

## CISC106 Summer 2011 Lab07

- This lab and all subsequent labs will be due Sunday at 11:55 PM EDT on Sakai.

### Preparation (do not submit for grading)

1. Start up the python interpreter and enter the following statement:  
`fout = open('newfile.txt', 'w')`.<sup>1</sup> You've just successfully opened a file for writing. You can write strings to the file by calling `fout.write('some string')`.<sup>2</sup> Try writing some things to the file. You can jump to a new line in the file by calling `fout.fwrite('\n')`. Write some strings to the file on a few different lines, and then close it by calling `fout.close()`
2. Now re-open `newfile`, but instead open it for reading this time. You can do this with the statement `fin = open('newfile.txt', 'r')` Enter the following statement: `for line in fin: print line` and see what happens. You can now close `newfile.txt`
3. Copy the following bit of code into the interpreter:

```
try:
    raise RuntimeError
    print 'Does this message show up at all?'
except RuntimeError:
    print 'This message had better show up.'
```

Carefully examine what happens.

### Programs (to be graded)

1. Download the files `ff7.py`, `lab07_tests.py`, `save00.ff7` and `save05.ff7` from the course website. Now create a `lab07.py` and make sure you import `ff7` in it. Implement a function `make_row` which takes a string and then returns another string which is a list of *Comma-separated values*.<sup>3</sup> The function should work such that the test for it in `lab07_tests.py` passes.
2. Now you should implement a function `make_save_csv` which takes a string `fname` denoting a path to a file. It will write a CSV file containing all of the saves in the two files `save00.ff7` and `save05.ff7` to the file at `fname`.<sup>4</sup> N.B. that you'll need to open the two save files as *binary files*. You can do this by giving the call to open the appropriate flag, e.g. `fin = open('save00.ff7', 'rb')`. The `b` in the second parameter of course stands for binary. Also note that you should write a header to your CSV file before writing any of the saves. The header is just a comma-separated list of column names, e.g. `'Name,Level,Location'` would be a three column header where the column names are

---

<sup>1</sup>If you already have a file called `newfile.txt` in your `pwd`, it will be overwritten. Be careful.

<sup>2</sup>this call will write `some string` to `newfile.txt`

<sup>3</sup>for more information on Comma-separated values (or CSV) files, see [http://en.wikipedia.org/wiki/Comma-separated\\_values](http://en.wikipedia.org/wiki/Comma-separated_values)

<sup>4</sup>You can get the string representation of a save by calling `ff7.read_save_string` and passing it the open save file.

Name, Level and Location, respectively. You should make sure to put your column titles in the correct order, which you know based on how you implemented `make_row`.<sup>5</sup>

3. Finally, you should implement a `print_csv` which takes a string `fname`, opens the CSV file at `fname` and prints out a summary for each of the file's rows. Use the format in the example below as a guideline for how your summaries should look.

```
Name: Yuffie
Level: 99
Current HP: 9999
Max HP: 9999
Current MP: 999
Max MP: 999
Character 1: Yuffie
Character 2: Cloud
Character 3: Cid
Gil: 7730384
Location: "Midgar Area"
```

Make sure you eventually call `close()` on any file you `open()`. You should submit your `lab07.py`, the CSV file you created, and any other docs required by your TA on Sakai.

---

<sup>5</sup>If you have Excel, OpenOffice.org or some other spreadsheet program, you can open your CSV file in it to view it like a spreadsheet.