# Assignment 1 – Part 1

The main aim of the assignment was to learn the concepts of handling missing data, summarizing metrics and learning about pipelining Docker applications and using buckets on Amazons S3 cloud to store data.

Part 1 – The first section of the assignment included getting data from the Edgar File set and programmatically generating a URL to import data onto the local filesystem. These files are later saved as CSV files and uploaded to the S3 buckets created. We followed the following steps to get the part done:

1. Configuring buckets on Amazon S3

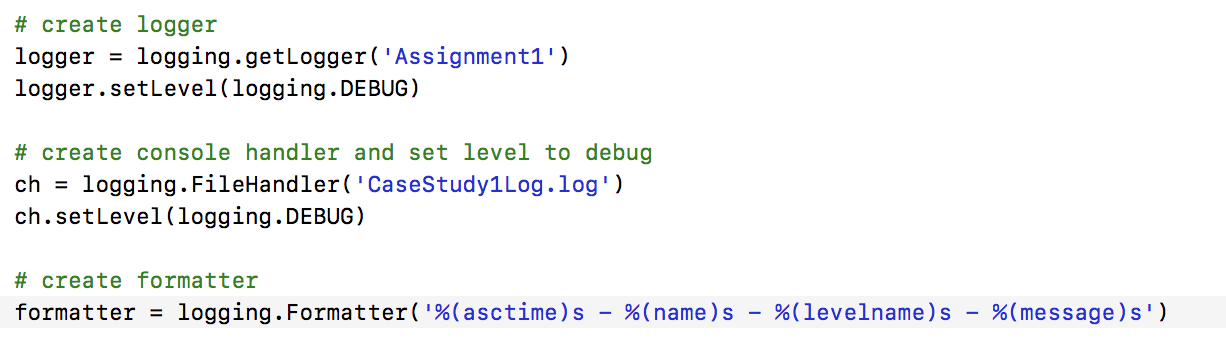
* Created users with read/ write permissions and programmatic access
* Created a bucket with appropriate permissions

1. Used a Python script to take inputs from the command line

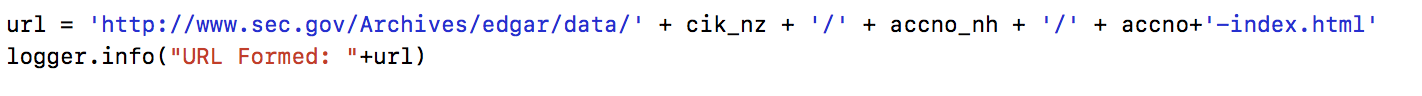
* Set loggers to verify statuses of http requests and log errors in the case of exceptions
* Used a command line input to take in fresh CIK, ACCNO, Filetype, AWS keys and bucket names. For this particular instance, the bucket name remains the same to avoid collisions.
* Split the URL into segments to get to the intended CIK file and beautified using BeautifulSoup
* Used boto to connect to S3 and upload the zip and log files.

1. Used GIT to store and Docker hub to pipeline the application for cross platform library independent use.

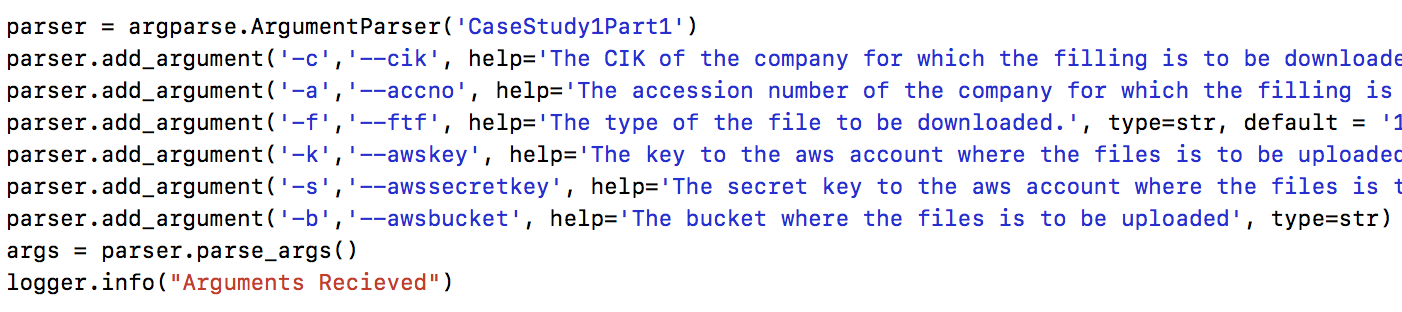
Some screenshots of the code are to follow along with the code snippets.



