



CSCI5408 – ASSIGNMENT 3

Dhrumil Amish Shah (B00857606)
dh416386@dal.ca

Problem 1 – Task 1

Task Description

Cluster Setup – Apache Spark Framework on Google Cloud Platform (GCP) [1].

Steps for setting up the pre-requisites for the Apache Spark installation and configuration on Google Cloud Platform (GCP) [1]

A virtual machine instance will be created using Google Cloud Platform's Compute Engine for creating and setting up Apache Spark clusters.

Step 1 – Create a project named **Data5408-MongoNews** on GCP and link it to a billing account.

Step 2 – Enable the Compute Engine API for the project **Data5408-MongoNews**.

(i.e., compute.googleapis.com)

Step 3 – Create an instance **mongonewsinstance** in the project **Data5408-MongoNews** with the below configurations:

- Name – mongonewsinstance
- Region – us-central1 (Iowa)
- Zone – us-central1-a
- Machine configuration
 - Machine family – General-purpose
 - Series – E2
 - Machine type – e2-medium (2vCPU, 4GB memory)
- Boot Disk
 - Boot disk image – Public images
 - Operating system – Ubuntu
 - Version – Ubuntu 20.04 LTS
 - Boot disk type – Balanced persistent disk
 - Size(GB) – 10
- Identify and API access
 - Service account – Compute Engine default service account
 - Access scopes – Allow default access
- Firewall
 - Check option Allow HTTP traffic
 - Check option Allow HTTPS traffic

Step 4 – Click **Create** to create instance **mongonewsinstance**.

Step 5 – Click **mongonewsinstance** instance name.

Step 6 – Click **nic0** network interface name to view the network interface details.

Step 7 – On the left side panel, below the VPC network, click the **Firewall** option.

Step 8 – Edit settings for **default-allow-http** and **default-allow-https** options in the list.

- Click on the option name. (**default-allow-http** or **default-allow-https**)
- Click the edit option next to the Firewall rule details.
- Change Protocols and ports from **Specified protocols and ports** to **Allow all**.
- Do this for both the rules. (**default-allow-http** and **default-allow-https**)

⇒ **Figure 1** displays the created virtual machine (VM) instance **mongonewsinstance** under the project **Data5408-MongoNews** on GCP.

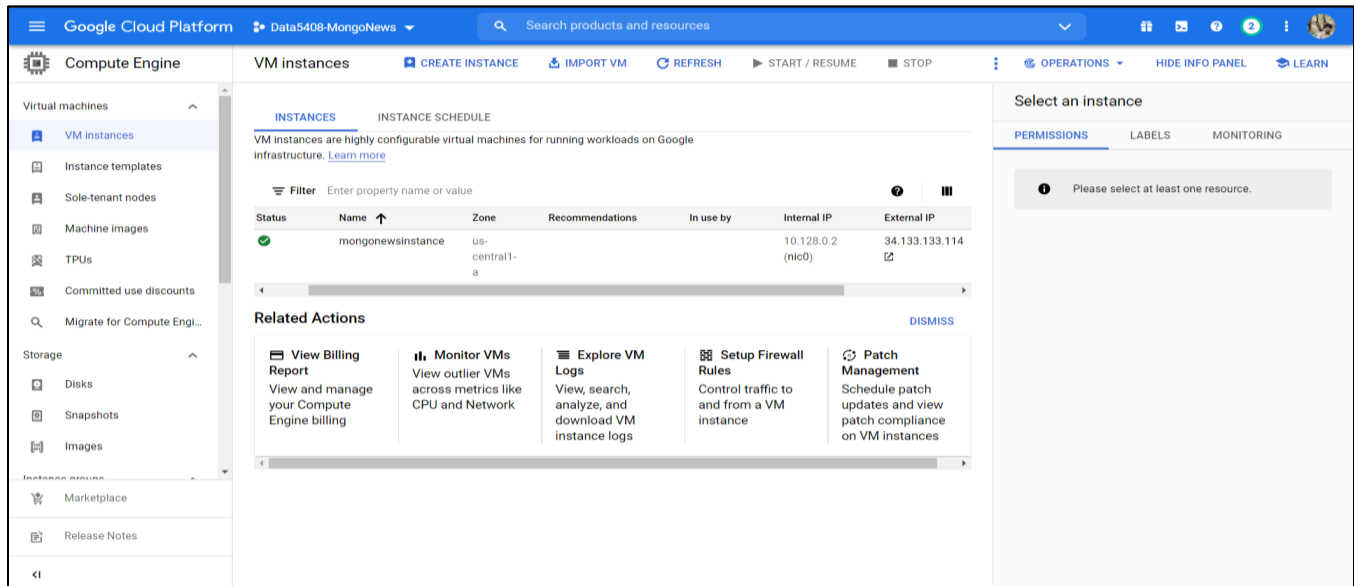


Figure 1 – mongonewsinstance virtual machine hosted on GCP [1]

Step 9 – Connect to the **mongonewsinstance**'s shell by clicking **SSH** under Connect option.

Step 10 – As the instance is new, update the OS using the command **sudo apt update**.

Step 11 – Install Java and Scala. (Apache Spark requirements). To verify the installation, run commands **java -version** and **scala -version**. Below are the installation commands [2]:

- For java – **sudo apt-get install default-jdk -y**
- For scala – **sudo apt-get install scala -y**

⇒ **Figure 2** displays the successful installation of Java and Spark on the VM.



Figure 2 - Successful installation of Java and Spark [2]

Step 12 – Download and Setup Spark [2]

- Visit <https://spark.apache.org/downloads.html> website and select the latest Spark release and package type. For the current installation, I chose the latest version of Spark [3].
 - Spark release – 3.1.2 (Jun 01 2021)
 - Package type – Pre-built for Apache Hadoop 3.2 and later
- Based on the versions of Spark release and Package type, the direct URL to download the Spark archive is <https://downloads.apache.org/spark/spark-3.1.2/spark-3.1.2-bin-hadoop3.2.tgz>.
- Use the **wget** command and the direct URL to download the Spark archive.
 - **wget https://downloads.apache.org/spark/spark-3.1.2/spark-3.1.2-bin-hadoop3.2.tgz**

- Extract the saved archive using the **tar** command.
 - **tar xvf spark-3.1.2-bin-hadoop3.2.tgz**
- Move the extracted tar(unpacked directory) **spark-3.1.2-bin-hadoop3.2** to the **opt/spark** directory.
 - **sudo mv spark-3.1.2-bin-hadoop3.2 /opt/spark**

⇒ **Figure 3** displays the download and extraction of the Spark archive.

```

shah_dhruvil1998@mongonewsinstance:~$ wget https://downloads.apache.org/spark/spark-3.1.2/spark-3.1.2-bin-hadoop3.2.tgz
--2021-06-28 23:44:34-- https://downloads.apache.org/spark/spark-3.1.2/spark-3.1.2-bin-hadoop3.2.tgz
Resolving downloads.apache.org (downloads.apache.org)... 135.181.209.10, 88.99.95.219, 135.181.214.104, ...
Connecting to downloads.apache.org (downloads.apache.org)|135.181.209.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 228834641 (218M) [application/x-gzip]
Saving to: 'spark-3.1.2-bin-hadoop3.2.tgz'

spark-3.1.2-bin-hadoop3.2.tgz      100%[=====] 218.23M  23.3MB/s   in 10s
2021-06-28 23:44:45 (21.4 MB/s) - 'spark-3.1.2-bin-hadoop3.2.tgz' saved [228834641/228834641]

shah_dhruvil1998@mongonewsinstance:~$ tar xvf spark-3.1.2-bin-hadoop3.2.tgz
spark-3.1.2-bin-hadoop3.2/
spark-3.1.2-bin-hadoop3.2/R/
spark-3.1.2-bin-hadoop3.2/R/lib/
spark-3.1.2-bin-hadoop3.2/R/lib/sparkr.zip
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/worker/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/worker/worker.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/worker/daemon.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/tests/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/tests/testthat/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/tests/testthat/test_basic.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/profile/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/profile/shell.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/profile/general.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/doc/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/doc/sparkr-vignettes.html
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/doc/sparkr-vignettes.Rmd
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/doc/sparkr-vignettes.R
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/doc/index.html
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/R/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/R/SparkR
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/R/SparkR.rdx
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/R/SparkR.rdb
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/features.rds
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/package.rds
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/naInfo.rds
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/vignette.rds
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/Rd.rds
spark-3.1.2-bin-hadoop3.2/R/lib/SparkR/Meta/links.rds

```

Figure 3 – Download and extraction of Spark archive [2]

⇒ **Figure 4** displays the successful migration of the unpacked Spark directory to the **/opt/spark** directory.

```

shah_dhruvil1998@mongonewsinstance:~$ sudo mv spark-3.1.2-bin-hadoop3.2 /opt/spark
shah_dhruvil1998@mongonewsinstance:~$

```

Figure 4 – Migration of unpacked Spark directory [2]

Step 13 – Configure Spark environment [2]

- Configure environment variables for the Spark environment. Below are the commands to be added in the user profile.
 - **echo "export SPARK_HOME=/opt/spark" >> ~/.profile**
 - **echo "export PATH=\$PATH:\$SPARK_HOME/bin:\$SPARK_HOME/sbin" >> ~/.profile**
 - **echo "export PYSARK_PYTHON=/usr/bin/python3" >> ~/.profile**
- Update the .profile file in the command line using the command
 - **source ~/.profile**

Step 14 – Start Master server [2]

- Start master server using the following command:
 - **sudo /opt/spark/sbin/start-master.sh**

- To view the Spark Master web user interface, open a web browser and enter the following in the address bar:
 - http://[External IP]:8080/** (For instance: http://34.133.133.114:8080/)

⇒ **Figure 5** displays the command executed to start the Master server.

```
shah_dhru11998@mongonewsinstance:~$ sudo /opt/spark/sbin/start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /opt/spark/logs/spark-root-org.apache.spark.deploy.master.Master-1-mongonewsinstance.out
shah_dhru11998@mongonewsinstance:~$
```

Figure 5 – Master server start command execution [2]

⇒ **Figure 6** displays the Spark Master web user interface.

