ASSIGNMENT – 4
DATA WAREHOUSING AND MANAGEMENT

SUBMITTED TO: PROF SAURABH DEY
SUBMITTED BY: DHRUV DOSHI
MAIL: dh722257@dal.ca

GITLAB: https://git.cs.dal.ca/doshi/csci-5408-s2021-b00883311-dhruv-doshi/-/tree/master/A4/src

PART – A
Business Intelligence Reporting using Cognos

Using dataset, https://www.kaggle.com/PROPPG-PPG/hourly-weather-surface-brazil-southeast-region?select=sudeste.csv

Upon analyzing the whole dataset I found that out, it was like 1.8 Gigs and there were huge amount of data. Firstly my computer was not in the condition to handle the data hence I used the VM to put the data and then with ssh and Libre office Calc – Open source alternative for MS Excel I purified the data. Now while looking through the IBM Cognos I found that the maximum file size supported for my account was 100Mb hence I make every file under that limit.

What I did mainly was just removing the zeros and null values and then I formatted the date properly. Along with that there were abundance of duplicate hence I removed some of the duplicate information accordingly.

After completing this I divided the whole data set into 9 different parts -

- Dew_points
- Humidity_data
- Precipitation_data
- location_info
- Pressure_data
- radiation_info
- station_data
- temperature_info
- time
- wind

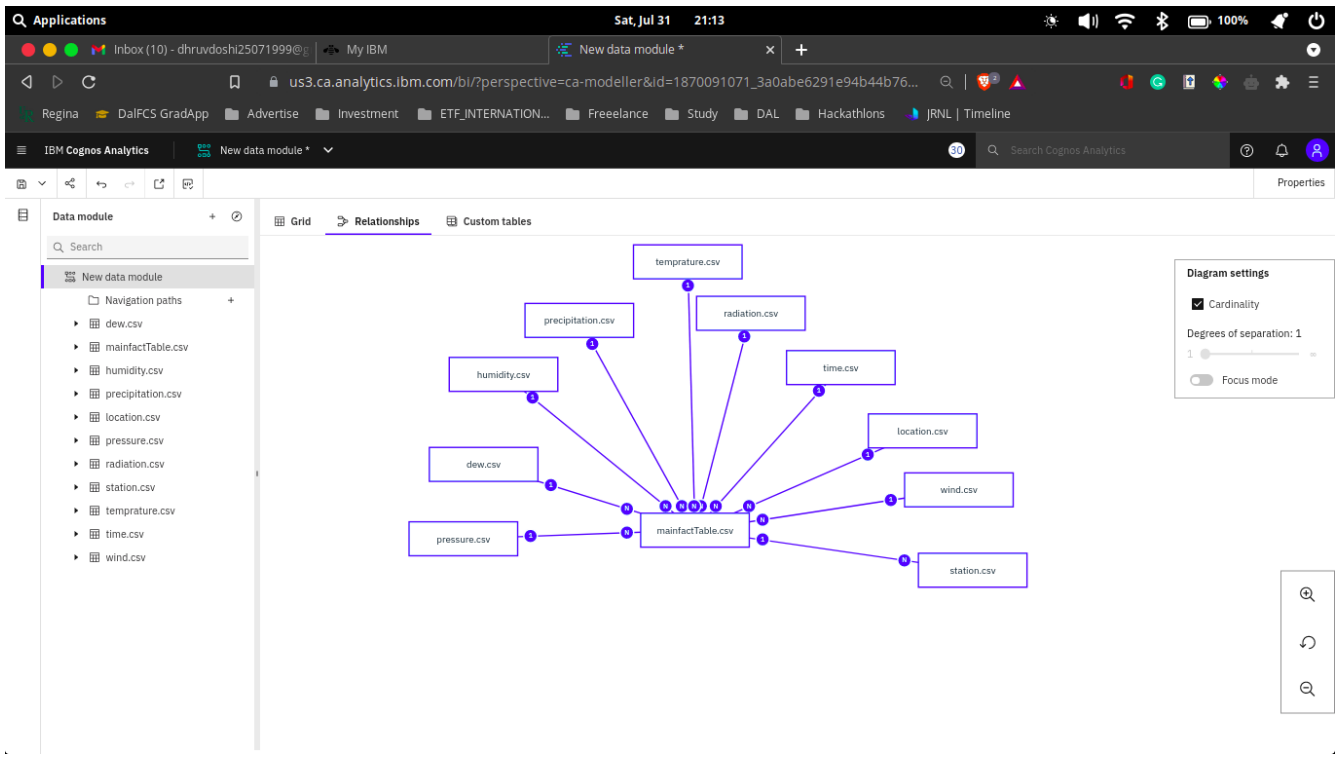
IBM COGNOS QUERY:

Show query information

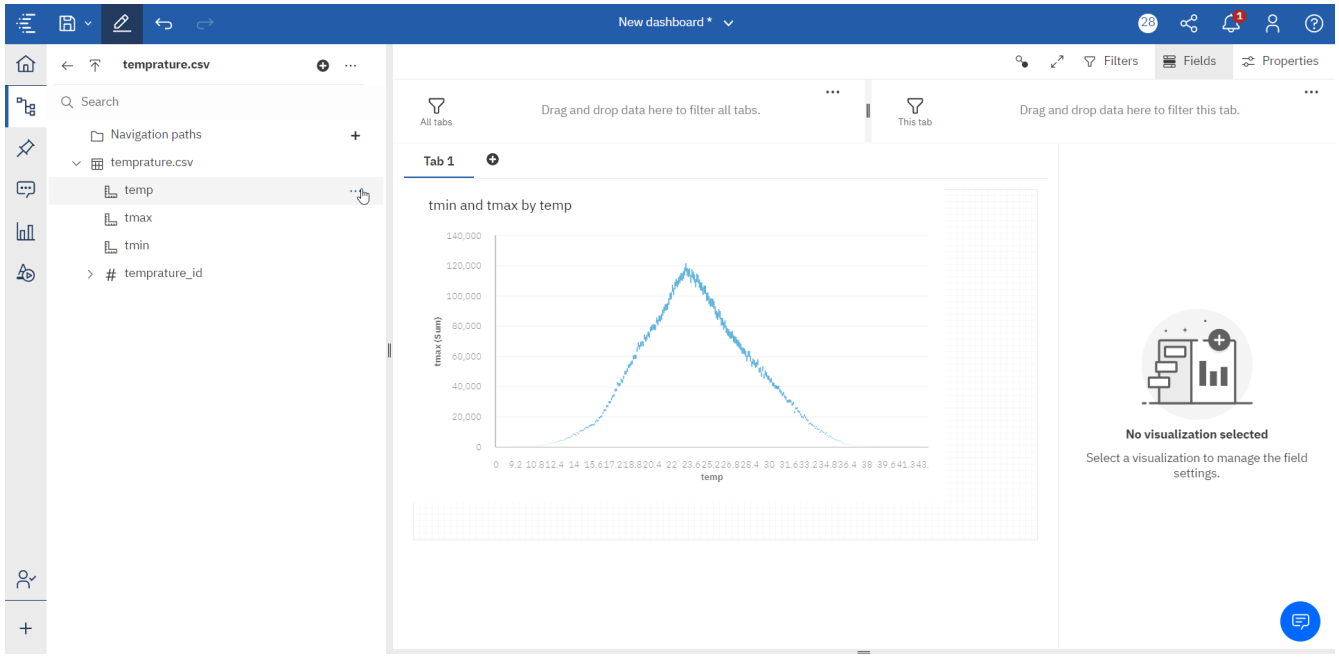
mainfactTable.csv

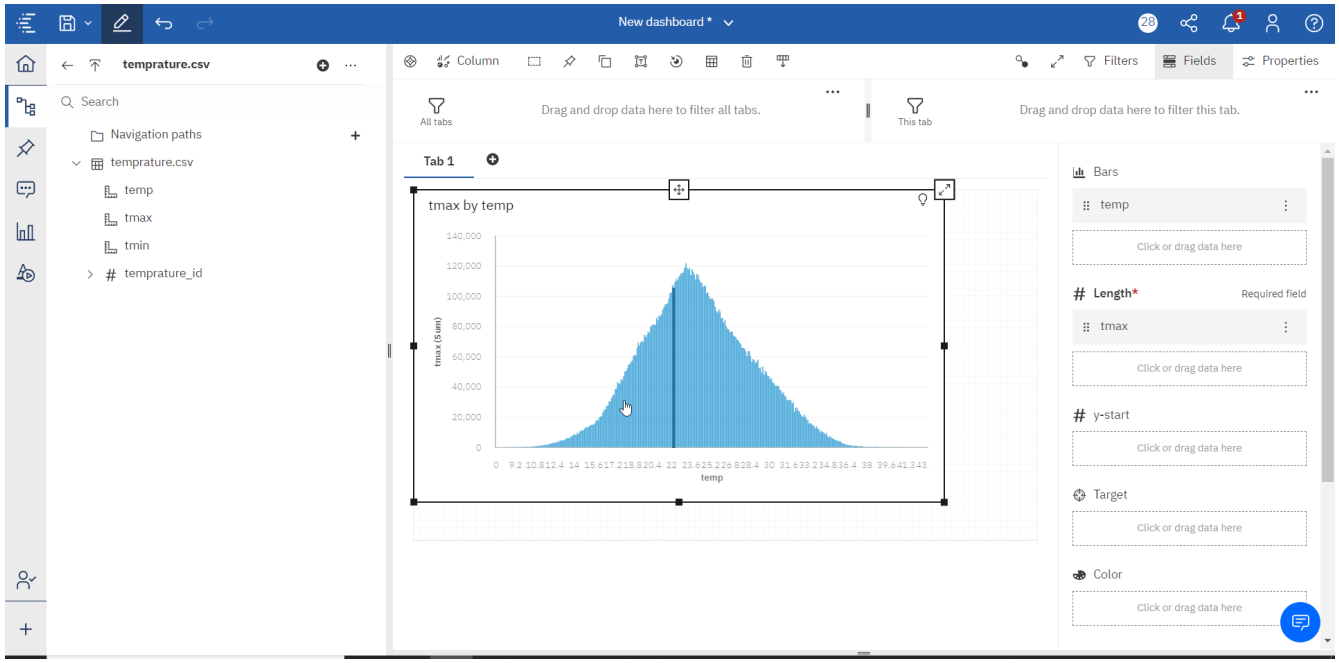
Query information type: Cognos SQL

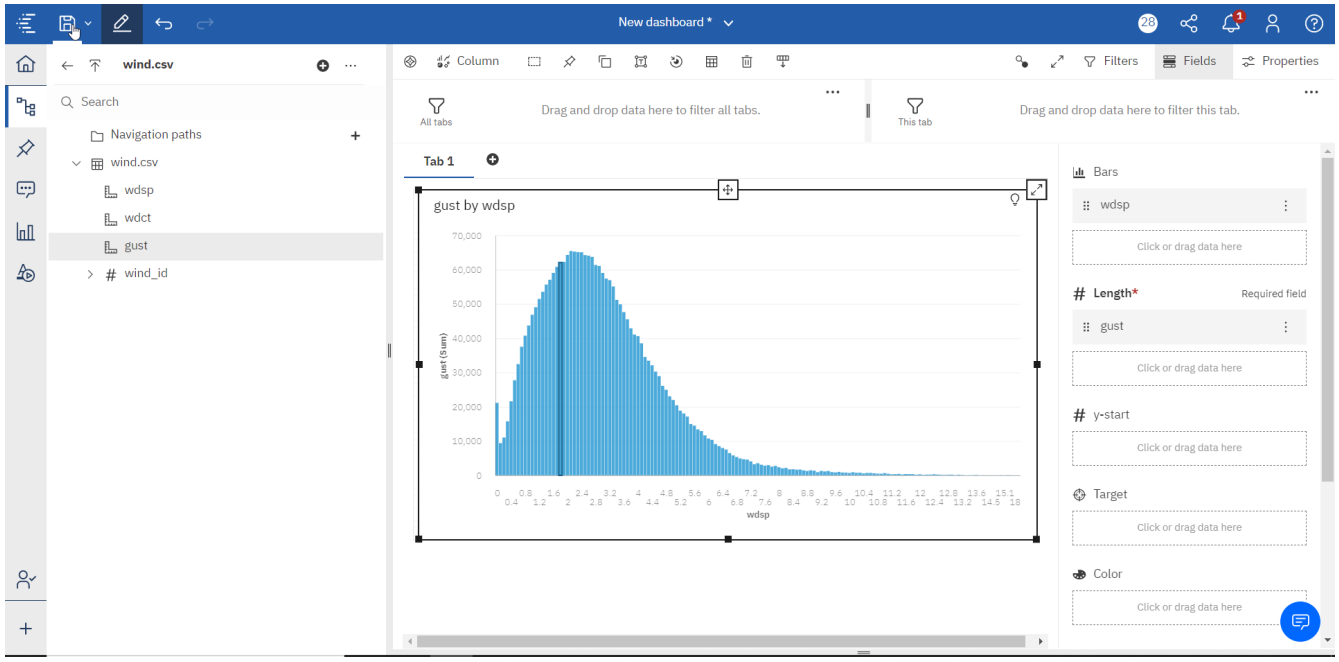
```sql
SELECT
    factTable_csv0._row_id AS C0,
    factTable_csv0.wind_id AS C1,
    factTable_csv0.dew_id AS C2,
    factTable_csv0.humidity_id AS C3,
    factTable_csv0.location_id AS C4,
    factTable_csv0.precipitation_id AS C5,
    factTable_csv0.pressure_id AS C6,
    factTable_csv0.radiation_id AS C7,
    factTable_csv0.station_id AS C8,
    factTable_csv0.temprature_id AS C9,
    factTable_csv0.time_id AS C10,
    factTable_csv0.wind_id_1 AS C11
FROM
    2765206185...factTable_csv factTable_csv0
```