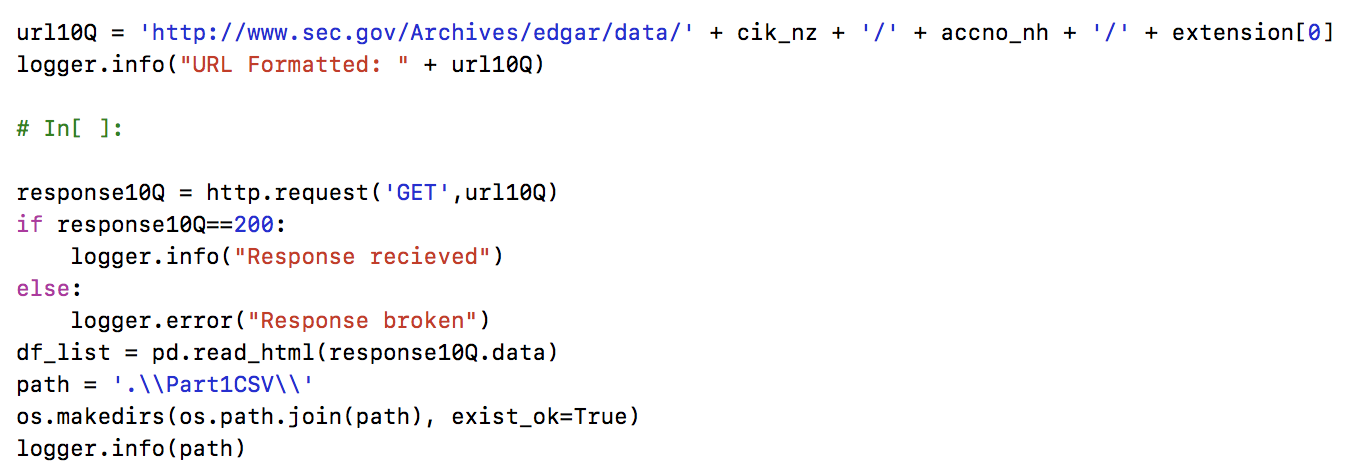
Logger Snippet



URL formation for file download as CSV



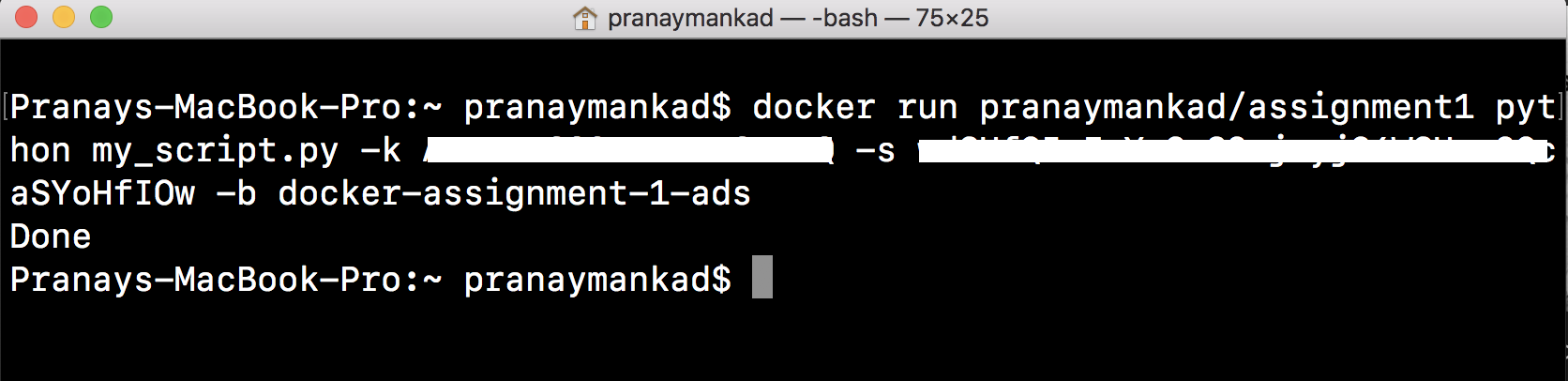
Accepting Command Line Arguments for a more dynamic environment