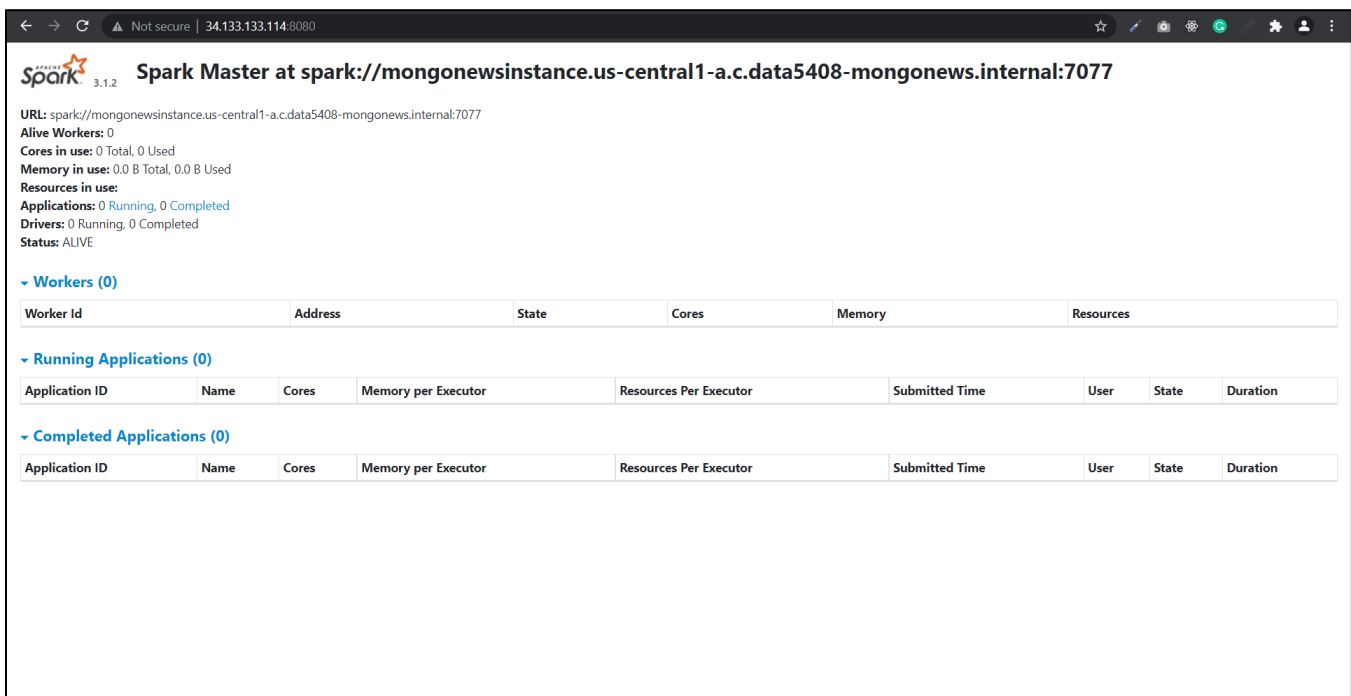


Figure 6 – Spark Master web user interface [2]

Step 15 – Start Worker server [2]

- Start Worker server along with the master server using the below command:
 - sudo /opt/spark/sbin/start-worker.sh spark://[master]:[port]** (For instance: `sudo /opt/spark/sbin/start-worker.sh spark://mongonewsinstance.us-central1-a.c.data5408-mongonews.internal:7077`)
- Refresh the Spark Master web user interface from the browser. The web user interface now shows the Worker server connected to the Master server.

⇒ **Figure 7** displays the command executed to start the Worker server.

```
shah_dhruvil1998@mongonewsinstance:~$ sudo /opt/spark/sbin/start-worker.sh spark://mongonewsinstance.us-central1-a.c.data5408-mongonews.internal:7077
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-root-org.apache.spark.deploy.worker.Worker-1-mongonewsinstance.out
shah_dhruvil1998@mongonewsinstance:~$
```

Figure 7 - Worker server start command execution [2]

⇒ **Figure 8** displays the Master server web user interface connected to the Worker server.

The screenshot shows the Spark Master web interface at the URL `spark://mongonewsinstance.us-central1-a.c.data5408-mongonews.internal:7077`. The interface displays the following information:

- URL:** `spark://mongonewsinstance.us-central1-a.c.data5408-mongonews.internal:7077`
- Alive Workers:** 1
- Cores in use:** 2 Total, 0 Used
- Memory in use:** 2.8 GiB Total, 0.0 B Used
- Resources in use:**
- Applications:** 0 Running, 0 Completed
- Drivers:** 0 Running, 0 Completed
- Status:** ALIVE

Below this summary, there are three expandable sections:

- Workers (1):** A table showing one worker with ID `worker-20210629151640-10.128.0.2-35595`, Address `10.128.0.2:35595`, State `ALIVE`, Cores `2 (0 Used)`, and Memory `2.8 GiB (0.0 B Used)`.
- Running Applications (0):** An empty table with columns: Application ID, Name, Cores, Memory per Executor, Resources Per Executor, Submitted Time, User, State, and Duration.
- Completed Applications (0):** An empty table with the same columns as the Running Applications section.

Figure 8 - Spark Master web user interface connected to the Worker server [2]

Step 16 – Test Spark Shell [2]

- Test the working of Spark Shell by starting the Spark Shell using the below command:
 - **`sudo /opt/spark/bin/spark-shell`**

⇒ **Figure 9** displays the Spark Shell started.

```
shah_dhruvil1998@mongonewsinstance:~$ sudo /opt/spark/bin/spark-shell
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/opt/spark/jars/spark-unsafe_2.12-3.1.2.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
21/06/29 15:18:58 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Spark context Web UI available at http://mongonewsinstance.us-central1-a.c.data5408-mongonews.internal:4040
Spark context available as 'sc' (master = local[*], app id = local-1624979947732).
Spark session available as 'spark'.
Welcome to

  ____  __
 / ___/  / /
/ /   /  / /
/ /___/  / /
\___/___/ /_

version 3.1.2

Using Scala version 2.12.10 (OpenJDK 64-Bit Server VM, Java 11.0.11)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

Figure 9 - Spark Shell started [2]

Problem 1 – Task 2

Task Description

Data Extraction and Preprocessing Engine: Sources - NewsAPI [4]

Final Code (GitLab link) [5]

⇒ <https://git.cs.dal.ca/dashah/csci-5408-s2021-b00857606-dhruvil-amish-shah/-/tree/master/A3>

Extraction Engine

The extraction engine is responsible for extracting the news data (news articles) from the <https://newsapi.org/> [4] website and store the extracted raw news data in files programmatically. Also, each file can only contain at most five articles. Search keywords for which articles are to be collected are Canada, University, Dalhousie, Halifax, Canada Education, Moncton, and Toronto.

For each search keyword, I extracted a hundred articles. So, for seven keywords, seven hundred news articles in total are saved. Since each file can only contain five articles maximum, hundred and forty files are created. The news articles are stored in JSON files with the file extension ".json" in the folder named "newsData". The naming convention of files is

<search_keyword_lowercase_without_space>_<current_time_in_milliseconds>.json. For example, halifax_1625417503199.json and canadaeducation_1625417513842.json.

I have used JSON-java library (Package org.json) [6] in the Extraction Engine to store the response from NewsAPI [4].

⇒ **Extraction Engine Pseudocode [7]**

The pseudocode for the Extraction Engine contains four methods – extractNewsData, fetchNews, prepareAndStoreNews and storeNews. Implementation of the Extraction Engine is in file [NewsAPIExtractionEngine.java](#) [5].

extractNewsData()

1. BEGIN
2. SET searchKeywords to all the search keywords
3. FOR each searchKeyword in the searchKeywords
 - 3.1. CALL **fetchNews** method with argument searchKeyword RETURNING newsJSON
 - 3.2. IF newsJSON is not NULL THEN
 - 3.2.1. CALL **prepareAndStoreNews** method with arguments searchKeyword, newsJSON
 - 3.3. ENDIF
4. ENDFOR
5. END

fetchNews(searchKeyword)

1. BEGIN
2. REPLACE all " "(i.e., space) in searchKeyword to "%20"
3. SET newsAPIURL to the NewsAPI URL with query parameters "q"(query) SET to searchKeyword, "language" SET to "en", and "pageSize" SET to "100"

4. SET httpRequest to a new HttpRequest object with arguments newsAPIURL, API key passed in header field “X-API-Key”, timeout duration set to 30 seconds, method set to HTTP GET
5. SET httpClient to a new HttpClient object
6. CALL **send** method of httpClient with arguments httpRequest, responseBodyHandler RETURNING httpResponse.
7. IF httpResponse’s status code is not 200 THEN
 - 7.1. RETURN NULL
8. ELSE
 - 8.1. RETURN httpResponse’s JSON body
9. ENDIF
10. END

prepareAndStoreNews(searchKeyword, newsJSON)

1. BEGIN
2. SET newsJSONObject to a new JSONObject object with arguments newsJSON
3. SET newsArticlesArray to newsJSONObject.getJSONArray with arguments “articles”
4. SET articlesStringBuilder to an empty StringBuilder object
5. SET TOTAL_ARTICLES_ALLOWED_IN_EACH_FILE to 5
6. SET totalArticlesInEachFile to 0
7. FOR each article JSON object in the newsArticlesArray
 - 7.1. APPEND article JSON object to articlesStringBuilder
 - 7.2. APPEND “,” to articlesStringBuilder
 - 7.3. INCREMENT totalArticlesInEachFile
 - 7.4. IF totalArticlesInEachFile is equal to TOTAL_ARTICLES_ALLOWED_IN_EACH_FILE THEN
 - 7.4.1. REMOVE last “,” from articlesStringBuilder
 - 7.4.2. SET articlesInEachFile to empty String object
 - 7.4.3. APPEND “[“ to articlesInEachFile
 - 7.4.4. APPEND articlesStringBuilder to articlesInEachFile
 - 7.4.5. APPEND “]“ to articlesInEachFile
 - 7.4.6. CALL **storeNews** method with arguments searchKeyword, articlesInEachFile
 - 7.4.7. SET totalArticlesInEachFile again to 0
 - 7.4.8. CLEAR articlesStringBuilder
 - 7.5. ENDIF
8. ENDFOR
9. IF totalArticlesInEachFile is greater than 0 THEN
 - 9.1. REMOVE last “,” from articlesStringBuilder
 - 9.2. SET articlesInEachFile to empty String object
 - 9.3. APPEND “[“ to articlesInEachFile
 - 9.4. APPEND articlesStringBuilder to articlesInEachFile
 - 9.5. APPEND “]“ to articlesInEachFile
 - 9.6. CALL **storeNews** method with arguments searchKeyword, articlesInEachFile
 - 9.7. CLEAR articlesStringBuilder
10. ENDIF
11. END

storeNews(searchKeyword, newsArticleJSON)

1. BEGIN
2. REPLACE all “ ”(i.e., space) in searchKeyword to “”
3. CONVERT searchKeyword to lower case String object
4. SET NEWS_DATA_DIRECTORY_NAME to “./newsData”
5. SET path to directory NEWS_DATA_DIRECTORY_NAME
6. IF path does not exists THEN
 - 6.1. CREATE NEWS_DATA_DIRECTORY_NAME at the path
7. SET newsDataFileName to searchKeyword_<current_time_in_milliseconds>.json
8. SET fileWriter to a new FileWriter object with arguments newsDataFileName, UTF_8
9. WRITE newsArticleJSON to fileWriter (i.e., in newsDataFileName)
10. END

⇒ **Figure 10** displays a total of 140 JSON files containing 700 news articles extracted for 7 search keywords. For each search keyword, there are total of 20 files.

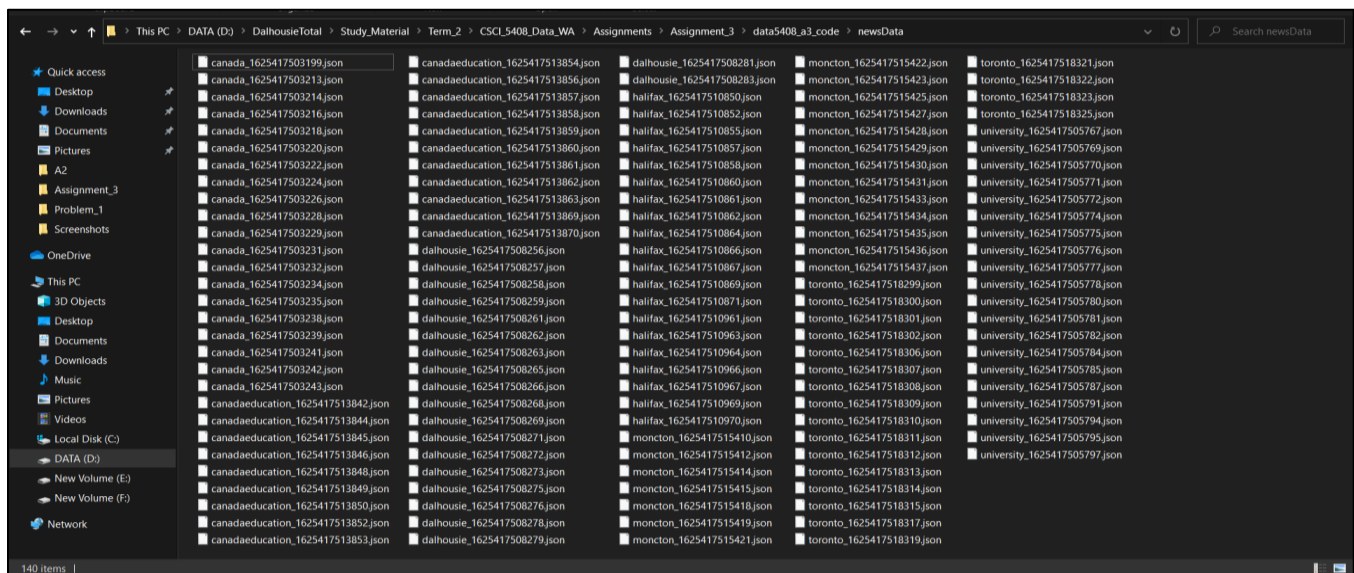
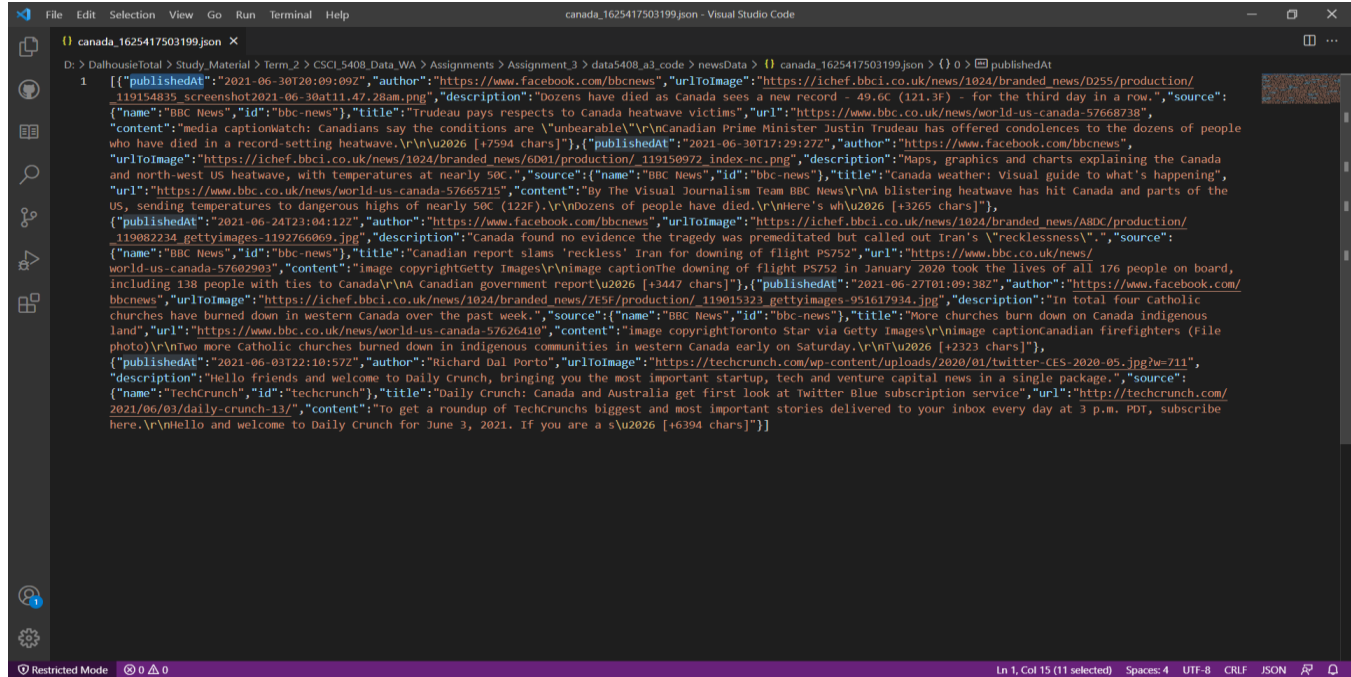


Figure 10 - 140 JSON files created after extraction of news articles for 7 search keywords

⇒ **Figure 11** displays the sample of one of the 140 JSON files extracted.



