

Programming Project 05

This assignment is worth 40 points (4.0% of the course grade) and must be **completed and turned in before 11:59 on Monday, February 22, 2010**. This is a two week deadline because of the midterm on Feb 18.

Assignment Overview

The learning objectives of this project are to further increase your skills with loops and conditionals, as well as working more with strings. This project will also introduce the use of an external library and the use of lists

Background

The “We Feel Fine” project, located at <http://www.wefeelfine.org> has recently gotten a lot of press. Go there, take a look, then come back here. They describe their project as follows:

At the core of We Feel Fine is a data collection engine that automatically scours the Internet every ten minutes, harvesting human feelings from a large number of blogs. Blog data comes from a variety of online sources, including LiveJournal, MSN Spaces, MySpace, Blogger, Flickr, Technorati, Feedster, Ice Rocket, and Google.

We Feel Fine scans blog posts for occurrences of the phrases "I feel" and "I am feeling. ... Once a sentence containing "I feel" or "I am feeling" is found, the system looks backward to the beginning of the sentence, and forward to the end of the sentence, and then saves the full sentence in a database.

Once saved, the sentence is scanned to see if it includes one of about 5,000 pre-identified "feelings". This list of valid feelings was constructed by hand, but basically consists of adjectives and some adverbs. The full list of valid feelings, along with the total count of each feeling, and the color assigned to each feeling.

Our goal is to provide a text interface to their “feelings” discovery process and report on the results of that search.

The Project Interface and Python’s Toolbox

The project provides a general way to interact with the gathered data using simple web address requests; meaning you just need to send a long html address to the site to gather the data. So there are two problems to solve: constructing the html address properly and using Python’s toolbox to send the html and receive the data.

Constructing the html address

All requests to the website begin with the following html address string:

`‘http://api.wefeelfine.org:8080/ShowFeelings?display=text&returnfields=sentence&limit=1500’`

We then add on to the end of the address the various data we would like to collect. Each query is a string of the form `‘&element=value’`, where the `‘&’` is required. There are many queries we could use, but the project will require only three:

1. The person’s age. The form is `‘&agerange=num’` where `num` is a value evenly divided by 10 (i.e., 10, 20, 30). Thus `‘&agerange=50’` would gather data on everyone who had posted feelings and whose age was from 50-59. The system only allows these ranges.

2. The person's gender. The form is '`&gender=num`' where 1 means male and 0 means female (sorry ladies, I didn't make the interface). Thus a valid query is '`&gender=0`', meaning only female feelings.
3. The city the person is from. The form is '`&city=cityString`', where the `cityString` is any city. Thus a valid query is '`&city=chicago`', meaning only feelings from people in Chicago.

A full, valid html address then for all females from 50-59 from Chicago would be:

```
'http://api.wefeelfine.org:8080/ShowFeelings?display=text&returnfields=sentence&limit=1500&gender=0&city=chicago&agerange=50'
```

There are many other fields. Some in the above address include: a limit of 1500 sentences (the max the site will provide), providing the entire sentence in the returned data and providing the results as text (as opposed to XML). Look at <http://www.wefeelfine.org/api.html> for more info.

Copy and paste the query above into a browser and see what you get.

Getting the data using Python

So we can get the data using a browser, but how do we get it using Python? Python provides a simple library to get data via the web (http) protocol. There is a program in the directory called

`simpleHTML.py` that demonstrates the process. Essentially you:

1. import the `urllib` module
2. make a connection using `connection=urllib.urlopen('htmlAddressString')`. You provide the html address string (like the one we just did above).
3. Gather all the data from the web page using `data = connection.read()`. The variable `data` now has the contents of the web page as a single string.
4. Then close the connection, `connection.close()`

Project Description / Specification

Here is a summary of what you have to do :

1. Prompt the user for the age, city and gender of the feelings that will be returned. Note a few things:
 - a. gender responses, male or female, must be turned into 1 or 0 respectively.
 - b. age responses must be turned into a number divisible by 10.
 - c. If the user chooses to not provide the data, then **do not** include that query in the web address.
 - d. If an illegal value is entered (like 'dog' for gender), then do not include the query.
2. The site provides a list of over 5000 feelings and their frequencies (see <http://www.wefeelfine.org/data/files/feelings.txt>). The first 25 feelings in `feelings.txt` are the "top25" feelings in order of frequency. For each feeling in the top25, you are to calculate and list the frequency of that feeling in the result of your request.
3. Besides the top25 feelings, prompt the user for any other feeling they wish to check for, as few or as many as they like.
4. The output of your program should be:
 - a. The constructed html address (so we can check its accuracy)
 - b. A list of the top25 feelings and their frequency, along with the added feelings and their frequency that the user provided

Deliverables

proj05.py -- your source code solution (include section, date, project number and comments).

