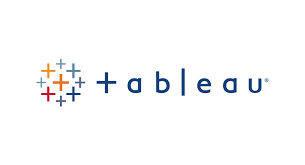
[Enter Post Title Here]

Data Wrangling, Data

Analysis & Visualization

-*Under Guidance of Prof. Sri Krishnamurthy*

**

*Compiled By:*

Mohit Mittal

Sneha Ravikumar

Taj Poovaiah

Table of Contents

[OBJECTIVE 3](#_Toc476001040)

[INTRODUCTION TO DOCKER 3](#_Toc476001041)

[DOCKER DESIGN AND IMPLEMENTATION 3](#_Toc476001042)

[PULLING A DOCKER IMAGE AND RUNNING IT ON YOUR MACHINE 4](#_Toc476001043)

[TABLEAU DESIGN AND IMPLEMENTATION 8](#_Toc476001044)

[WHY TABLEAU IS USED IN THIS PROJECT 8](#_Toc476001045)

[GETTING STARTED WITH TABLEAU 8](#_Toc476001046)

[ANALYSIS AND REVIEW OF TABLEAU VISUALIZATION 9](#_Toc476001047)

[DASHBOARD 12](#_Toc476001048)

[AWS S3 DATA ANALYSIS 13](#_Toc476001049)

[REVIEW OF OUTPUTS STORED ON AMAZON S3 14](#_Toc476001050)

[PART1 RESULT OUTPUT 14](#_Toc476001051)

[PART2 RESULT OUTPUT 14](#_Toc476001052)

[EXCEPTION HANDLING- Part1 14](#_Toc476001053)

[EXCEPTION HANDLING- Part2 14](#_Toc476001054)

[INVALID AMAZON KEY EXCEPTION HANDLING 15](#_Toc476001055)

[IF DATA IS NOT FOUND 15](#_Toc476001056)

# OBJECTIVE

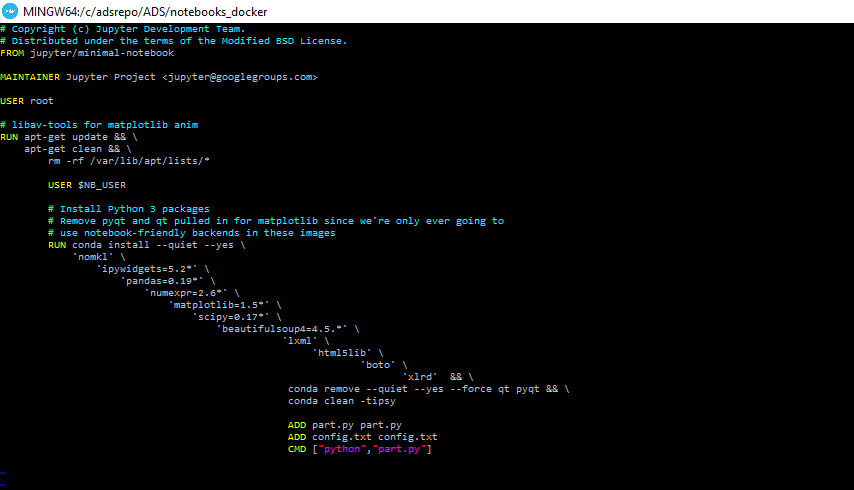
The objective of our assignment is to gather, retrieve, clean, and analyze the data using python code, dockerize the pipeline so as to upload our end results onto Amazon S3 buckets. The use of configuration file and Docker image makes it easy for flexible usage.

# INTRODUCTION TO DOCKER

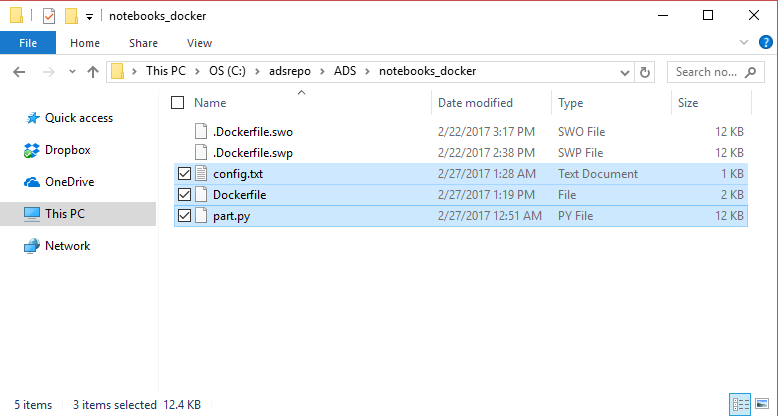
Docker is an open-source project that automates the deployment of applications inside software containers. Quote of features from Docker web pages: “Docker containers wrap up a piece of software in a complete filesystem that contains everything it needs to run: code, runtime, system tools, and system libraries – anything you can install on a server. This guarantees that it will always run the same, regardless of the environment it is running in.”