STAR SCHEMA


New dashboard *
temprature.csv
Search
Navigation paths
temprature.csv
temp
tmax
tmin
temprature_id
Filters
Fields
Properties
All tabs
Drag and drop data here to filter all tabs.
This tab
Drag and drop data here to filter this tab.
Tab 1
tmin and tmax by temp
tmax (Sum)
temp
No visualization selected
Select a visualization to manage the field settings.


New dashboard *
temprature.csv
Search
Navigation paths
temprature.csv
temp
tmax
tmin
temprature_id
Column
Filters
Fields
Properties
All tabs
Drag and drop data here to filter all tabs.
This tab
Drag and drop data here to filter this tab.
Tab 1
tmax by temp
tmax (Sum)
temp
Bars
temp
Click or drag data here
Length*
Required field
tmax
Click or drag data here
y-start
Click or drag data here
Target
Click or drag data here
Color
Click or drag data here

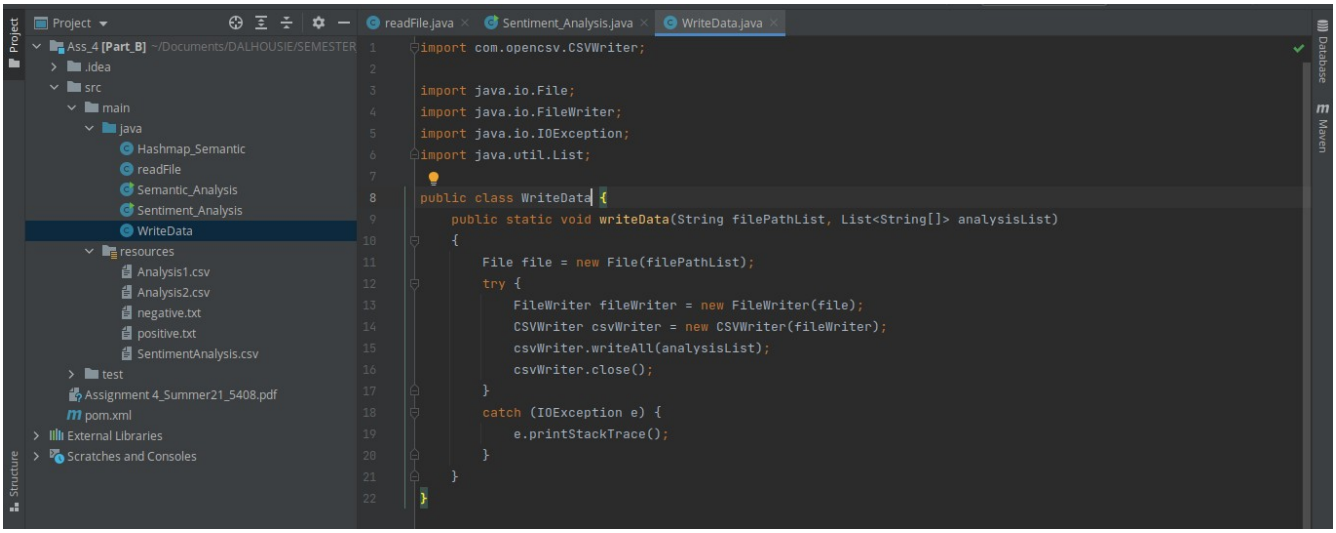

New dashboard *
wind.csv
Search
Navigation paths
wind.csv
wdsp