Notes and Hints:

As always, think about breaking the problem down into. Here are some suggestions:

- Try using the urllib in a program to fetch data from an html address. Do this first in the IDLE shell then do it in a program.
- Try prompting for one of the three values and construct an html address. Print out the address and see if it looks right. Copy and Paste the address into a browser and see if it is valid.
- Combine one and two and look at the data that is returned. When you look at that data, you will see a couple of things:
 - It is one big string
 - The individual feeling sentences in that big string are separated by the character sequence '
'. You need to use the '.split()' method to gather the individual feeling-sentence strings as a list.
 - Each feeling-sentence string now has junk at the front and the back. Use the '.strip()' method to clean that up
 - Now you should have a list of feeling sentences
- You need to gather the top 25 feelings as strings in a list. You can type or cut and paste them into a list or read them from a file into a list—whatever works for you.
- As a test, you need to see if you can check any one individual feeling sentence to find which of the top 25 occurred in that sentence.
- If you can do that, then you need to find a way to count the occurrence of each of the top 25.
- Remember, even though you ask for 1500 responses, you may get less depending on the specificity of the request.
- Some of the feeling words may not occur in any sentences you retrieve.



```
Python Shell
>>>
===== RESTART =====
>>>
We'll use the top25 feelings, but if you want to add some, type in a comma separated list
Extra Feelings (comma separated, <enter> for none):cold,warm
Age (<enter> for all):53
Gender, male or female (<enter> for both):
Which city (<enter> for all):chicago
The request http://api.wefeelfine.org:8080/ShowFeelings?display=text&returnfields=sentence&
&limit=1500&agerange=500&city=chicago got 76 responses
better : 3
bad : 3
good : 3
guilty : 1
same : 1
sorry : 1
well : 2
down : 2
alone : 1
happy : 2
comfortable : 1
sad : 2
free : 2
lost : 1
stupid : 1
old : 4
home : 1
>>>
===== RESTART =====
>>>
```

```
Python Shell
We'll use the top25 feelings, but if you want to add some, type in a comma separated list
Extra Feelings (comma separated, <enter> for none):cold,warm
Age (<enter> for all):
Gender, male or female (<enter> for both):
Which city (<enter> for all):
The request http://api.wefeelfine.org:8080/ShowFeelings?display=text&returnfields=sentence&
limit=1500 got 1501 responses
better : 90
bad : 43
good : 72
right : 70
guilty : 17
sick : 12
same : 21
shit : 6
sorry : 12
well : 24
down : 39
alone : 10
happy : 19
great : 25
comfortable : 14
sad : 20
free : 15
lost : 17
stupid : 11
tired : 12
weird : 8
lonely : 7
old : 39
home : 22
cold : 5
warm : 14
>>>
```

```
Python Shell
We'll use the top25 feelings, but if you want to add some, type in a comma separated list
Extra Feelings (comma separated, <enter> for none):cold
Age (<enter> for all):
Gender, male or female (<enter> for both):female
Which city (<enter> for all):chicago
The request http://api.wefeelfine.org:8080/ShowFeelings?display=text&returnfields=sentence&
limit=1500&city=chicago got 1501 responses
better : 70
bad : 61
good : 75
right : 61
guilty : 18
sick : 19
same : 20
shit : 22
sorry : 17
well : 31
down : 36
alone : 19
happy : 10
great : 20
comfortable : 11
sad : 8
free : 15
lost : 15
stupid : 14
tired : 17
weird : 10
lonely : 4
old : 49
home : 22
cold : 13
>>>
===== RESTART =====
>>>
```