# DOCKER DESIGN AND IMPLEMENTATION

Docker file is used to specify the configurations and settings required for the application to run:



The Docker Directory contains the files required to run the application

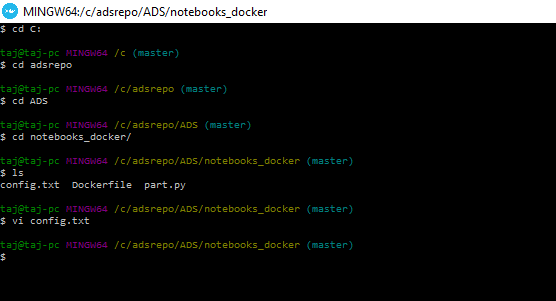


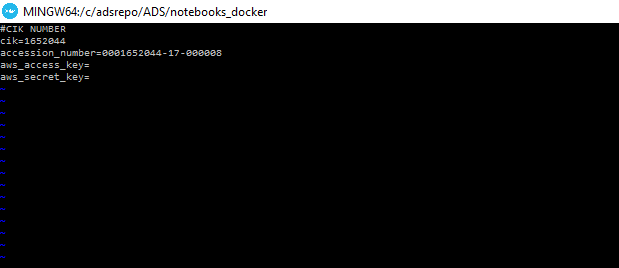
## PULLING A DOCKER IMAGE AND RUNNING IT ON YOUR MACHINE

Open the Docker Quickstart terminal on your machine to get started by running the commands on Docker

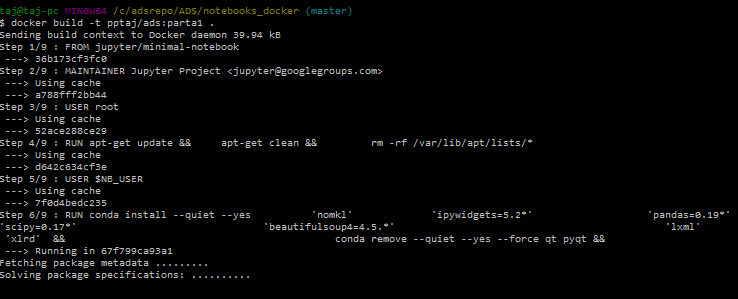
1. Run **$docker pull “repo name”** to pull the image from any repo on­­to your system

2. Run **$ls** and then **$vi config.txt** to Update the configuration file with the values that you want to test (CIK, accession number, AWS access key, AWS secret key)

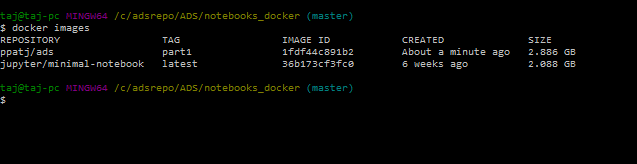




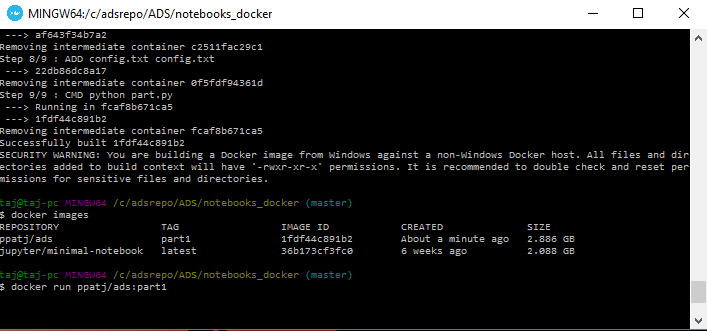
1. Run **$docker build –t “repo name:tag name”** . to build the image that you pulled