```

1 [{"publishedAt": "2021-06-30T20:09:09Z", "author": "https://www.facebook.com/bbcnews", "urlToImage": "https://ichef.bbci.co.uk/news/1024/branded_news/D255/production/119154835_screenshot2021-06-30at11.47.28am.png", "description": "Dozens have died as Canada sees a new record - 49.6C (121.3F) - for the third day in a row.", "source": {"name": "BBC News", "id": "bbc-news"}, "title": "Trudeau pays respects to Canada heatwave victims", "url": "https://www.bbc.co.uk/news/world-us-canada-57668738", "content": "media captionwatch: Canadians say the conditions are \"unbearable\"\\n\\nCanadian Prime Minister Justin Trudeau has offered condolences to the dozens of people who have died in a record-setting heatwave.\\n\\n\\u0026 [7594 chars]\"}, {\"publishedAt\": \"2021-06-30T17:29:27Z\", \"author\": \"https://www.facebook.com/bbcnews\", \"urlToImage\": \"https://ichef.bbci.co.uk/news/1024/branded_news/6D01/production/119150972_index-nc.png\", \"description\": \"Maps, graphics and charts explaining the Canada and north-west US heatwave, with temperatures at nearly 50C.\", \"source\": {\"name\": \"BBC News\", \"id\": \"bbc-news\"}, \"title\": \"Canada weather: Visual guide to what's happening\", \"url\": \"https://www.bbc.co.uk/news/world-us-canada-57665715\", \"content\": \"By The Visual Journalism Team BBC News\\n\\nA blistering heatwave has hit Canada and parts of the US, sending temperatures to dangerous highs of nearly 50C (122F).\\n\\nDozens of people have died.\\n\\nWhere's wh\\u0026 [3265 chars]\"}, {\"publishedAt\": \"2021-06-24T23:04:12Z\", \"author\": \"https://www.facebook.com/bbcnews\", \"urlToImage\": \"https://ichef.bbci.co.uk/news/1024/branded_news/A8DC/production/119082234_gettyimages-1192766069.jpg\", \"description\": \"Canada found no evidence the tragedy was premeditated but called out Iran's 'recklessness'\", \"source\": {\"name\": \"BBC News\", \"id\": \"bbc-news\"}, \"title\": \"Canadian report slams 'reckless' Iran for downing of flight PS752\", \"url\": \"https://www.bbc.co.uk/news/world-us-canada-57602903\", \"content\": \"image copyrightGetty Images\\n\\nimage captionThe downing of flight PS752 in January 2020 took the lives of all 176 people on board, including 138 people with ties to Canada\\n\\nA Canadian government report\\u0026 [3447 chars]\"}, {\"publishedAt\": \"2021-06-27T01:09:38Z\", \"author\": \"https://www.facebook.com/bbcnews\", \"urlToImage\": \"https://ichef.bbci.co.uk/news/1024/branded_news/7E5F/production/119015323_gettyimages-951617934.jpg\", \"description\": \"In total four Catholic churches have burned down in western Canada over the past week.\", \"source\": {\"name\": \"BBC News\", \"id\": \"bbc-news\"}, \"title\": \"More churches burn down on Canada Indigenous land\", \"url\": \"https://www.bbc.co.uk/news/world-us-canada-57626410\", \"content\": \"image copyrightToronto Star via Getty Images\\n\\nimage captionCanadian firefighters (file photo)\\n\\nTwo more Catholic churches burned down in Indigenous communities in western Canada early on Saturday.\\n\\n\\u0026 [2323 chars]\"}, {\"publishedAt\": \"2021-06-03T22:10:57Z\", \"author\": \"Richard Dal Porto\", \"urlToImage\": \"https://techcrunch.com/wp-content/uploads/2020/01/twitter-CES-2020-05.jpg?w=711\", \"description\": \"Hello friends and welcome to Daily Crunch, bringing you the most important startup, tech and venture capital news in a single package.\", \"source\": {\"name\": \"TechCrunch\", \"id\": \"techcrunch\"}, \"title\": \"Daily crunch: Canada and Australia get first look at Twitter Blue subscription service\", \"url\": \"http://techcrunch.com/2021/06/03/daily-crunch-13/\", \"content\": \"To get a roundup of TechCrunch's biggest and most important stories delivered to your inbox every day at 3 p.m. PDT, subscribe here.\\n\\nHello and welcome to Daily Crunch for June 3, 2021. If you are a s\\u0026 [6394 chars]\"}]

```

Figure 11 - Sample JSON file with its content

Filtration Engine

The Filtration Engine is responsible for cleaning and transforming the news articles stored in the JSON files. Further, the news articles are to be uploaded on the MongoDB Atlas [8] under database myMongoNews. Cleaning and transforming of data are to be done using custom regular expressions.

For each JSON file, first, I am filtering the file content using the custom regular expressions. **Table 1** is a list of regexes used for filtration and their working:

Table 1 – List of regexes and their description

Regex	Description
<code>[^p{L}\p{N}\p{P}\p{Z}]</code>	This regex removes all the emojis from news articles.
<code>("urlToImage":("http.*?" null "null" ""),)</code>	This regex removes the urlToImage field from all news articles.
<code>("url":("http.*?" null "null" ""),)</code>	This regex removes the url field from all news articles.
<code>("author":("http.*?" null "null" ""),)</code>	This regex removes the author field from all news articles if the author is url or empty string or null.
<code>(,"id":(null "null" ""))</code>	This regex removes the id field from all news articles if the id is a null or empty string.
<code>(\\[ntr])(NBSP)((<[>]*>))</code>	This regex removes general things like \n, \r, \t, non-breaking space and HTML tags from all news articles.

After filtration of each file, the JSON string representing a single news article is parsed into a MongoDB Document and stored in a list. Also, I am keeping track of the total articles read and filtered using the variable totalArticlesRead. Finally, after filtering all the extracted files, a connection is established with the MongoDB using a connection string. Then, using the insertMany() command of MongoDB, I am uploading the list of documents into the database myMongoNews under the collection mongoNews. This uploads total of 700 articles on MongoDB.

