

CMPSCI 119
LAB #1 – Bar Graphs
Professor William T. Verts

The goal of this first Python programming assignment is to successfully create, enter, debug, and run a simple program in the JES environment. If you have not done so already, you must first install the JES environment on your own personal computer. Please set up a special folder on your desktop or in your documents folder to hold all your Python projects. I also suggest that you buy a flash drive (SanDisk recommended) and put a folder for your Python projects there as well; the flash drive can then be used for backup and for bringing code in for us to help you debug.

Background

In this assignment you are going to write a program to enter a set of grades for students who have taken an exam, and then generate a set of text-based bar graphs based on those grades. For example, if student Fred received a 78 on his exam, the line of output for Fred will be 78 copies of the | character (the vertical bar), followed by a space, his name, another space, and his score. When your program runs, the expected output from JES will look like the following image:

```
>>> Main()
Enter Name ---- Fred
Enter Score --- 78
Enter Name ---- Sam
Enter Score --- 34
Enter Name ---- Mary
Enter Score --- 97
Enter Name ---- Carol
Enter Score --- 69
Enter Name ---- Joe
Enter Score --- 82
Enter Name ---- Bob
Enter Score --- 46
Enter Name ---- Sue
Enter Score --- 89
Enter Name ----
||||| Fred 78
||||| Sam 34
||||| Mary 97
||||| Carol 69
||||| Joe 82
||||| Bob 46
||||| Sue 89
>>> |
```

Setting up the Assignment

In the JES environment, write a comment at the top of your program containing *your own name* and the lab number. For example, I would put in a comment as the first line of text:

```
# William T. Verts - Lab #1
```

After that line, type in the following program code framework just as you see it here, and then save it in your Python folder with `Lab1.py` as the filename. For now, leave the gray areas blank; you will write your own code there later.

```
JES - Untitled
File Edit Watcher MediaTools JES Functions Window Layout Help
1 def GetANameAndScore ():
2   Result = [ "", 0]
3   [Gray box: Your code goes here]
4
5
6   return Result
7
8 def Process (Name, Score):
9   S = ""
10  [Gray box: Your code goes here]
11
12
13  return S
14
15 def Main():
16  Names = []
17  Scores = []
18  [Gray box: Your code goes here]
19
20
21
22  return
23
```


Task #3 - Main

Now we need to complete the main program, which is in two parts.

The first part of `Main` is the interactive portion, which asks the user for names and scores by calling the `GetANameAndScore` function many times in a loop. Each time `GetANameAndScore` returns a list containing a non-blank name (regardless of the score), the name is added to the end of the `Names` list and the score is added to the end of the `Scores` list. Only when the user enters a blank name does this process stop, and the program will then proceed on to the second part of `Main`.

The second part of `Main`, once the `Names` and `Scores` lists have been constructed, is to call the `Process` function for every set of associated values from those two lists, and in each case print out the result returned from `Process`. If you've done everything correctly, `Names` and `Scores` will contain the same number of elements, but your program must not make any assumptions about what that number is (i.e., you cannot assume a constant for the length of the lists – your program **MUST** be adaptable to any number of students).

This function also requires the use of a loop; for this assignment I want you to again use a `while`-loop (not a `for`-loop).

Notice that "Fred" is the first name to be entered, and 78 is the first score to be entered. These values will be the first elements of `Names` and `Scores`, respectively. These will be the first values passed to the `Process` function. The result of that call is printed. Then, in the second call to `Process`, your program must use the second values from `Names` and `Scores` ("Sam" and 34), in the third call to `Process` it uses the third values from `Names` and `Scores` ("Mary" and 97), and so on for as many items as there are in the lists.

Running the Program

Run the program by typing `Main()` at the command prompt.

You should be asked for names and scores. Use the following values: Fred=78, Sam=34, Mary=97, Carol=69, Joe=82, Bob=46, and Sue=89. After Sue's name and score have been entered, enter a blank string to terminate the data entry process. You should then get a histogram as shown in the screenshot on the first page.

If the output does not contain the correct information, debug your program and run it again. Continue to edit and test your program until the resulting output does contain the correct information.

Finishing Up

When you are finished and everything runs correctly, go to the class site and click on the link for submitting lab assignments. In JES select all the text, copy it to the clipboard, in the Web page paste the text into the program area of the submission form, fill in your name and ID number and the lab number in the appropriate slots, and then submit the assignment for grading.

The graders will score your program by running it to see if it correctly generates the appropriate graphs. You will also be graded on efficiency and completeness, and the graders will be explicitly looking to see if your program will work for an arbitrary number of students (number of entries in `Names` and `Scores`), that it does not ask for a score if the name is blank, and so on.