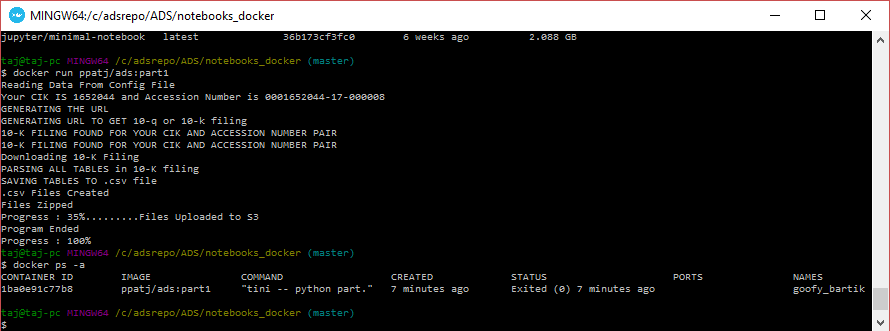
1. Run **$docker images** to check the list of docker images in the system



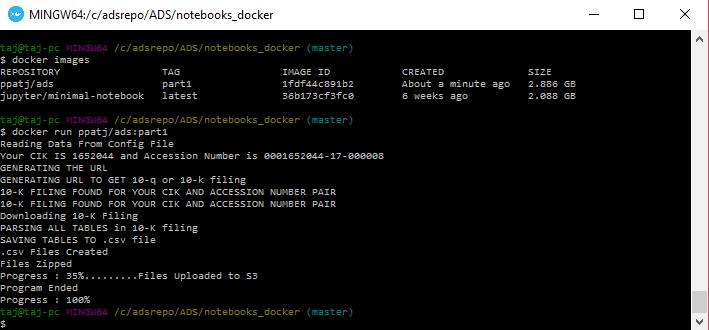
1. Run **$docker run “repo name:tag name”** to run the application on Docker



1. Run **$docker ps -a** to check the list of docker images in the system



The status of the program being executed will be updated on the terminal as shown below



# TABLEAU DESIGN AND IMPLEMENTATION

## WHY TABLEAU IS USED IN THIS PROJECT

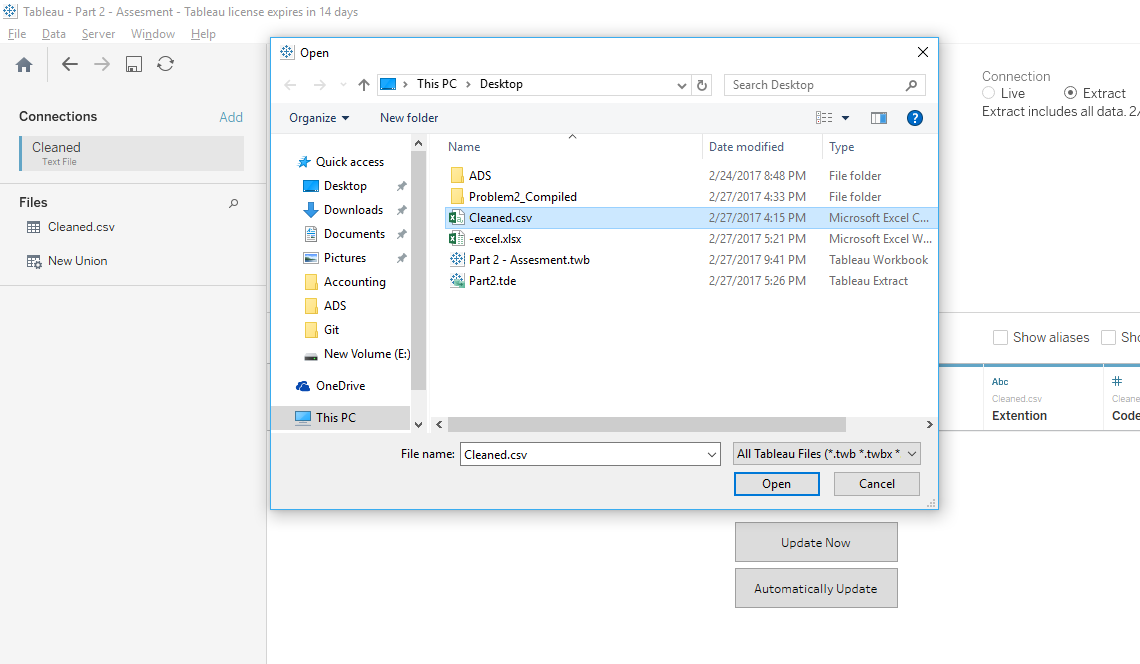
Once the data sets are gathered, cleaned, summarized, and analyzed, tableau is used to show the visual representation of the summary metrics to make it easier to understand and interpret

*Tableau can help anyone see and understand their data. Connect to almost any database, drag and drop to create visualizations, and share with a click.*