wdct
gust
wind_id
Column
Filters
Fields
Properties
All tabs
Drag and drop data here to filter all tabs.
This tab
Drag and drop data here to filter this tab.
Tab 1
gust by wdsp
gust (Sum)
wdsp
Bars
wdsp
Click or drag data here
Length*
Required field
gust
Click or drag data here
y-start
Click or drag data here
Target
Click or drag data here
Color
Click or drag data here

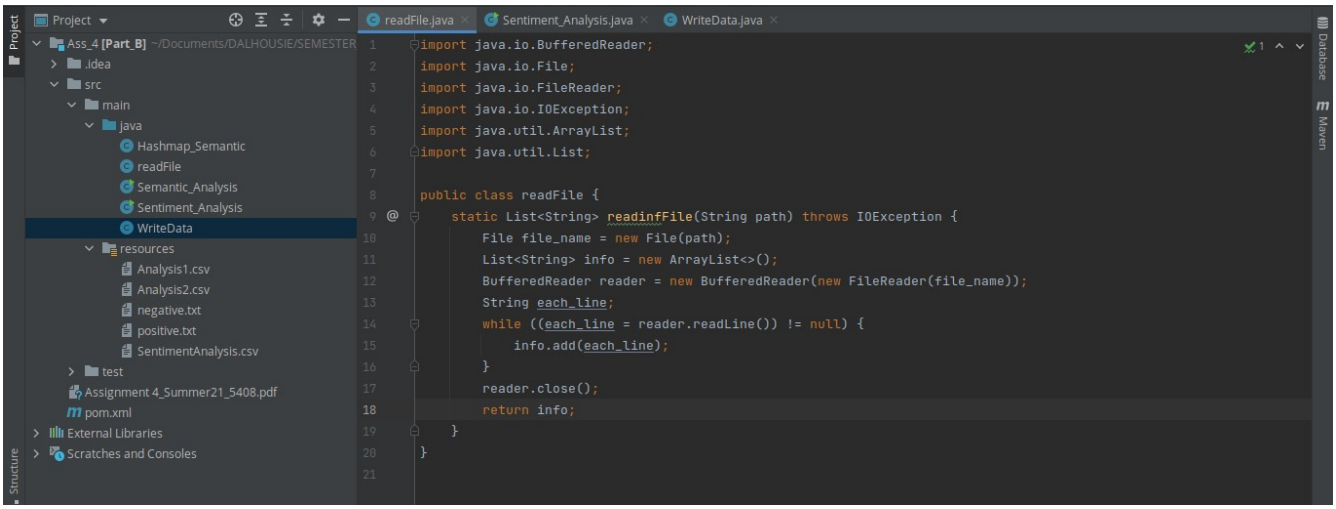
CODE:

```
SELECT
    factTable_csv0._row_id AS C0,
    factTable_csv0.wind_id AS C1,
    factTable_csv0.dew_id AS C2,
    factTable_csv0.humidity_id AS C3,
    factTable_csv0.location_id AS C4,
    factTable_csv0.precipitation_id AS C5,
    factTable_csv0.pressure_id AS C6,
    factTable_csv0.radiation_id AS C7,
    factTable_csv0.station_id AS C8,
    factTable_csv0.temprature_id AS C9,
    factTable_csv0.time_id AS C10,
    factTable_csv0.wind_id_1 AS C11
FROM
    2765206185...factTable_csv factTable_csv0
```