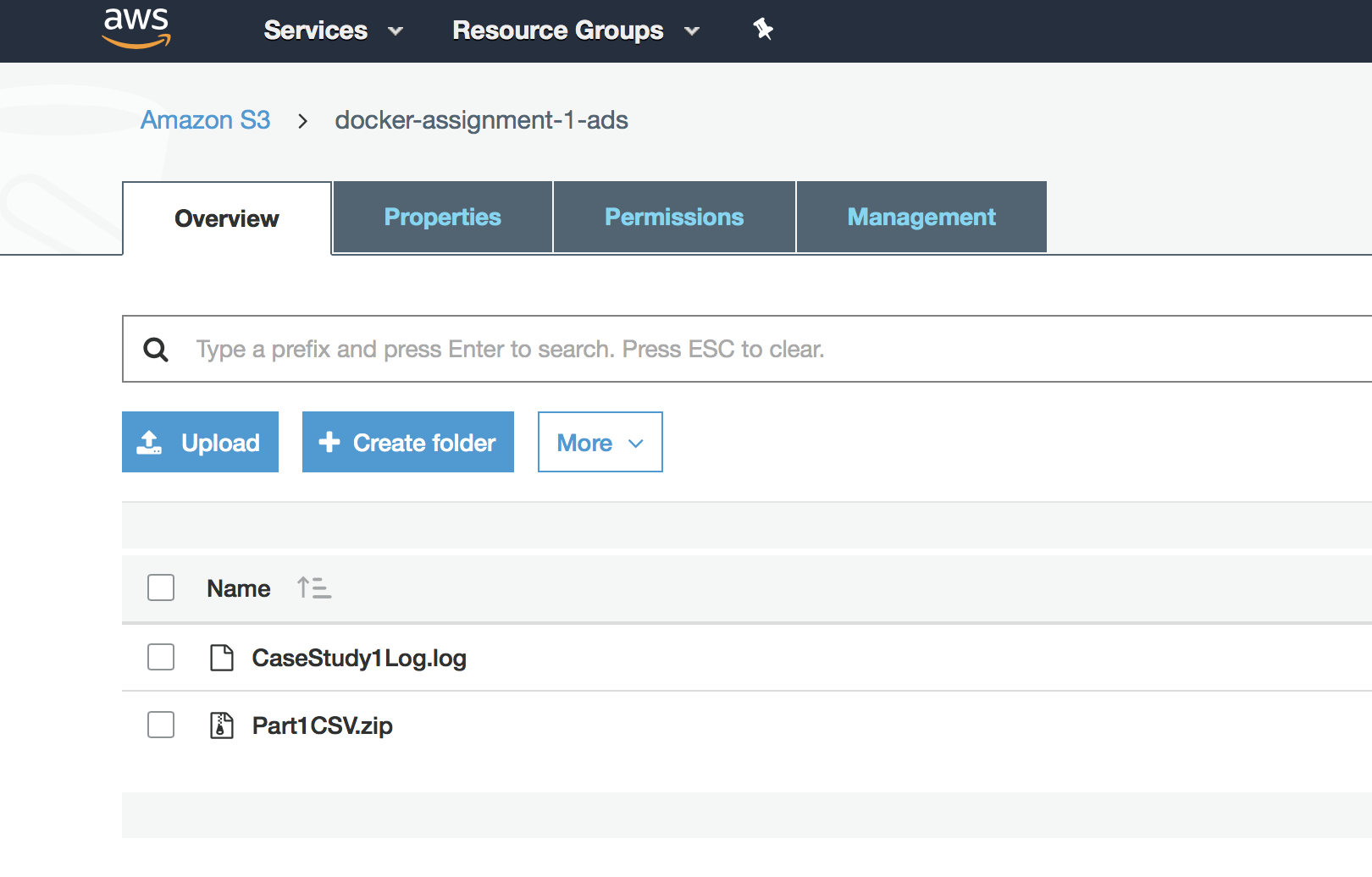
Picking up URLs from EDGAR and handling request status for logging



File Upload using boto to Amazons S3 cloud



Running Part 1 gives Done when completed.



Uploaded S3 Files on S3 Console