## GETTING STARTED WITH TABLEAU

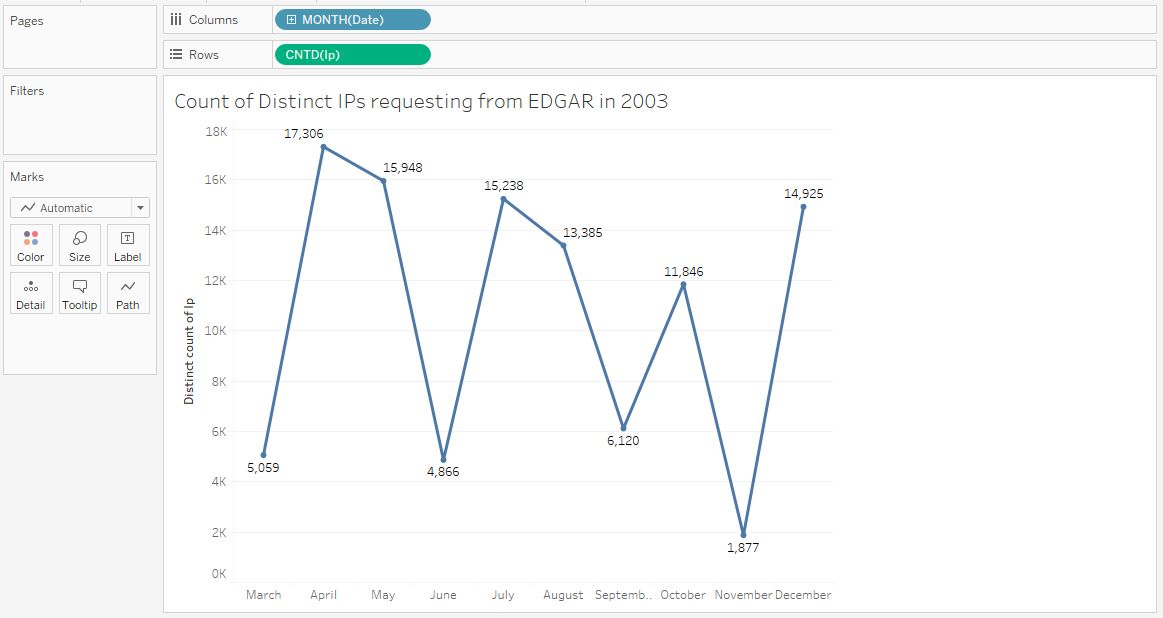
Import the file that contains the data you want to analyze on Tableau

Data Source -> File -> Open -> Cleaned.csv



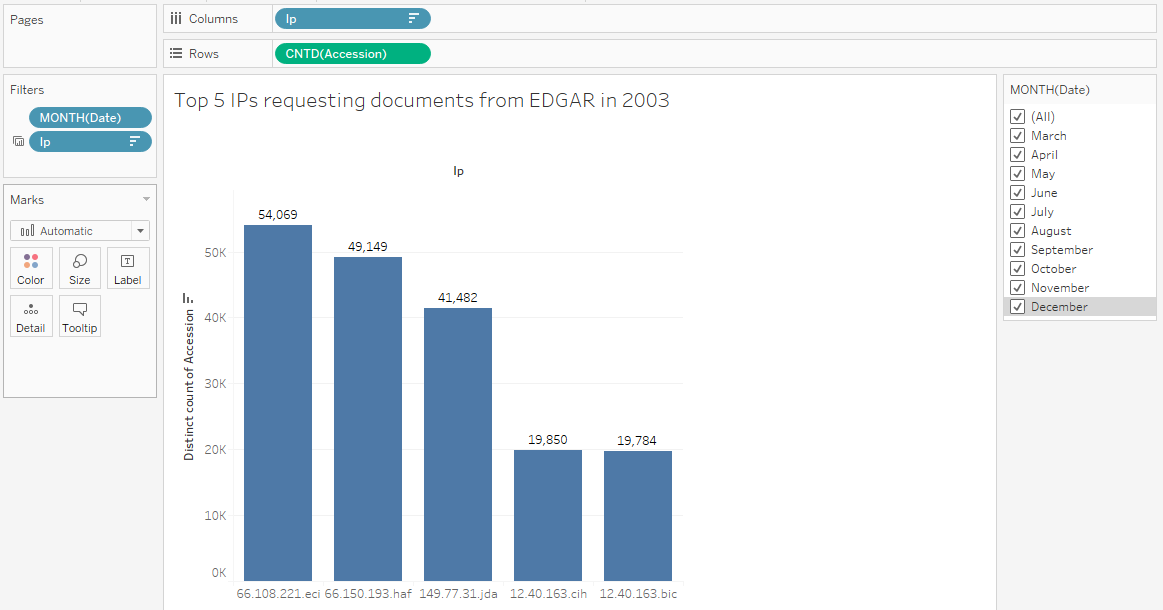
## analysis and review of tableau vizualizations