I have used MongoDB Java Driver (Version – 3.12.8) [9] for establishing a connection with MongoDB Atlas [8].

⇒ **Filtration Engine Flowchart** [10]

Filtration Engine flowchart describes the process flow of the filtration engine to achieve the filtered dataset and upload it to the MongoDB database. Implementation of the Filtration Engine is in file **NewsAPIFiltrationEngine.java**.

⇒ **Figure 12** displays the flowchart of the Filtration Engine.

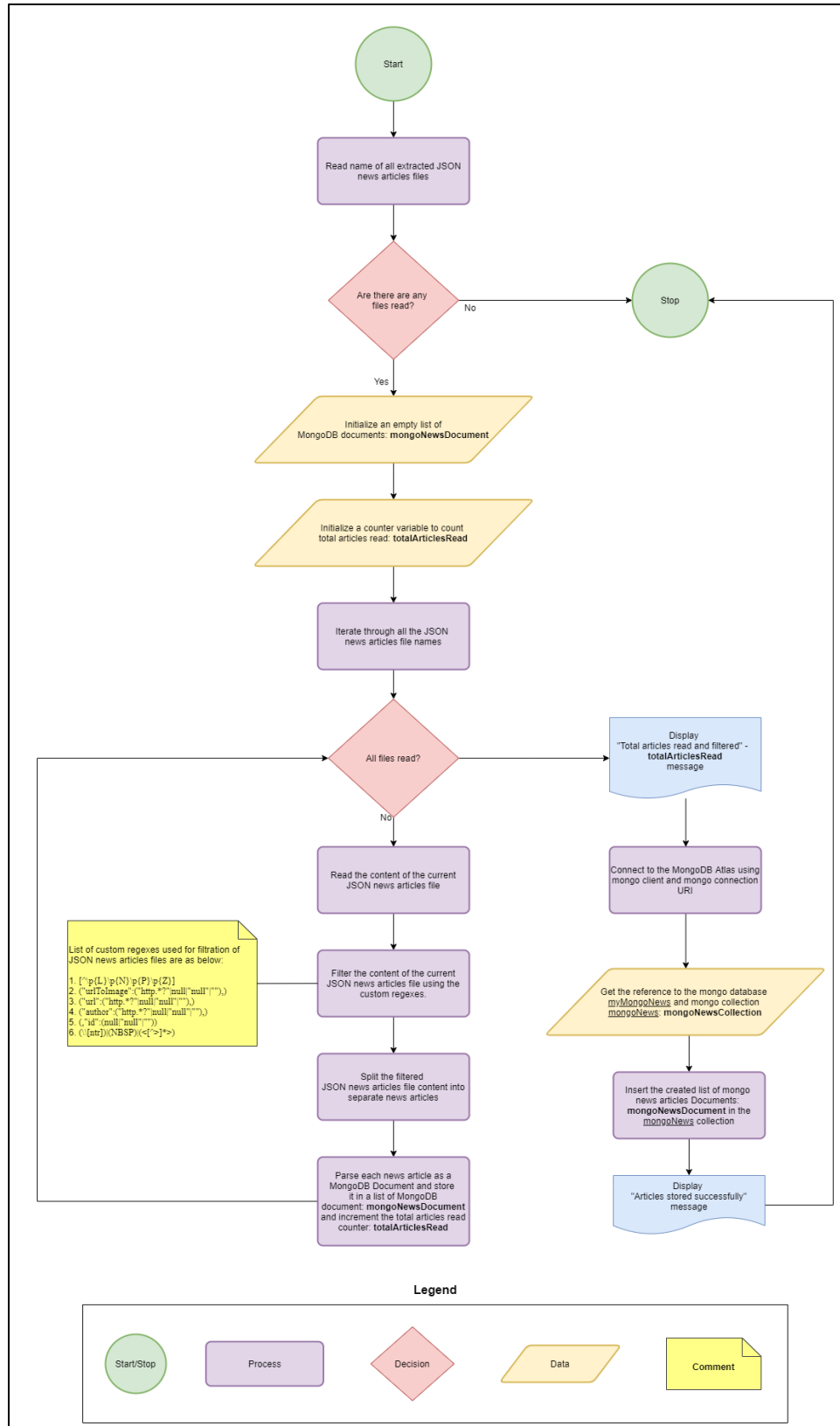


Figure 12 - Filtration Engine Flowchart [10]

⇒ **Figure 13** displays the snapshot of news articles uploaded on MongoDB under database myMongoNews in collection mongoNews. In total, 700 news articles are uploaded.

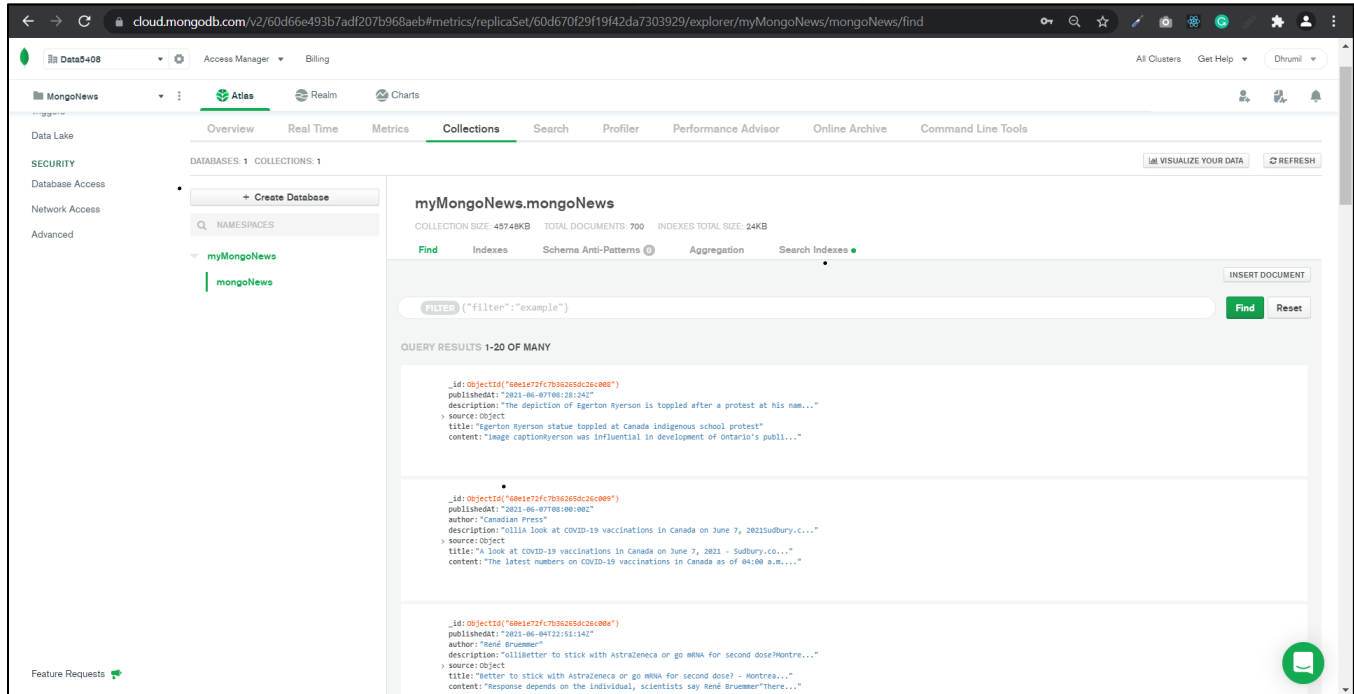


Figure 13 - Snapshot of data uploaded on MongoDB (700 news articles) [8]

Problem 1 – Task 3

WordCounter Engine

The WordCounter Engine is responsible for counting the frequency of searched keywords in the raw news files. For counting the frequency, only titles and contents of news articles are considered. The keywords for which frequency count is to be calculated are Canada, Nova Scotia, education, higher, learning, city, accommodation, and price. These keywords are to be searched and counted in case sensitive manner. The working of all the private methods in class NewsAPIWordCounter is as below:

1. **initWordCounterMap()** - This method creates a hashmap of the keywords to be searched as a string and their frequency count as value. Initially, the frequency of each of these keywords is set to 0.
2. **readAllFileNames()** - This method creates an array of type File and stores the names of all the raw news files in it.
3. **performMapReduce()** - This method iterates through all the file names stored in the array of type File, reads the contents of each file and performs map and reduce operations.
4. **map()** - This method reads the news articles present in each of the read files and, using the Matcher and Pattern classes, it compiles the regular expressions for extracting title and content from each of the news articles. The matched substrings for title and content are appended to a StringBuilder object whose string value is returned.
5. **reduce()** - This method is invoked for every string returned by the **map()** method. It will search for the keyword in the string until the last index is reached and will increase the word count of the subsequent keyword in the **wordCounterMap** hashmap by 1 every time it encounters a match.

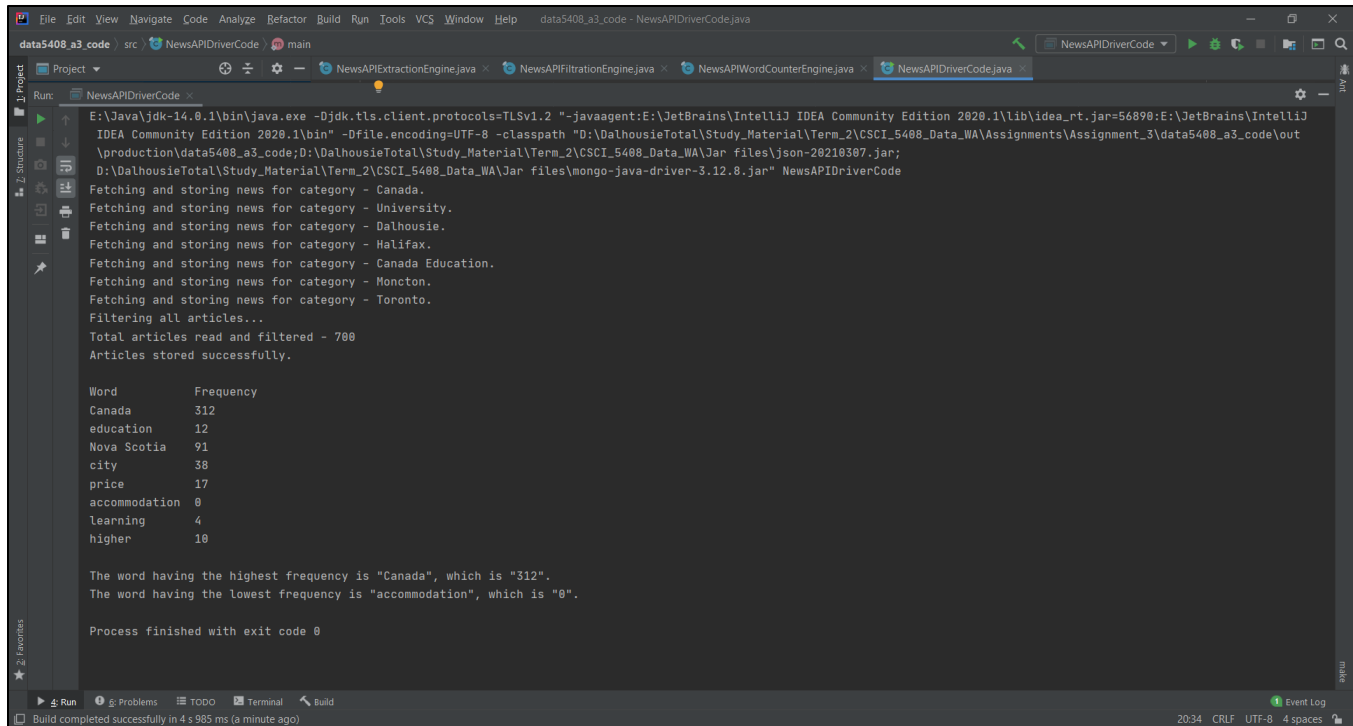
⇒ **WordCounter Engine Algorithm**

- Step 1: Start
- Step 2: Initialize variable **wordCounterMap** which is an empty word counter hashmap having the key of type String and the value of type Integer.
- Step 3: Initialize the variable **initialWordCount** to 0.
- Step 4: For all the keywords to be searched, add keyword name as the key and **initialWordCount** as the value in **wordCounterMap**.
- Step 5: Initialize the File array with the variable name **allNewsFiles** to store the name of all the news articles JSON files.
- Step 6: If **wordCounterMap** is empty, go to step 25.
- Step 7: If **allNewsFiles** is null or empty, go to step 25.
- Step 8: Iterate through all the JSON news articles file names stored in **allNewsFiles**.
- Step 9: For each JSON news articles file, read the contents of the file in the variable **newsFileContent**.
- Step 10: Initialize the variable **titleMatcher** which is an object of the Matcher class to match the pattern compiling a regex matching the title in the **newsFileContent**.
- Step 11: Initialize the variable **contentMatcher** which is an object of the Matcher class to match the pattern compiling a regex matching the content in the **newsFileContent**.
- Step 12: Initialize the variable **mappedStringBuilder** to an empty StringBuilder object.
- Step 13: Append all the matches found by **titleMatcher**'s find method to the **mappedStringBuilder** object.
- Step 14: Append all the matches found by **contentMatcher**'s find method to the **mappedStringBuilder** object.
- Step 15: Convert the **mappedStringBuilder** to a string value and store it in the variable **mappedString**.
- Step 16: Iterate through all the keywords to be searched.
- Step 17: Initialize the variable **lastEncounterIndex** to 0 for the current keyword to be searched.
- Step 18: Iterate till the **lastEncounterIndex** is not equal to -1.
- Step 19: Find the index of the occurrence of the current keyword to be searched from the **lastEncounterIndex** and assign it to **lastEncounterIndex**.
- Step 20: If the **lastEncounterIndex** is not equal to -1, go to step 21.
- Step 21: Increment the word count of the current keyword to be searched in the **wordCounterMap**.
- Step 22: Update the **lastEncounteredIndex** by adding the current keyword to be searched length to **lastEncounteredIndex**.
- Step 23: Continue till all the files are read.
- Step 24: Return the **wordCounterMap**
- Step 25: Stop

⇒ **Program Output****Table 2 – Words having highest and lowest frequencies**

Word	Frequency
Canada	312 – Highest frequency count
accommodation	0 – Lowest frequency count

⇒ **Figure 14** displays the output of executing the Extraction Engine ([NewsAPIExtractionEngine.java](#)), Filtration Engine ([NewsAPIFiltrationEngine.java](#)), and WordCounter Engine ([NewsAPIWordCounterEngine.java](#)). The [NewsAPIDriverCode.java](#) is the main driver class that runs all three java files.