PART-B

SENTIMENT ANALYSIS





Applications Sat, Jul 31 23:33 100%

Part_B – Sentiment_Analysis.java

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Ass_4 〉 src 〉 main 〉 java 〉 Sentiment_Analysis 〉 main

Add Configuration...

readFile.java | Sentiment_Analysis.java | WriteData.java

```java
import java.util.Map;

public class Sentiment_Analysis {
    static List<String> negativeList = new ArrayList<>();
    static List<String> positiveList = new ArrayList<>();
    static List<String[]> analysisList = new ArrayList<>();

    static int count = 1;
    static WriteData import_func_write_data;
    static readFile import_func_read_file;

    public static void main(String[] args) throws IOException {

        String positivePath = "src/main/resources/positive.txt";
        String negativePath = "src/main/resources/negative.txt";
        positiveList = import_func_read_file.readinfFile(positivePath);
        negativeList = import_func_read_file.readinfFile(negativePath);

        // Conencting with MONGODB
        MongoClient client = MongoClients.create("mongodb://localhost");
        MongoDatabase database = client.getDatabase( s: "CSCI5408");
        MongoCollection<Document> mongocollection = database.getCollection( s: "mongonews");
        FindIterable<Document> iterable_Items = mongocollection.find();

        analysisList.add(new String[]{"Article", "News Description", "Match", "Polarity"});

        for (Document document_loop : iterable_Items) {
```

Project: Ass_4 [Part_B] ~/Documents/DALHOUSIE/SEMESTER
.idea, src, main, java, Hashmap_Semantic, readFile, Semantic_Analysis, Sentiment_Analysis, WriteData, resources, Analysis1.csv, Analysis2.csv, negative.txt, positive.txt, SentimentAnalysis.csv, test, Assignment 4_Summer21_5408.pdf, pom.xml, External Libraries, Scratches and Consoles

TODO Problems Terminal Profiler Event Log

New JDK version is available: Oracle OpenJDK 16.0.2 // Current version is 16.0.1 (JDK "JavaSE-1.8") // Download Skip this update (a minute ago) 20:1 LF UTF-8 4 spaces

Applications Sat, Jul 31 23:34 100%

Part_B – Sentiment_Analysis.java

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Ass_4 〉 src 〉 main 〉 java 〉 Sentiment_Analysis 〉 main

Add Configuration...