1. How many times distinct machines accessed data on EDGAR?



Its observed that over the nine months’ data we have, April observes the most traffic with over seventeen thousand visitors and November with a low of only over thousand.

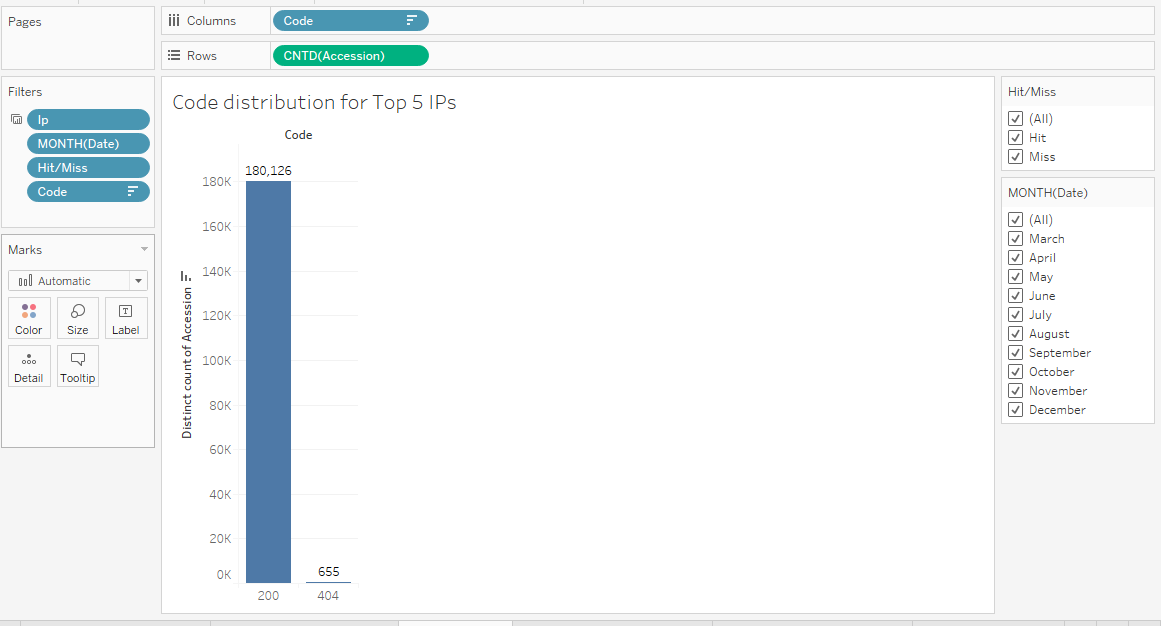
1. What are the popular IPs accessing data?



Of the thousands of IPs hitting the website, we understand that these top 5 IPs keep coming back across time and access files most frequently.