```
E:\Java\jdk-14.0.1\bin\java.exe -Djdk.tls.client.protocols=TLSv1.2 --javaagent:E:\JetBrains\IntelliJ IDEA Community Edition 2020.1\lib\idea_rt.jar=56890:E:\JetBrains\IntelliJ IDEA Community Edition 2020.1\bin" -Dfile.encoding=UTF-8 -classpath "D:\DalhousieTotal\Study_Material\Term_2\CSCI_5408_Data_WA\Assignments\Assignment_3\data5408_a3_code\out\production\data5408_a3_code;D:\DalhousieTotal\Study_Material\Term_2\CSCI_5408_Data_WA\Jar_files\json-20210307.jar;D:\DalhousieTotal\Study_Material\Term_2\CSCI_5408_Data_WA\Jar_files\mongo-java-driver-3.12.8.jar" NewsAPIDriverCode
Fetching and storing news for category - Canada.
Fetching and storing news for category - University.
Fetching and storing news for category - Dalhousie.
Fetching and storing news for category - Halifax.
Fetching and storing news for category - Canada Education.
Fetching and storing news for category - Moncton.
Fetching and storing news for category - Toronto.
Filtering all articles...
Total articles read and filtered - 700
Articles stored successfully.

Word      Frequency
Canada    312
education 12
Nova Scotia 91
city       38
price      17
accommodation 0
learning   4
higher     10

The word having the highest frequency is "Canada", which is "312".
The word having the lowest frequency is "accommodation", which is "0".

Process finished with exit code 0
```

Figure 14 - Snapshot of program output for Extraction, Filtration and WordCounter Engine Java files

Problem 2 – Task 1

Task Description

Data Visualization using Graph Database – Neo4j for graph generation

Website Visited

<https://parks.novascotia.ca/region/nova-scotia> [11]

Regions (Total regions added – 7)

Query to CREATE [12] regions in Nova Scotia (NS) is as below:

```
CREATE
(cbi:region{name:'Cape Breton Island',short_region_name:'Cape Breton Region',province:'NS'}),
(nus:region{name:'Northumberland Shore',short_region_name:'North Region',province:'NS'}),
(es:region{name:'Eastern Shore',short_region_name:'Eastern Region',province:'NS'}),
(hr:region{name:'Halifax Region',short_region_name:'HRM',province:'NS'}),
(fsaav:region{name:'Fundy Shore and Annapolis Valley',short_region_name:'Valley
Region',province:'NS'}),
(ss:region{name:'South Shore',short_region_name:'Southern Region',province:'NS'}),
(yaas:region{name:'Yarmouth and Acadian Shores',short_region_name:'Western
Region',province:'NS'});
```

Parks in Cape Breton Island (Total parks added - 9)

Query to CREATE [12] parks in Cape Breton Island region is as below:

```
CREATE
(clpp:park{name:'Cabots Landing Provincial Park',location:'Sugar Loaf',street_address:'1904 Bay St
Lawrence Rd',province:'NS',zip_code:'B0C1R0'}),
(cspp:park{name:'Cape Smokey Provincial Park',location:'Ingonish Beach',street_address:'40301 Cabot
Trail Ingonish Beach',province:'NS',zip_code:'B0C1L0'}),
(tbpp:park{name:'Trout Brook Provincial Park',location:'East Lake Ainslie',street_address:'2535
Highway 395',province:'NS',zip_code:'B0E3E0'}),
(mrpp:park{name:'Mira River Provincial Park',location:'Albert Bridge',street_address:'439 Brickyard
Rd',province:'NS',zip_code:'B1K2R9'}),
(bpp:park{name:'Barrachois Provincial Park',location:'Barrachois',street_address:'2315 Highway
223',province:'NS',zip_code:'A0N2C0'}),
(bepp:park{name:'Ben Eoin Provincial Park',location:'Ben Eoin',street_address:'5549 Highway
4',province:'NS',zip_code:'B1J1N5'}),
(gppp:park{name:'Groves Point Provincial Park',location:'Groves Point',street_address:'1055 Hillside
Boularderie Rd',province:'NS',zip_code:'B1Y2V7'}),
(dlpp:park{name:'Dalem Lake Provincial Park',location:'Boularderie',street_address:'220 New Dominion
Rd',province:'NS',zip_code:'B1Y3Z3'}),
(rfmp:park{name:'Ross Ferry Marine Park',location:'Ross Ferry',street_address:'9685 Kempt Head
Road',province:'NS',zip_code:'B1X1N9'});
```

Parks in Northumberland Shore (Total parks added – 8)

Query to CREATE [12] parks in Northumberland Shore Island region is as below:

CREATE

```
(aspp:park{ name:'Amherst Shore Provincial Park',location:'Northport',street_address:'6596 NS-366',province:'NS',zip_code:'B0L1E0'}),
(cmipp:park{ name:'Caribou/Munroes Island Provincial Park',location:'Pictou',street_address:'2119 Three Brooks Rd',province:'NS',zip_code:'B0K1H0'}),
(tdpp:park{ name:'Tidnish Dock Provincial Park',location:'Amherst',street_address:'Prounis Park',province:'NS',zip_code:'B4H3X9'}),
(nbpp:park{ name:'Northport Beach Provincial Park',location:'Amherst',street_address:'8301 Tyndal Rd',province:'NS',zip_code:'B4H3Y2'}),
(gspp:park{ name:'Gulf Shore Provincial Park',location:'Pugwash',street_address:'1033 Gulf Shore Rd',province:'NS',zip_code:'B0K1L0'}),
(ghpp:park{ name:'Green Hill Provincial Park',location:'Scotsburn',street_address:'209 Dan Fraser Rd',province:'NS',zip_code:'B0K1R0'}),
(app:park{ name:'Arisaig Provincial Park',location:'Antigonish',street_address:'5704 NS-245',province:'NS',zip_code:'B2G2L2'}),
(bbpp:park{ name:'Bayfield Beach Provincial Park',location:'Afton Station',street_address:'151 Bayfield Beach Rd',province:'NS',zip_code:'B0H1A0'});
```

Parks in Eastern Shore (Total parks added – 7)

Query to CREATE [12] parks in Eastern Shore region is as below:

CREATE

