow objects from the built-in data set. This will call the **`Cow`** constructor and **`set_image()`** methods of the cow class to properly initialize new cow objects uniquely for each data set.

Entry Point

Your program must accept command line arguments. Command line arguments are captured as part of the **`args`** variable in the built-in **`sys`** module. (Review lecture slides for examples!)

The command line arguments that must be supported are as follows:

`python3 cowsay.py -l`	Lists the available cows
`python3 cowsay.py MESSAGE`	Prints out the MESSAGE using the default COW
`python3 cowsay.py -n COW MESSAGE`	Prints out the MESSAGE using the specified COW

If a user calls for a cow that does not exist, the program should print out "Could not find ***[COWNAME]*** cow!"

Output Samples

```
>python3 cowsay.py I am a potato!

I am a potato!
    \
     \
      \
        ^__^
        (oo)\_______
        (__)\       )\/\
            ||----w |
            ||     ||
```

```
>python3 cowsay.py -n kitteh I am pritteh.

I am pritteh.
    \
     \
      \
       ("`-'  '-/") .___..--' ' "`-._
         ` *_ *  )    `-.   (      ) .`-.__. `)
        (_Y_.) ' ._   )   `._` ;  `` -. .-'
     _.. `--'_..-_/   /--' _ .' ,4
  ( i l ),-''  ( l i),'  ( ( ! .-'
```

```
>python3 cowsay.py -l
Cows available: heifer kitteh
```

```
>python3 cowsay.py -n ninja WUT
Could not find ninja cow!
```

Suggested Methods

The following methods are suggested to make development easier, but are not required:

`_list_cows``(cows: list[Cow])`
Displays the list of available cows from an array of **`Cow`** objects.

`_find_cow``(name: str, cows: list[Cow])`
Given a name and an array of **`Cow`** objects, return the **`Cow`** object with the specified **`name`**.

Cow Class

The **`Cow`** class facilitates the creation and use of cow objects by providing the following methods (which students must implement):

`__init__``(name: str)`
This method should be the **only** constructor for this class. There should not be a default constructor!

`get_name``() -> *str*`
Returns the name of the cow.

`get_image``() -> *str*`
Returns the image used to display the cow (after the message).

`set_image``(_image: str)`
Sets the image used to display the cow (after the message).

Testing

Run the **`cow_test.sh`** script provided. To run it, place it in the same folder as the rest of your files and make it executable via **`chmod +x cow_test.sh`**. Then run it: **`./cow_test.sh`**. This script contains a series of commands to run **`cowsay.py`** in various ways, streaming the results into the text file **`output.txt`**. It then uses **`less`** command to view the output. Your output file should match the sample output in this document.

Submissions

NOTE: Your output must match the example output *exactly*. If it does not, ***you will not receive full credit for your submission***!

Files: cowsay.py
Method: Submit on Canvas

Sample Output

```
>python3 cowsay.py Hello World!

Hello World!
    \
     \
      \
        ^__^
        (oo)\_______
        (__)\       )\/\
            ||----w |
            ||     ||

>python3 cowsay.py -n kitteh Hello World!

Hello World!
    \
     \
      \
       ("`-'  '-/") .___..--' ' "`-._
         ` *_ *  )    `-.   (      ) .`-.__. `)
         (_Y_.) ' ._   )   `._` ;  `` -. .-'
       _.. `--'_..-_/   /--' _ .' ,4
   ( i l ),-''  ( l i),'  ( ( ! .-'

>python3 cowsay.py -l
Cows available: heifer kitteh

>python3 cowsay.py -n ninja Hello world!
Could not find ninja cow!

>python3 cowsay.py Hello -n kitteh

Hello -n kitteh
    \
     \
      \
        ^__^
        (oo)\_______
        (__)\       )\/\
            ||----w |
            ||     ||
```