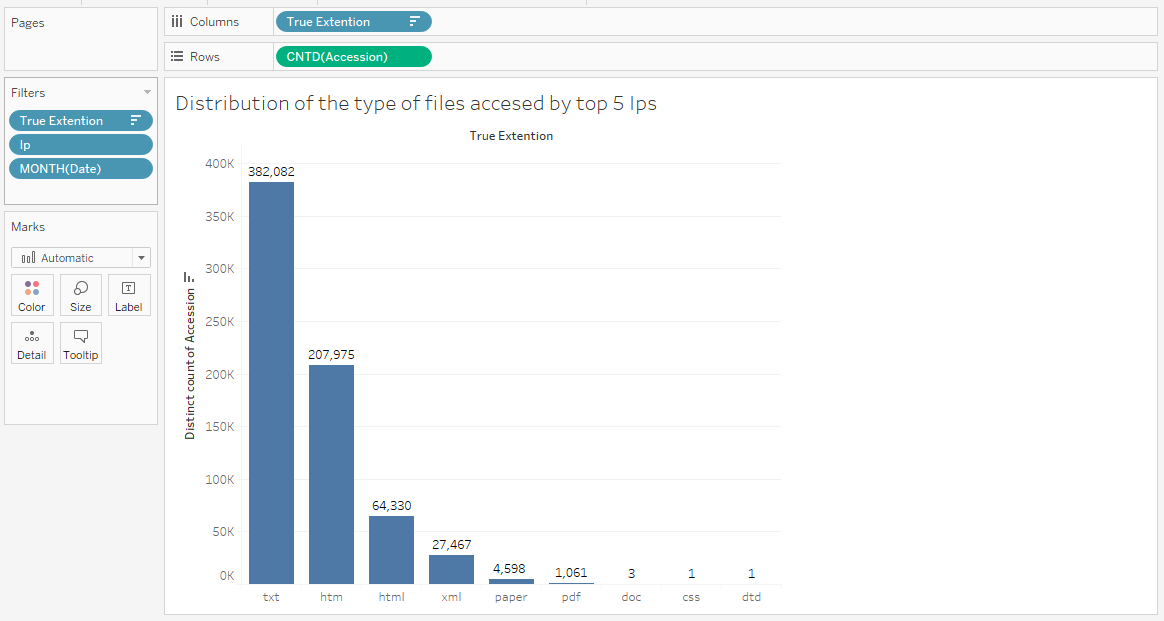
The IP 66.108.221.eci accesses 54,069 accession files over the 12 months in 2003.

1. For the requests made by the IPs, what is the hit to miss ratio?



We observe that most of the requests pass with an ‘OK’ (200) code.

1. What type of files are being accessed?



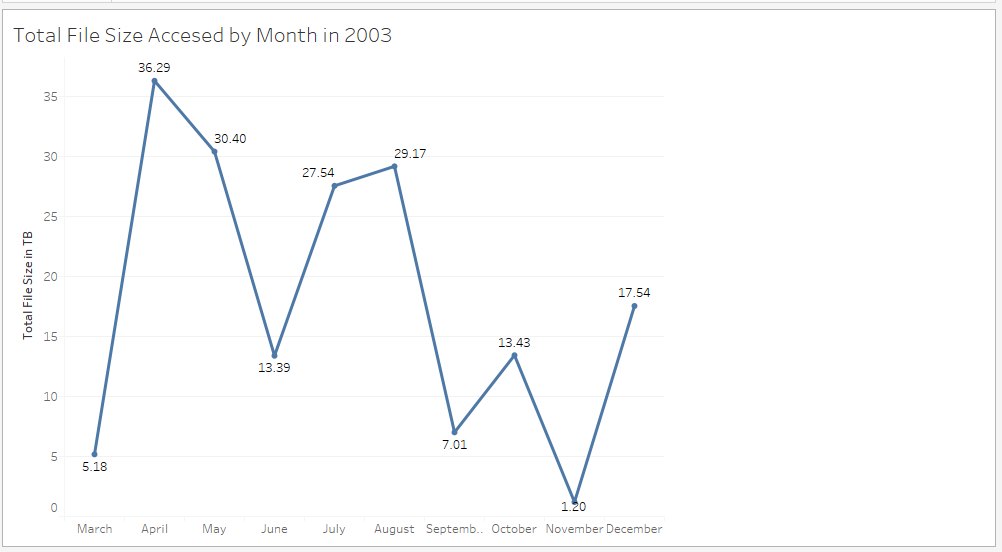
We observe from the above distribution that most of the files accessed are .txt files.