```
(bdcpp:park{ name:'Black Duck Cove Provincial Park',location:'Little Dover',street_address:'Little Dover',province:'NS',zip_code:'B0H1V0'}),
(spp:park{ name:'Sherbrooke Provincial Park',location:'BESbswy',street_address:'8407 Hwy 7 Sherbrooke',province:'NS',zip_code:'B0J3C0'}),
(bspp:park{ name:'Boylston Provincial Park',location:'Guysborough',street_address:'11131 NS-16',province:'NS',zip_code:'B0H1N0'}),
(sspp:park{ name:'Salsman Provincial Park',location:'Bickerton West',street_address:'15641 NS-316',province:'NS',zip_code:'B0J1A0'}),
(thpp:park{ name:'Taylor Head Provincial Park',location:'Spry Bay',street_address:'20140 Hwy 7',province:'NS',zip_code:'B0J3H0'}),
(ebpp:park{ name:'Elderbank Provincial Park',location:'Middle Musquodoboit',street_address:'5819 NS-357',province:'NS',zip_code:'B0N1X0'}),
(plpp:park{ name:'Porters Lake Provincial Park',location:'West Porters Lake',street_address:'1160 W Porters Lake Rd',province:'NS',zip_code:'B3E1L4'});
```

Parks in Halifax Region (Total parks added – 6)

Query to CREATE [12] parks in Halifax Region is as below:

CREATE

```
(opp:park{name:'Oakfield Provincial Park',location:'Oakfield',street_address:'366 Oakfield Park Rd',province:'NS',zip_code:'B2T1B3'}),
(lpp:park{name:'Laurie Provincial Park',location:'Grand Lake',street_address:'4949 Nova Scotia Trunk 2',province:'NS',zip_code:'B2T0S5'}),
(jlpp:park{name:'Jerry Lawrence Provincial Park',location:'Upper Tantallon',street_address:'4775 St Margarets Bay Rd',province:'NS',zip_code:'B3Z1N5'}),
(mlipp:park{name:'McNabs and Lawlor Islands Provincial Park',location:'Shearwater',street_address:'Shearwater',province:'NS',zip_code:'B0J3A0'}),
(mcbpp:park{name:'MacCormack Beach Provincial Park',location:'Iona',street_address:'St Columba Rd',province:'NS',zip_code:'B2C1B4'}),
(ccbpp:park{name:'Crystal Crescent Beach Provincial Park',location:'Sambro Creek',street_address:'220 Sambro Creek Rd',province:'NS',zip_code:'B3V1L8'});
```

Parks in Fundy Shore and Annapolis Valley (Total parks added – 7)

Query to CREATE [12] parks in Fundy Shore and Annapolis Valley region is as below:

CREATE

```
(sbpp:park{name:'Scots Bay Provincial Park',location:'Canning',street_address:'24 Wharf Rd',province:'NS',zip_code:'B0P1H0'}),
(vvpp:park{name:'Valleyview Provincial Park',location:'Hampton',street_address:'960 Hampton Mountain Rd',province:'NS',zip_code:'B0S1L0'}),
(fipp:park{name:'Five Islands Provincial Park',location:'Five Islands',street_address:'618 Bentley Branch Rd',province:'NS',zip_code:'B0M1K0'}),
(crlpp:park{name:'Caddell Rapids Lookoff Provincial Park',location:'Riverside',street_address:'1609 Riverside Rd',province:'NS',zip_code:'B0N2J0'}),
(spp:park{name:'Savary Provincial Park',location:'Plympton',street_address:'Plympton',province:'NS',zip_code:'B0W2R0'}),
(wpp:park{name:'Wentworth Provincial Park',location:'Wentworth',street_address:'Valley Rd',province:'NS',zip_code:'B0M1Z0'}),
(mhpp:park{name:'Mickey Hill Provincial Park',location:'Annapolis',street_address:'7906 Nova Scotia Trunk 8',province:'NS',zip_code:'B0S1A0'});
```

Parks in South Shore (Total parks added – 7)

Query to CREATE [12] parks in South Shore region is as below:

CREATE

```
(srpp:park{name:'Sable River Provincial Park',location:'Lockeport',street_address:'140 W Sable Rd',province:'NS',zip_code:'B0T1L0'}),
(trpp:park{name:'Thomas Raddall Provincial Park',location:'Port Joli',street_address:'529 Raddall Park Rd',province:'NS',zip_code:'B0T1S0'}),
(cbpp:park{name:'Camerons Brook Provincial Park',location:'South Brookfield',street_address:'8000 Nova Scotia Trunk 8',province:'NS',zip_code:'B0T1X0'}),
(cpp:park{name:'Cookville Provincial Park',location:'Lower Branch',street_address:'238 Lower Branch Rd',province:'NS',zip_code:'B4V4L9'});
```

```
(bwbpp:park{name:'Bayswater Beach Provincial Park',location:'Hubbards',street_address:'4015 NS-329',province:'NS',zip_code:'B0J1T0'}),
(flpp:park{name:'Fancy Lake Provincial Park',location:'Conquerall Mills',street_address:'854 Conquerall Mills Rd',province:'NS',zip_code:'B4V6A2'}),
(clbpp:park{name:'Cleveland Beach Provincial Park',location:'Black Point',street_address:'8880 St Margarets Bay Rd',province:'NS',zip_code:'B0J1B0'});
```

Parks in Yarmouth and Acadian Shores (Total parks added – 5)

Query to CREATE [12] parks in Acadian Shores region is as below:

CREATE

```
(elpp:park{name:'Ellenwood Lake Provincial Park',location:'Yarmouth',street_address:'1888 Mood Rd',province:'NS',zip_code:'B5A4A8'}),
(gpp:park{name:'Glenwood Provincial Park',location:'Glenwood',street_address:'ROUTE 3 & HIGHWAY 103',province:'NS',zip_code:'B0W1W0'}),
(pmpp:park{name:'Port Maitland Provincial Park',location:'Port Maitland',street_address:'Spider Rd',province:'NS',zip_code:'B5A4A5'}),
(mbpp:park{name:'Mavillette Beach Provincial Park',location:'Mavillette',street_address:'124 John Doucette Rd',province:'NS',zip_code:'B0W2H0'}),
(scpp:park{name:'Smugglers Cove Provincial Park',location:'Meteghan',street_address:'7651 Highway 1',province:'NS',zip_code:'B0W2J0'});
```

Relationships from parks in Cape Breton Island to region Cape Breton Island

Queries to CREATE [12] relationships from parks in Cape Breton Island to region Cape Breton Island are as below:

```
MATCH (cbi:region),(clpp:park) WHERE cbi.name='Cape Breton Island' AND clpp.name='Cabots Landing Provincial Park' CREATE (clpp)-[clpp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(cspp:park) WHERE cbi.name='Cape Breton Island' AND cspp.name='Cape Smokey Provincial Park' CREATE (cspp)-[cspp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(tbpp:park) WHERE cbi.name='Cape Breton Island' AND tbpp.name='Trout Brook Provincial Park' CREATE (tbpp)-[tbpp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(mrpp:park) WHERE cbi.name='Cape Breton Island' AND mrpp.name='Mira River Provincial Park' CREATE (mrpp)-[mrpp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(bpp:park) WHERE cbi.name='Cape Breton Island' AND bpp.name='Barrachois Provincial Park' CREATE (bpp)-[bpp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(bepp:park) WHERE cbi.name='Cape Breton Island' AND bepp.name='Ben Eoin Provincial Park' CREATE (bepp)-[bepp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(gppp:park) WHERE cbi.name='Cape Breton Island' AND gppp.name='Groves Point Provincial Park' CREATE (gppp)-[gppp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(dlpp:park) WHERE cbi.name='Cape Breton Island' AND dlpp.name='Dalem Lake Provincial Park' CREATE (dlpp)-[dlpp_cbi:BELONGS_TO]->(cbi);
MATCH (cbi:region),(rfmp:park) WHERE cbi.name='Cape Breton Island' AND rfmp.name='Ross Ferry Marine Park' CREATE (rfmp)-[rfmp_cbi:BELONGS_TO]->(cbi);
```

Relationships from parks in Northumberland Shore to region Northumberland Shore

Queries to CREATE [12] relationships from parks in Northumberland Shore to region Northumberland Shore are as below:

```
MATCH (nus:region),(aspp:park) WHERE nus.name='Northumberland Shore' AND
aspp.name='Amherst Shore Provincial Park' CREATE (aspp)-[aspp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(cmipp:park) WHERE nus.name='Northumberland Shore' AND
cmipp.name='Caribou/Munroes Island Provincial Park' CREATE (cmipp)-
[cmipp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(tdpp:park) WHERE nus.name='Northumberland Shore' AND
tdpp.name='Tidnish Dock Provincial Park' CREATE (tdpp)-[tdpp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(nbpp:park) WHERE nus.name='Northumberland Shore' AND
nbpp.name='Northport Beach Provincial Park' CREATE (nbpp)-[nbpp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(gspp:park) WHERE nus.name='Northumberland Shore' AND gspp.name='Gulf
Shore Provincial Park' CREATE (gspp)-[gspp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(ghpp:park) WHERE nus.name='Northumberland Shore' AND ghpp.name='Green
Hill Provincial Park' CREATE (ghpp)-[ghpp_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(app:park) WHERE nus.name='Northumberland Shore' AND app.name='Arisaig
Provincial Park' CREATE (app)-[app_nus:BELONGS_TO]->(nus);
MATCH (nus:region),(bbpp:park) WHERE nus.name='Northumberland Shore' AND
bbpp.name='Bayfield Beach Provincial Park' CREATE (bbpp)-[bbpp_nus:BELONGS_TO]->(nus);
```

Relationships from parks in Eastern Shore to region Eastern Shore

Queries to CREATE [12] relationships from parks in Eastern Shore to region Eastern Shore are as below:

```
MATCH (es:region),(bdcpp:park) WHERE es.name='Eastern Shore' AND bdcpp.name='Black Duck
Cove Provincial Park' CREATE (bdcpp)-[bdcpp_es:BELONGS_TO]->(es);
MATCH (es:region),(spp:park) WHERE es.name='Eastern Shore' AND spp.name='Sherbrooke
Provincial Park' CREATE (spp)-[spp_es:BELONGS_TO]->(es);
MATCH (es:region),(bspp:park) WHERE es.name='Eastern Shore' AND bspp.name='Boylston
Provincial Park' CREATE (bspp)-[bspp_es:BELONGS_TO]->(es);
MATCH (es:region),(sspp:park) WHERE es.name='Eastern Shore' AND sspp.name='Salsman
Provincial Park' CREATE (sspp)-[sspp_es:BELONGS_TO]->(es);
MATCH (es:region),(thpp:park) WHERE es.name='Eastern Shore' AND thpp.name='Taylor Head
Provincial Park' CREATE (thpp)-[thpp_es:BELONGS_TO]->(es);
MATCH (es:region),(ebpp:park) WHERE es.name='Eastern Shore' AND ebpp.name='Elderbank
Provincial Park' CREATE (ebpp)-[ebpp_es:BELONGS_TO]->(es);
MATCH (es:region),(plpp:park) WHERE es.name='Eastern Shore' AND plpp.name='Porters Lake
Provincial Park' CREATE (plpp)-[plpp_es:BELONGS_TO]->(es);
```

Relationships from parks in Halifax Region to region Halifax

Queries to CREATE [12] relationships from parks in Halifax Region to region Halifax are as below:

