Part 1

Abstract:

The objective of solving part 1 was to scrape the 10K filings and clean the data to make it usable. As our initial steps we constructed the URL. We then attempted to scrape the 10Q file to retrieve the data. Once, the data was retrieved which was in an inconsistent, noisy and incomplete format we moved to cleanse the data. Our approach encompasses retrieving the right data, making it consistent and eliminating the noise as much as possible, hence allowing data insights to be drawn.

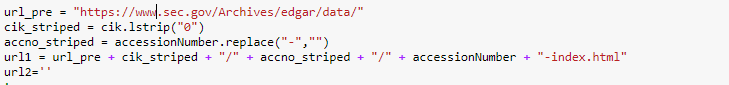
Steps

1. Web Scraping: We used python to first navigate to <http://www.sec.gov/Archives/edgar/data/51143/000005114313000007/0000051143-13-000007-index.html>.We programmatically generated the URL and navigated to the 10Q website.
2. Data Cleansing:
3. This process encompassed making the data readable. We extrapolated all the tables from the form.
4. We replaced the ‘$’ symbols. Next, we reset all indexes and re-indexed the tables.
5. While doing so, we realized that python reindexes by filling empty columns to replace missing rows. Hence, we had re-label the columns and then run the re-index.
6. Following which, we replaced the empty strings using regex and eliminated the uneven number of parenthesis and percentages. We repeated the removal of NAN’s and None and reindexed the tables
7. Following which, we programmatically used a logic that would realign column as per the respective headings.

Data Ingestion and Wrangling

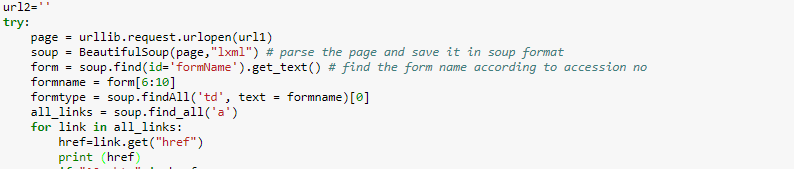
Step 1

Constructing the URL

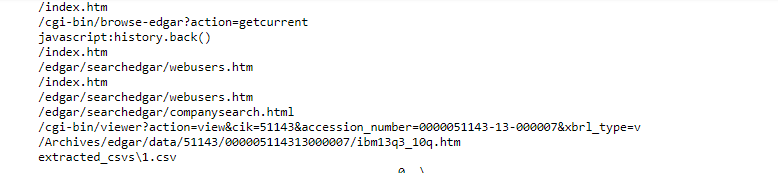


Step2

Selecting all href links

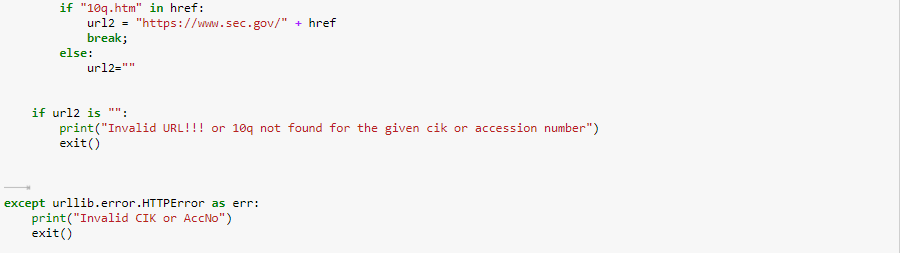


List of href links generated



Step 3

Explicitly selecting the 10Q links and Exception Handling



10 Q URL generated

