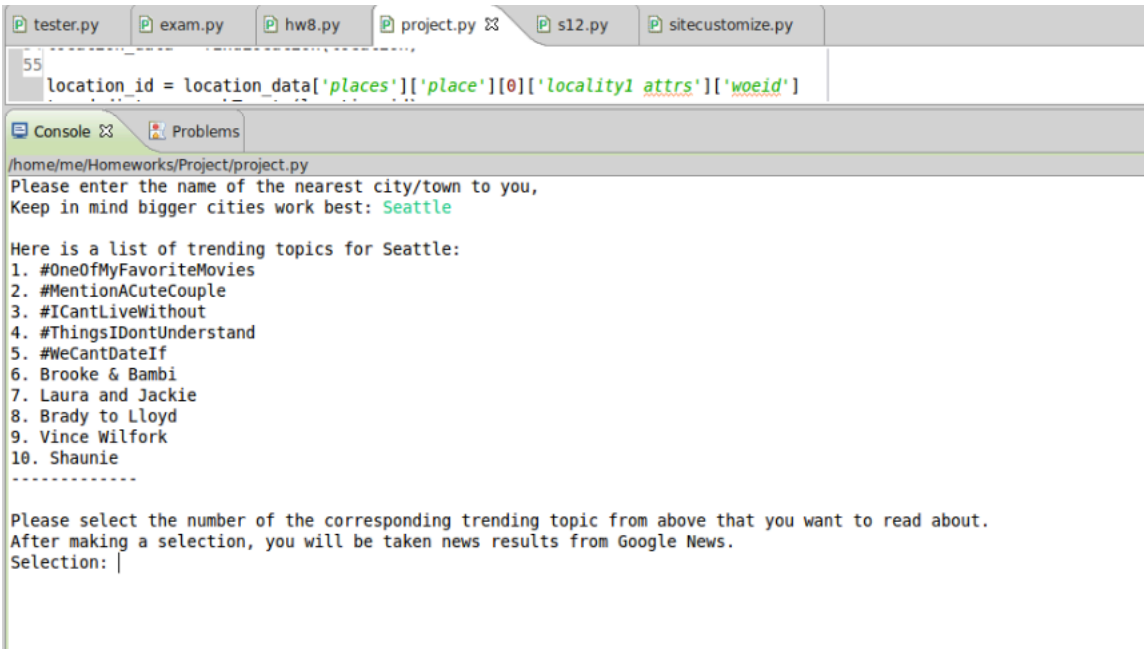
Martin Reyes
December 10, 2012
HCDE 310

Project Writeup

Here's a brief walkthrough of my project, how it works, what doesn't work, and other commentary on running the application:

The first thing you'll notice when running the application is a prompt to enter in a city name. This works best with major cities because Twitter only has topics for major cities – smaller cities and towns are usually shown trending topics from the closest major city. Once you enter in the name, it calls on the GeoPlanet API from Yahoo! to determine the WOEID of the location entered. Using this API was tricky since the documentation is poor and there aren't many examples online that I could find to use the API.

After that, a list of trending Twitter topics will be output to the console as shown in the screenshot below. The user is prompted to enter in the corresponding number of the topic they want to read more about. Something I originally overlooked was the use of hash tags in Twitter – most of these hash tagged topics are specific to Twitter only (meaning they aren't really related to news topics, which is the focus of this project), and they typically won't return any results on Google News, so it's recommended that the user search for one of the non-hash tagged topics.



Once the user has entered in the topic of their choice, the application will then open a browser window with search results from Google News on that topic. The most difficult part of getting those results to open was related to reading the JSON dictionary returned by the Google News API. The returned dictionary contains tons of data and tons of URLs, so I had to scavenge through the results to find the URL that I was looking for that takes the user to Google News instead of a specific news article site. The final output is shown below – I'm not really sure what some of the red text in the console is, though it doesn't affect the use of the application, so my guess is that it's related to the 'webbrowswer' library.