```
MATCH (hr:region),(opp:park) WHERE hr.name='Halifax Region' AND opp.name='Oakfield
Provincial Park' CREATE (opp)-[opp_hr:BELONGS_TO]->(hr);
```


MATCH (hr:region),(lpp:park) WHERE hr.name='Halifax Region' AND lpp.name='Laurie Provincial Park' CREATE (lpp)-[lpp_hr:BELONGS_TO]->(hr);
 MATCH (hr:region),(jlpp:park) WHERE hr.name='Halifax Region' AND jlpp.name='Jerry Lawrence Provincial Park' CREATE (jlpp)-[jlpp_hr:BELONGS_TO]->(hr);
 MATCH (hr:region),(mlpp:park) WHERE hr.name='Halifax Region' AND mlpp.name='McNabs and Lawlor Islands Provincial Park' CREATE (mlpp)-[mlpp_hr:BELONGS_TO]->(hr);
 MATCH (hr:region),(mcbpp:park) WHERE hr.name='Halifax Region' AND mcbpp.name='MacCormack Beach Provincial Park' CREATE (mcbpp)-[mcbpp_hr:BELONGS_TO]->(hr);
 MATCH (hr:region),(ccbpp:park) WHERE hr.name='Halifax Region' AND ccbpp.name='Crystal Crescent Beach Provincial Park' CREATE (ccbpp)-[ccbpp_hr:BELONGS_TO]->(hr);

Relationships from parks in Fundy Shore and Annapolis Valley to region Fundy Shore and Annapolis Valley

Queries to CREATE [12] relationships from parks in Fundy Shore and Annapolis Valley to region Fundy Shore and Annapolis Valley are as below:

MATCH (fsaav:region),(sbpp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND sbpp.name='Scots Bay Provincial Park' CREATE (sbpp)-[sbpp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(vvpp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND vvpp.name='Valleyview Provincial Park' CREATE (vvpp)-[vvpp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(fipp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND fipp.name='Five Islands Provincial Park' CREATE (fipp)-[fipp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(crlpp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND crlpp.name='Caddell Rapids Lookoff Provincial Park' CREATE (crlpp)-[crlpp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(spp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND spp.name='Savary Provincial Park' CREATE (spp)-[spp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(wpp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND wpp.name='Wentworth Provincial Park' CREATE (wpp)-[wpp_fsaav:BELONGS_TO]->(fsaav);
 MATCH (fsaav:region),(mhpp:park) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND mhpp.name='Mickey Hill Provincial Park' CREATE (mhpp)-[mhpp_fsaav:BELONGS_TO]->(fsaav);

Relationships from parks in South Shore to region South Shore

Queries to CREATE [12] relationships from parks in South Shore to region South Shore are as below:

MATCH (ss:region),(srpp:park) WHERE ss.name='South Shore' AND srpp.name='Sable River Provincial Park' CREATE (srpp)-[srpp_ss:BELONGS_TO]->(ss);
 MATCH (ss:region),(trpp:park) WHERE ss.name='South Shore' AND trpp.name='Thomas Raddall Provincial Park' CREATE (trpp)-[trpp_ss:BELONGS_TO]->(ss);
 MATCH (ss:region),(cbpp:park) WHERE ss.name='South Shore' AND cbpp.name='Camerons Brook Provincial Park' CREATE (cbpp)-[cbpp_ss:BELONGS_TO]->(ss);
 MATCH (ss:region),(cpp:park) WHERE ss.name='South Shore' AND cpp.name='Cookville Provincial Park' CREATE (cpp)-[cpp_ss:BELONGS_TO]->(ss);
 MATCH (ss:region),(bwbpp:park) WHERE ss.name='South Shore' AND bwbpp.name='Bayswater Beach Provincial Park' CREATE (bwbpp)-[bwbpp_ss:BELONGS_TO]->(ss);

```
MATCH (ss:region),(flpp:park) WHERE ss.name='South Shore' AND flpp.name='Fancy Lake
Provincial Park' CREATE (flpp)-[flpp_ss:BELONGS_TO]->(ss);
MATCH (ss:region),(clbpp:park) WHERE ss.name='South Shore' AND clbpp.name='Cleveland Beach
Provincial Park' CREATE (clbpp)-[clbpp_ss:BELONGS_TO]->(ss);
```

Relationships from parks in Yarmouth and Acadian Shores to region Yarmouth and Acadian Shores

Queries to CREATE [12] relationships from parks in Yarmouth and Acadian Shores to region Yarmouth and Acadian Shores are as below:

```
MATCH (yaas:region),(elpp:park) WHERE yaas.name='Yarmouth and Acadian Shores' AND
elpp.name='Ellenwood Lake Provincial Park' CREATE (elpp)-[elpp_yaas:BELONGS_TO]->(yaas);
MATCH (yaas:region),(gpp:park) WHERE yaas.name='Yarmouth and Acadian Shores' AND
gpp.name='Glenwood Provincial Park' CREATE (gpp)-[gpp_yaas:BELONGS_TO]->(yaas);
MATCH (yaas:region),(pmpp:park) WHERE yaas.name='Yarmouth and Acadian Shores' AND
pmpp.name='Port Maitland Provincial Park' CREATE (pmpp)-[pmpp_yaas:BELONGS_TO]->(yaas);
MATCH (yaas:region),(mbpp:park) WHERE yaas.name='Yarmouth and Acadian Shores' AND
mbpp.name='Mavillette Beach Provincial Park' CREATE (mbpp)-[mbpp_yaas:BELONGS_TO]-
>(yaas);
MATCH (yaas:region),(scpp:park) WHERE yaas.name='Yarmouth and Acadian Shores' AND
scpp.name='Smugglers Cove Provincial Park' CREATE (scpp)-[scpp_yaas:BELONGS_TO]->(yaas);
```

Relationships between Regions

Queries to CREATE [12] relationships between regions are as below:

```
MATCH (yaas:region),(fsaav:region) WHERE yaas.name='Yarmouth and Acadian Shores' AND
fsaav.name='Fundy Shore and Annapolis Valley' CREATE (yaas)-[yaas_fsaav:CONNECTED_TO]-
>(fsaav);
MATCH (yaas:region),(ss:region) WHERE yaas.name='Yarmouth and Acadian Shores' AND
ss.name='South Shore' CREATE (yaas)-[yaas_ss:CONNECTED_TO]->(ss);
MATCH (fsaav:region),(yaas:region) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND
yaas.name='Yarmouth and Acadian Shores' CREATE (fsaav)-[fsaav_yaas:CONNECTED_TO]->(yaas);
MATCH (fsaav:region),(ss:region) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND
ss.name='South Shore' CREATE (fsaav)-[fsaav_ss:CONNECTED_TO]->(ss);
MATCH (fsaav:region),(hr:region) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND
hr.name='Halifax Region' CREATE (fsaav)-[fsaav_hr:CONNECTED_TO]->(hr);
MATCH (fsaav:region),(es:region) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND
es.name='Eastern Shore' CREATE (fsaav)-[fsaav_es:CONNECTED_TO]->(es);
MATCH (fsaav:region),(nus:region) WHERE fsaav.name='Fundy Shore and Annapolis Valley' AND
nus.name='Northumberland Shore' CREATE (fsaav)-[fsaav_nus:CONNECTED_TO]->(nus);
MATCH (ss:region),(yaas:region) WHERE ss.name='South Shore' AND yaas.name='Yarmouth and
Acadian Shores' CREATE (ss)-[ss_yaas:CONNECTED_TO]->(yaas);
MATCH (ss:region),(fsaav:region) WHERE ss.name='South Shore' AND fsaav.name='Fundy Shore and
Annapolis Valley' CREATE (ss)-[ss_fsaav:CONNECTED_TO]->(fsaav);
MATCH (ss:region),(hr:region) WHERE ss.name='South Shore' AND hr.name='Halifax Region'
CREATE (ss)-[ss_hr:CONNECTED_TO]->(hr);
```

