Lab: eBook Library List

CSI 3305: Introduction to Computational Thinking

November 12, 2010

1 Introduction

One of the strengths of a computer is the ability to make time consuming tasks easier. This is especially true of repetitive tasks. Although each iteration of a repetitive task may not take long, the time quickly compounds when you are forced to perform the same task hundreds, or thousands, of times. Computational thinking can help us avoid this type of unnecessary work by leveraging a computer's power to perform repetitive tasks fairly quickly.

2 Problem Statement

For this lab, imagine that you are part of an eBook sharing group, who lend one another eBooks and request books from other members. As part of the group, your responsibility is to create a listing of the eBooks you have available to lend and send this list to the other group members. The problem is that you have over two-hundred eBook titles in an "eBooks and PDFs" folder, which contains both ePub eBooks (which are those the group shares), as well as PDF books, which the group does not share. You could open up a text editor and manually type the names of the books as you scan the directory, only including ePub books; however, this is time consuming, boring, and error prone. A better approach is to create a small script to list the files in the directory and filter out those you do not want to include.

Your task is to create two scripts, one a command-line batch file and the second a short python script, for listing out your eBook library.

3 Tools

You will be using the Windows command prompt and Python programming language for this lab. The scripts will be written using a text editor, and will be run using the command line.

4 Setup and Programming

4.1 DOS Batch File

1. Download the eBook library zip file from the course website. Unzip the file and save to a local folder.
2. Within that folder, create a new file called list.bat

3. Open the list.bat in the text editor of your choice (such as Notepad or Vim.)

4. Enter the following code in the file:

   ```
   dir *.epub
   ```

   This line calls the **dir** utility, which lists out the contents of a directory. We include a filter, ‘*.epub’, which only lists files ending with the extension .epub. This allows us to exclude the PDF files contained within the same directory.

5. Save the file.

6. Open the command line (From the Windows menu bar: Start > Run... > cmd). Navigate to the folder where you created your file (use the following command: cd “C:\Folder\where file\is”.)

7. Call the batch file from the command prompt, using the following command and hitting enter:

   ```
   list.bat
   ```

   The result should be a list of files, along with additional information concerning the directory and files. Next, we’ll simplify the output, to exclude that additional information.

8. Open the list.bat file again, and modify the command within it to look like this:

   ```
   dir *.epub /a-d /b
   ```

   The first switch excludes directories from being printed and the second excludes the additional file information.

9. Save the file and run again from the command prompt. You should now get a simplified listing of the ePub eBooks.

10. We will now save this information to a text file, by using redirection of output. To do so, run the following command:

    ```
    list.bat > myLibrary.txt
    ```

11. Open the newly created file myLibrary.txt and verify that it contains a listing of your ePub eBooks.

4.2 Python Script

An alternative way of completing the same task is to use a programming language like Python. Although we were able to complete the task using simple batch commands, we may wish to do some processing to the filenames in our list, such as creating a separate list of authors. We will assume that all books have the following naming convention: Title - Author.epub

1. Within the same folder, create a new file called authors.py

2. Open the authors.py in the text editor of your choice (such as Notepad or Vim.)

3. Enter the following at the beginning of the file:

   ```
   import os

   authors = set()
   ```

The first line will import the os module, which will allow us to get a listing of the files in the directory. The second creates a local set variable for storing the names of authors.

4. Next we will walk over all files, processing them as we encounter them. Enter the following code:

   ```
   for root, dirs, files in os.walk("."):
       for name in files:
           if name[-5:].lower() == '.epub':
               pieces = name.split(" - ")
               a = pieces[1].strip()[:-5].title()
               authors.add(a)
   ```

   Note: Pay attention to the indention in the code, as Python is whitespace sensitive. You will get an error if the code is not properly indented.

   The first line allows us to iterate over everything contained in the current directory (denoted by ".").
   We next iterate over all files, and check that their file extensions are equal to .ePub. We do this by indexing the name string, beginning with the fifth character from the end (denoted by [-5:]). This will give us all characters beginning with the fifth from the end, all the way to the end. If the extension equals to .ePub, we process the filename and add the author to our list.

   To process the name, we first split on the hyphen surrounded by spaces. The produces a list of the pieces, stored in the *pieces* variable. We then use the second piece, and strip whitespace from it, and take all characters up to the last 5 (which removes the file extension). This becomes our author name. We also title case the name, so that irregularities in the capitalization of author names do not produce redundant author names.

   Lastly, we add the name to our set of author names.

5. We then create a list from our set, and sort it:

   ```
   authorsList = list(authors)
   authorsList.sort()
   ```

6. Finally, we output the list of authors to the standard output:

   ```
   for author in authorsList:
       print author
   ```

7. Save and close this file.

8. From the command line, type the following to run your program and hit enter:

   ```
   python authors.py
   ```

   You should see a list of authors printed to your screen.

9. We will now save this information to a text file, by using redirection of output. To do so, run the following command:

   ```
   python authors.py > myAuthors.txt
   ```

10. Open the newly created file myAuthors.txt and verify that it contains a listing of authors in your ePub library.

5 Questions

1. List all authors with first names that begin with J in the library.
2. What are the benefits of using the DOS batch file method?
3. What are the benefits of using the Python script method?
4. What Python function allows you to list all files in a directory?