1. What are the most frequently accessed accession numbers?



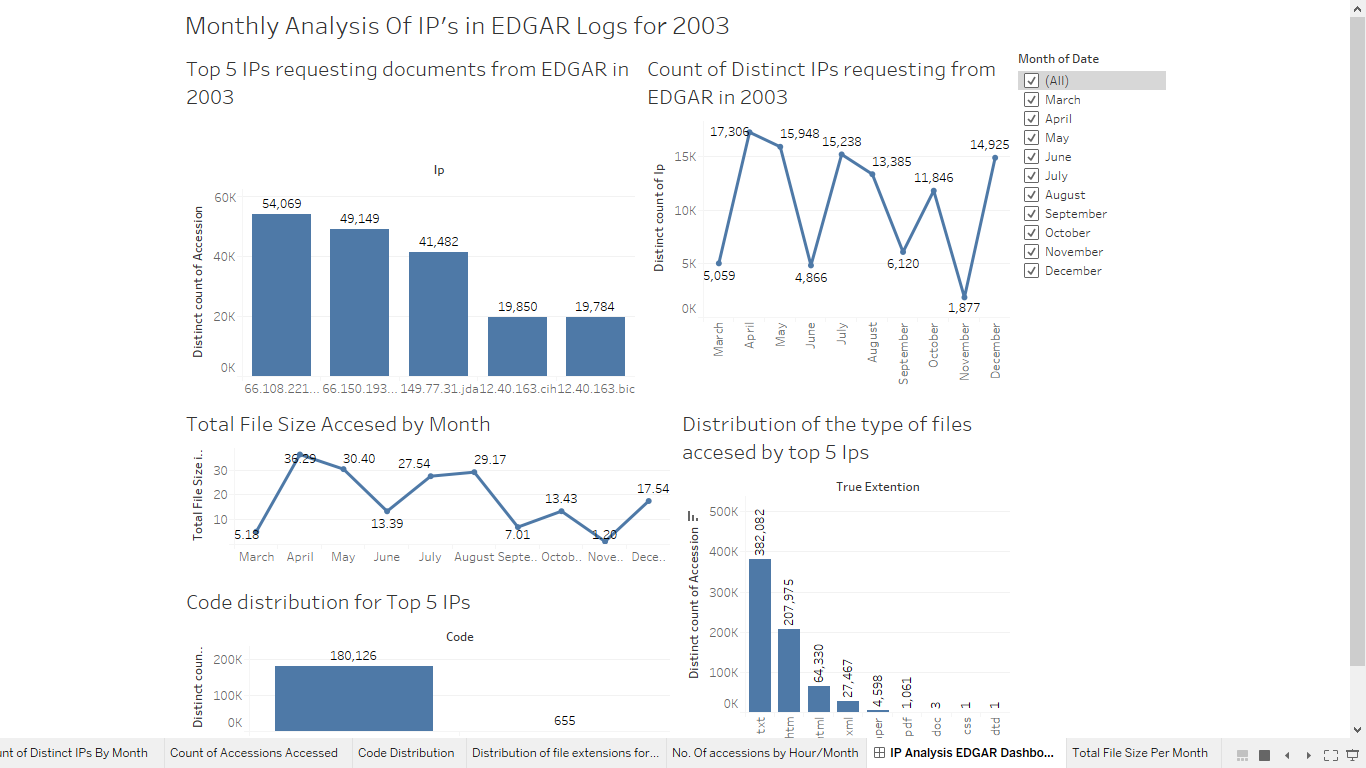
We plot the accession numbers against how many times they occur over 12 months to understand the most frequently requested accession number. It can be filtered for each month.

1. We understand there are multiple kinds of files accessed. What is the total file size of data downloaded over months?



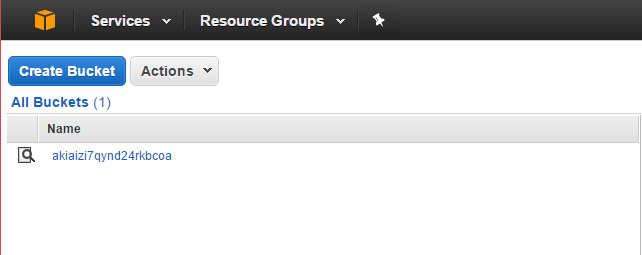
As observed, since April has the highest traffic, It also has the highest file size downloaded with a high of 36.29 TB

