**Classes:**

The project contains three classes

1. TextFormat.java: This contains main methods and other methods to check whether URL is valid or not. If URL is valid it uses other classes to perform scrapping.
2. TextFormat.java: This contains methods to perform text cleaning like removing stop words and special characters. Once processing is done this uses Format\_helper.java class to extract common topics.
3. Format\_helper.java: It uses HashMap data structure to store keywords and their frequency. Then we will use sorting (Descending order) algorithm to sort HashMap based on value in <key, value> pair. Then we will return the top words of map (as these are most common topics in document from URL) and printing to the console.

**Stop Word References:**

1.https://algs4.cs.princeton.edu/35applications/stopwords.txt

2.https://github.com/stanfordnlp/CoreNLP/blob/master/data/edu/stanford/nlp/patterns/surface/stopwords.txt

**External Libraries:**

I have used java HTML parser (jsoup-1.11.3.jar) to perform web scrapping.

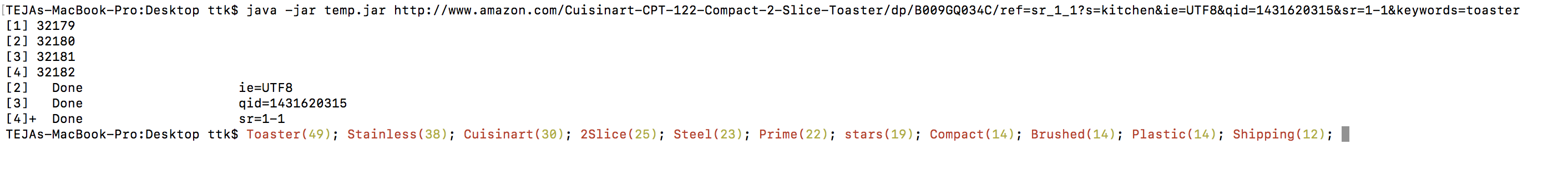
**Testing URL’s:**

[http://www.amazon.com/Cuisinart-CPT-122-Compact-2-Slice-Toaster/dp/B009GQ034C/ref=sr\_1\_1?s=kitchen&ie=UTF8&qid=1431620315&sr=1-1&keywords=toaster](https://t.lever-analytics.com/email-link?dest=http%3A%2F%2Fwww.amazon.com%2FCuisinart-CPT-122-Compact-2-Slice-Toaster%2Fdp%2FB009GQ034C%2Fref%3Dsr_1_1%3Fs%3Dkitchen%26ie%3DUTF8%26qid%3D1431620315%26sr%3D1-1%26keywords%3Dtoaster&eid=09e69ae5-7a73-438b-86b3-3307215e3917&idx=0&token=bDAcbMdv9p7JcBZpruZqa4i5ts0" \t "_blank)

[http://blog.rei.com/camp/how-to-introduce-your-indoorsy-friend-to-the-outdoors/](https://t.lever-analytics.com/email-link?dest=http%3A%2F%2Fblog.rei.com%2Fcamp%2Fhow-to-introduce-your-indoorsy-friend-to-the-outdoors%2F&eid=09e69ae5-7a73-438b-86b3-3307215e3917&idx=1&token=RU6GBBMwcvGWYeprhXJQPe4qDOQ)

[http://www.cnn.com/2013/06/10/politics/edward-snowden-profile/](https://t.lever-analytics.com/email-link?dest=http%3A%2F%2Fwww.cnn.com%2F2013%2F06%2F10%2Fpolitics%2Fedward-snowden-profile%2F&eid=09e69ae5-7a73-438b-86b3-3307215e3917&idx=2&token=GLMA2j3QQN4jdKvPbWAMHRiPz6I)

**Sample Output: (For Amazon Test case)**

****