HOMEWORK #9 - REGEX AND WEBCRAWLING
REGULAR VERSION

PART I - MOCHA

First, create a CSV file called **results.csv**

Next, create a text file called **ldaps.txt** and fill it with a list of at least 7 ldaps like this:

alberthwang, lassetjt, eguth, bridgetc, tiffanyliu, smadaan, satishm

Finally, write a script called **profiles.py** that will read this file and loop through the LDAPs. For each LDAP, it should crawl the user's orginfo page, extract the user's information, and populate **results.csv** so that it looks like this (with these fields):

ldap	fullname	title	managerldap	department
alberthwang	Albert Hwang	Internal Tools Developer	ninaye	People Operations
lassetjt	Lasse Thorenfeldt	People Analyst	abeer	People Operations
eguth	Elizabeth Guth	Engineering Staffing Researcher	amho	People Operations
bridgetc	Bridget Campbell	Learning Technology Specialist	sgiri	People Operations

Please place all relevant assignment files your **python > week9** directory. Before submitting, make sure all executable code is encapsulated in a `main()` function as described in the notes.

PART II - THE IMPORTANCE OF BEING EARNEST

Please copy the files here into your own directory:

`/home/alberthwang/python/week9/hw_files/wilde_earnest`

In this directory, you will find 7 text files that together comprise the first act of The Importance of Being Earnest by Oscar Wilde. This is downloaded HTML from the enotes.com version.

Please write a program that goes through all of these files and answers the following questions:

- Which character has the most lines?
- Which character is the most verbose? (says the most words)
- Which character asks the most questions? (look for frequency of “?”)
- Which character makes the most exclamations? (look for frequency of “!”)

A few hints:

- Since “.” matches all character except new line, just remove the new lines from the contents as you read them in (\n for unix, \r for mac, \r\n for PC)
- Watch out for whitespace!
- Look for patterns in the text files before starting. All the text files are formatted the same way.
- Do your best to strip out the stage directions and vocabulary links in the dialogue when doing your calculations, but don't kill yourself over it.

Please place all relevant code in a module named `earnest_parser.py` in your **python > week9** directory. Before submitting, make sure all executable code is encapsulated in a `main()` function as described in the notes.

PART III - OPTIONAL - EXTRA CREDIT+ - TOTAL INFLUENCE

HIGHLY recommended for those in data-centric roles.*

Write a script, that given an LDAP derives a Googler’s total influence including the total reportees (direct or indirect), the number of offices this person’s reportees are in, and the number of cost centers this person’s reportees belong to. Also break the person’s reportees down by these types - employees, interns, temps, vendors, and contractors.

Here is sample output:

```
Enter an ldap: schandra

Total Reportees: 170
Total Offices: 4
Total Cost Centers: 8
Total Employees: 102
Total Interns: 0
Total Temps: 56
Total Vendors: 12
Total Contractors: 0
```

A few notes:

- A reporting hierarchy is an N-level tree. Do not assume that there is a static number of

layers going up to any individual.
- Do **NOT** run the script for larry or an OC member with thousands of reportees (it would take forever to finish running and the orginfo team will get mad at me). Just run on managers that have a couple hundred reportees at most.
- Often those we think of as contractors actually have an employee type of temp (I'm actually not sure what the difference is myself).

Please place all relevant code in a module named `total_influence.py` in your **python > week9** directory. Before submitting, make sure all executable code is encapsulated in a `main()` function as described in the notes.