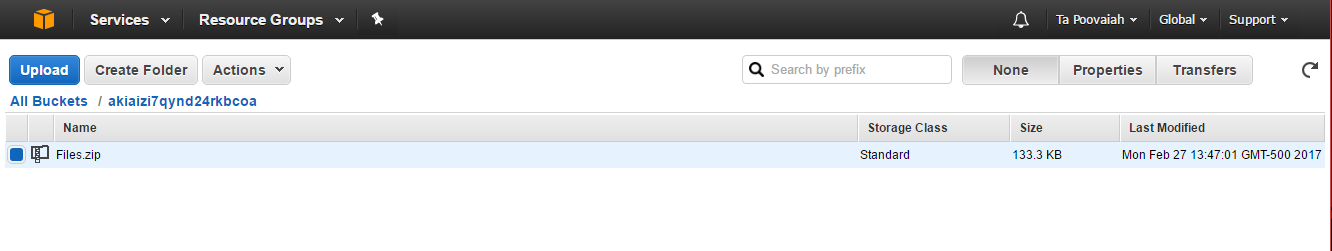
### dashboard



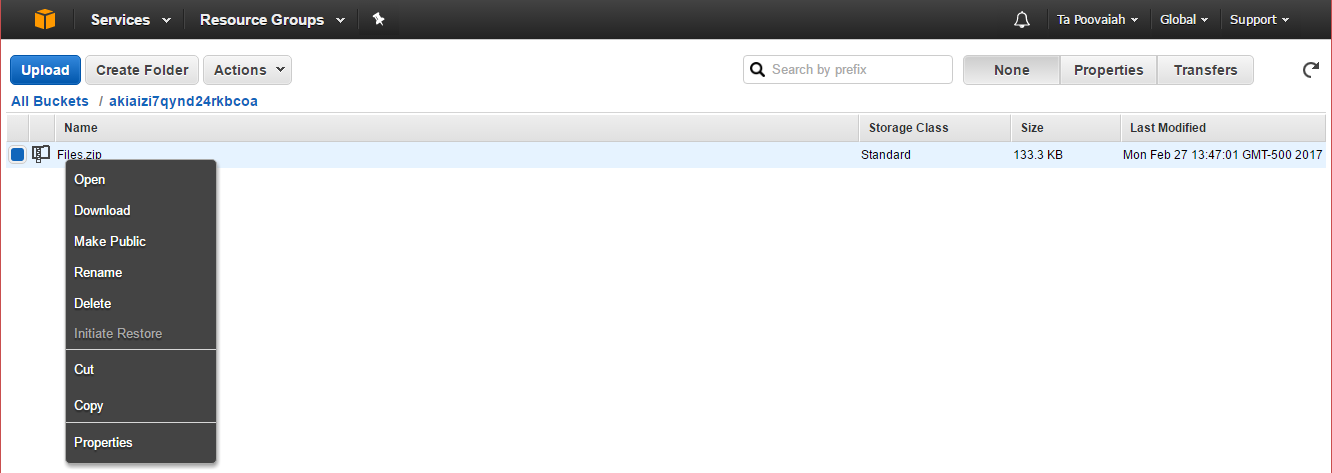
# AWS S3 DATA ANALYSIS

1. Log in to your AWS account, select S3, and check if a bucket with you aws access key has been created. Check if the Zip file has been uploaded onto the bucket





1. Download the zip folder and check the contents



# REVIEW OF OUTPUTS STORED ON AMAZON S3

## PART1 RESULT OUTPUT

The results in the Amazon S3 bucket consists of a zip file that has all of the csvs (the csvs have the special characters removed while scarping data) of either 10K/10Q documents of a particular company recognized based on the CIK and accession number provided by the user in the configuration file.

## PART2 RESULT OUTPUT

The results in the Amazon S3 bucket consists of a zip file. The zip file consists of the final result which is obtained by:

* The files for a particular year is downloaded into a folder
* The files are then cleansed, and put into another folder
* Summary metrics functions are performed on every cleaned csv for every month
* The cleaned files are compiled into a single csv
* The compilation of the summary for all the 12 months are then added onto one summary file and then into another folder
* All of these folders, are then zipped, and uploaded

# cleaning the data

On individual cleaned CSVs, we performed the following steps to get cleaned CSV files:

* Added columns by splitting Date and Time fields
* Removed all the rows with Error Code = 0 ( Non –existent) - outlier removal
* Obtained a unique column showing the extension for each file
* Removed junk extensions (.hd, .hmtl etc.)
* Replaced all the un-available data with ‘NA’

# EXCEPTION HANDLING- Part1

* Program handles the error if no CIK or Accession Number is provided
* Program handles the error if the CIK and Accession number is valid, if not, throws error
* Program handles the error for 10k/10q documents. If absent, throws error
* Program validates links provided, handles the error if link is invalid

# EXCEPTION HANDLING- Part2

* Program handles the error if year is empty
* Program handles the error if year is invalid (eg:3, 44)
* Program handles the error if the year provided does not exist

(eg: not 2003 <= year =< 2016 )

# INVALID AMAZON KEY EXCEPTION HANDLING

* Program Checks if Amazon keys (access and secret) provided by user is empty, throws error if it is
* Program checks if the keys are invalid. If the keys are not valid, the connection to S3 fails and throws an error as soon as it reads the keys from the configuration file
* Program checks if bucket exists. If it does, it adds the files into the same bucket
* Program replaces the files that are uploaded with the most recent upload if the file name is the same in the bucket

# IF DATA IS NOT FOUND

* If Data (10Q/10K documents) are not found, program handles it by prompting that the link provided doesn’t contain 10K/10Q documents

**END OF REPORT**