```

MATCH (hr:region),(ss:region) WHERE hr.name='Halifax Region' AND ss.name='South Shore'
CREATE (hr)-[hr_ss:CONNECTED_TO]->(ss);
MATCH (hr:region),(fsaav:region) WHERE hr.name='Halifax Region' AND fsaav.name='Fundy Shore
and Annapolis Valley' CREATE (hr)-[hr_fsaav:CONNECTED_TO]->(fsaav);
MATCH (hr:region),(es:region) WHERE hr.name='Halifax Region' AND es.name='Eastern Shore'
CREATE (hr)-[hr_es:CONNECTED_TO]->(es);
MATCH (es:region),(hr:region) WHERE es.name='Eastern Shore' AND hr.name='Halifax Region'
CREATE (es)-[es_hr:CONNECTED_TO]->(hr);
MATCH (es:region),(fsaav:region) WHERE es.name='Eastern Shore' AND fsaav.name='Fundy Shore
and Annapolis Valley' CREATE (es)-[es_fsaav:CONNECTED_TO]->(fsaav);
MATCH (es:region),(nus:region) WHERE es.name='Eastern Shore' AND nus.name='Northumberland
Shore' CREATE (es)-[es_nus:CONNECTED_TO]->(nus);
MATCH (es:region),(cbi:region) WHERE es.name='Eastern Shore' AND cbi.name='Cape Breton Island'
CREATE (es)-[es_cbi:CONNECTED_TO]->(cbi);
MATCH (nus:region),(es:region) WHERE nus.name='Northumberland Shore' AND es.name='Eastern
Shore' CREATE (nus)-[nus_es:CONNECTED_TO]->(es);
MATCH (nus:region),(fsaav:region) WHERE nus.name='Northumberland Shore' AND
fsaav.name='Fundy Shore and Annapolis Valley' CREATE (nus)-[nus_fsaav:CONNECTED_TO]-
>(fsaav);
MATCH (nus:region),(cbi:region) WHERE nus.name='Northumberland Shore' AND cbi.name='Cape
Breton Island' CREATE (nus)-[nus_cbi:CONNECTED_TO]->(cbi);
MATCH (cbi:region),(nus:region) WHERE cbi.name='Cape Breton Island' AND
nus.name='Northumberland Shore' CREATE (cbi)-[cbi_nus:CONNECTED_TO]->(nus);
MATCH (cbi:region),(es:region) WHERE cbi.name='Cape Breton Island' AND es.name='Eastern Shore'
CREATE (cbi)-[cbi_es:CONNECTED_TO]->(es);

```

⇒ **Figure 15** displays the generated graph containing regions and parks in Nova Scotia province.

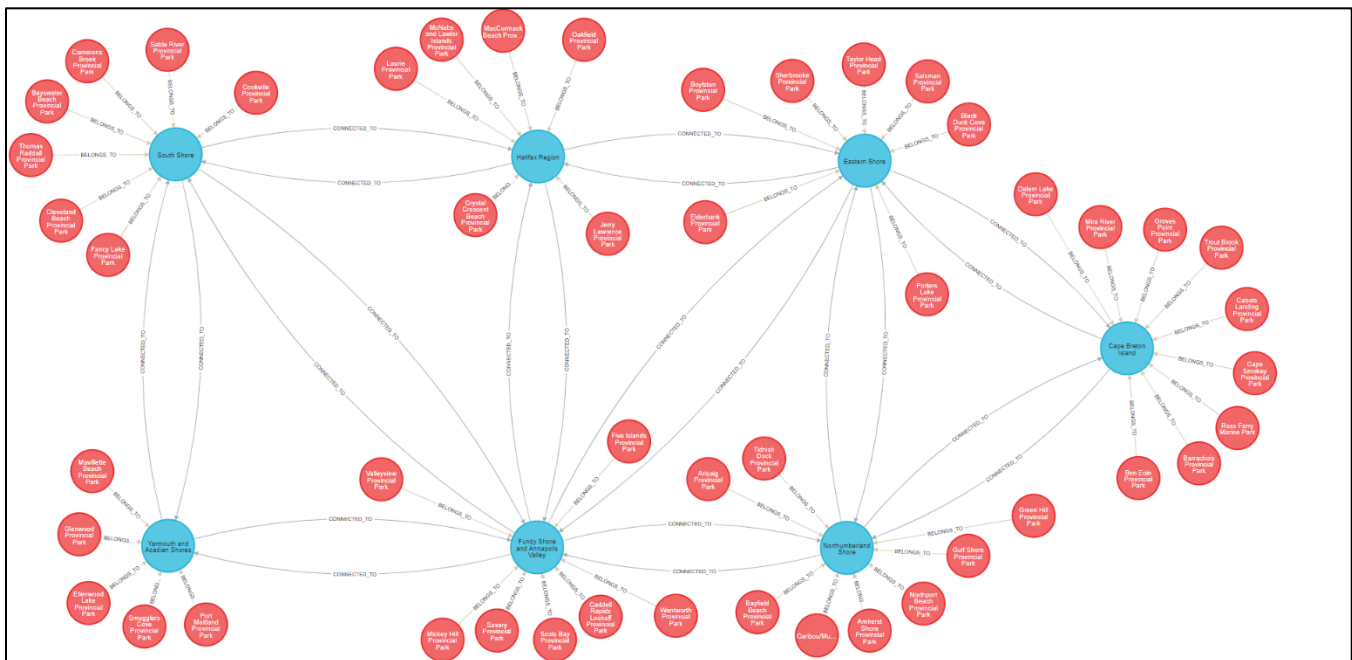


Figure 15 - Nova Scotia regions and parks

⇒ **Figure 16** displays information related to regions and the number of parks in these regions.

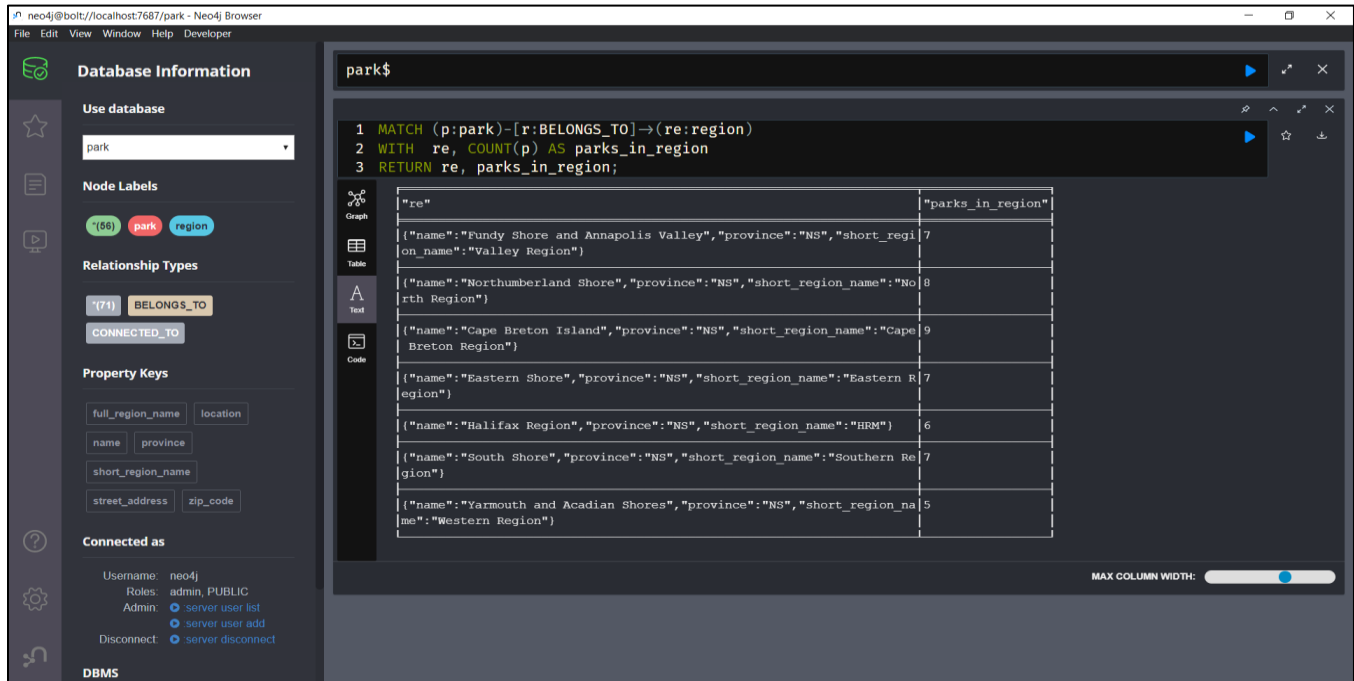


Figure 16 - Regions and number of parks in these regions

⇒ **Figure 17** displays the region “Cape Breton Island” with the highest number of parks (Total parks – 9).

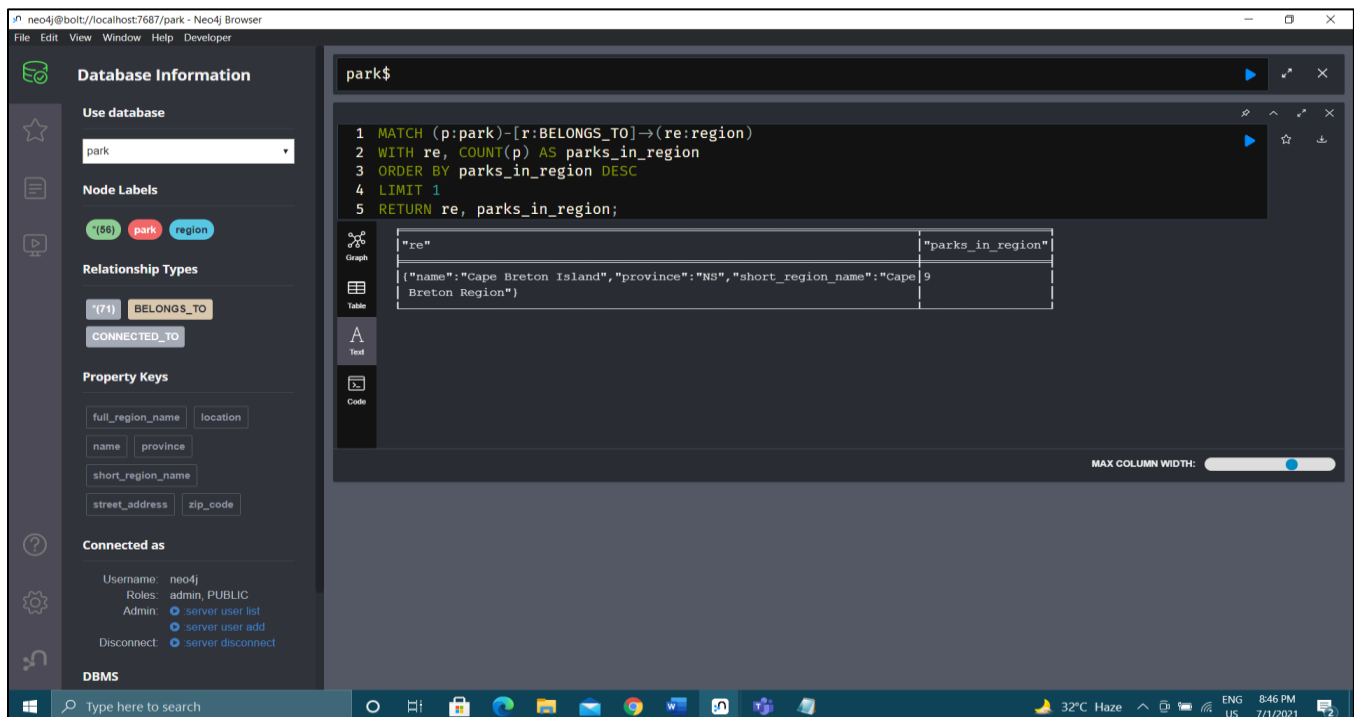


Figure 17 - Cape Breton Island with 9 parks

REFERENCES

- [1] Google, "Google Cloud Platform," Google, [Online]. Available: <https://console.cloud.google.com/>. [Accessed 01 July 2021].
- [2] D. Tucakov, "How to Install Spark on Ubuntu," phoenixNAP, [Online]. Available: <https://phoenixnap.com/kb/install-spark-on-ubuntu>. [Accessed 01 July 2021].
- [3] A. Spark, "Download Apache Spark™," Apache Spark, [Online]. Available: <https://spark.apache.org/downloads.html>. [Accessed 01 July 2021].
- [4] NewsAPI, "Search worldwide news with code," NewsAPI, [Online]. Available: <https://newsapi.org/>. [Accessed 02 July 2021].
- [5] GitLab, "Projects," GitLab, [Online]. Available: <https://git.cs.dal.ca/>. [Accessed 05 July 2021].
- [6] JSON-Java, "JSON In Java » 20210307," JSON-Java, [Online]. Available: <https://mvnrepository.com/artifact/org.json/json/20210307>. [Accessed 02 July 2021].
- [7] S. A. Metwalli, "Pseudocode 101: An Introduction to Writing Good Pseudocode," towards data science, [Online]. Available: <https://towardsdatascience.com/pseudocode-101-an-introduction-to-writing-good-pseudocode-1331cb855be7>. [Accessed 04 July 2021].
- [8] MongoDB, "MongoDB Atlas," MongoDB, Inc., [Online]. Available: <https://www.mongodb.com/cloud/atlas>. [Accessed 03 July 2021].
- [9] MongoDB, "MongoDB Java Driver >> 3.12.8," MongoDB, [Online]. Available: <https://mvnrepository.com/artifact/org.mongodb/mongo-java-driver/3.12.8>. [Accessed 03 July 2021].
- [10] Diagrams.net, "Diagram Software and Flowchart Maker," [Online]. Available: <https://app.diagrams.net/>. [Accessed 04 July 2021].
- [11] P. o. N. Scotia, "Find a Park," NovaScotia Provincial Parks, [Online]. Available: <https://parks.novascotia.ca/region/nova-scotia>. [Accessed 03 July 2021].
- [12] neo4j, "CREATE," Neo4j, Inc., [Online]. Available: <https://neo4j.com/docs/cypher-manual/current/clauses/create/>. [Accessed 03 July 2021].