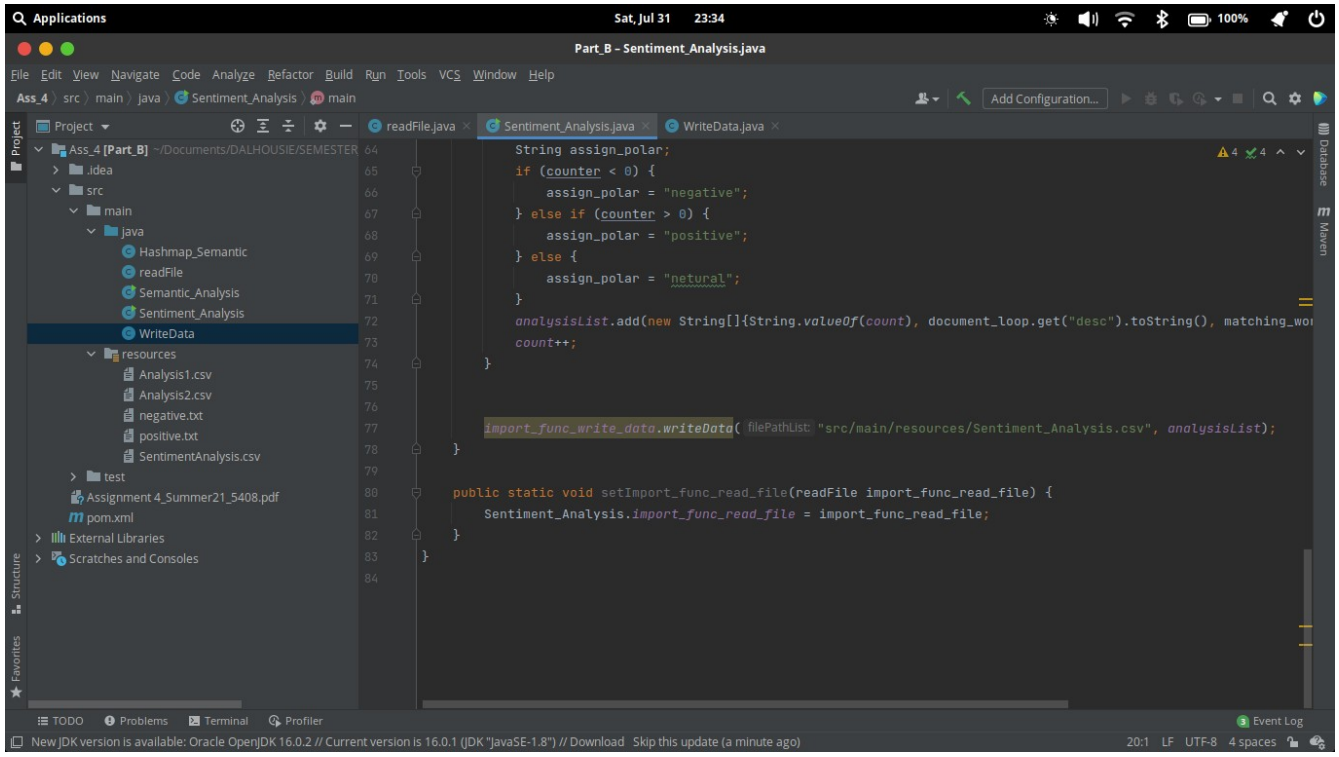
readFile.java | Sentiment_Analysis.java | WriteData.java

```java
        analysisList.add(new String[]{"Article", "News Description", "Match", "Polarity"});

        for (Document document_loop : iterable_Items) {
            int counter = 0;
            String[] list_of_words = document_loop.get("desc").toString().split( regex: " ");
            Map<String, Integer> bag_of_words = new HashMap<>();
            List<String> matching_word_list = new ArrayList<>();

            for (String word_loop : list_of_words) {
                if (!bag_of_words.containsKey(word_loop)) {
                    bag_of_words.put(word_loop, 1);
                } else {
                    int count = bag_of_words.get(word_loop);
                    bag_of_words.put(word_loop, count + 1);
                }
            }

            for (String key : bag_of_words.keySet()) {
                if (positiveList.contains(key)) {
                    if (!matching_word_list.contains(key)) {
                        matching_word_list.add(key);
                    }
                    counter = counter + bag_of_words.get(key);
                }
                if (negativeList.contains(key)) {
                    if (!matching_word_list.contains(key)) {
```

Project: Ass_4 [Part_B] ~/Documents/DALHOUSIE/SEMESTER
.idea, src, main, java, Hashmap_Semantic, readFile, Semantic_Analysis, Sentiment_Analysis, WriteData, resources, Analysis1.csv, Analysis2.csv, negative.txt, positive.txt, SentimentAnalysis.csv, test, Assignment 4_Summer21_5408.pdf, pom.xml, External Libraries, Scratches and Consoles

TODO Problems Terminal Profiler Event Log

New JDK version is available: Oracle OpenJDK 16.0.2 // Current version is 16.0.1 (JDK "JavaSE-1.8") // Download Skip this update (a minute ago) 20:1 LF UTF-8 4 spaces



CODE

```
import com.mongodb.client.*;
import org.bson.Document;

import java.io.*;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class Sentiment_Analysis {
    static List<String> negativeList = new ArrayList<>();
    static List<String> positiveList = new ArrayList<>();
    static List<String[]> analysisList = new ArrayList<>();

    static int count = 1;
    static WriteData import_func_write_data;
    static readFile import_func_read_file;

    public static void main(String[] args) throws IOException {

        String positivePath = "src/main/resources/positive.txt";
        String negativePath = "src/main/resources/negative.txt";
        positiveList = import_func_read_file.readinfFile(positivePath);
        negativeList = import_func_read_file.readinfFile(negativePath);

        // Conencting with MONGODB
        MongoClient client = MongoClients.create("mongodb://localhost");
```

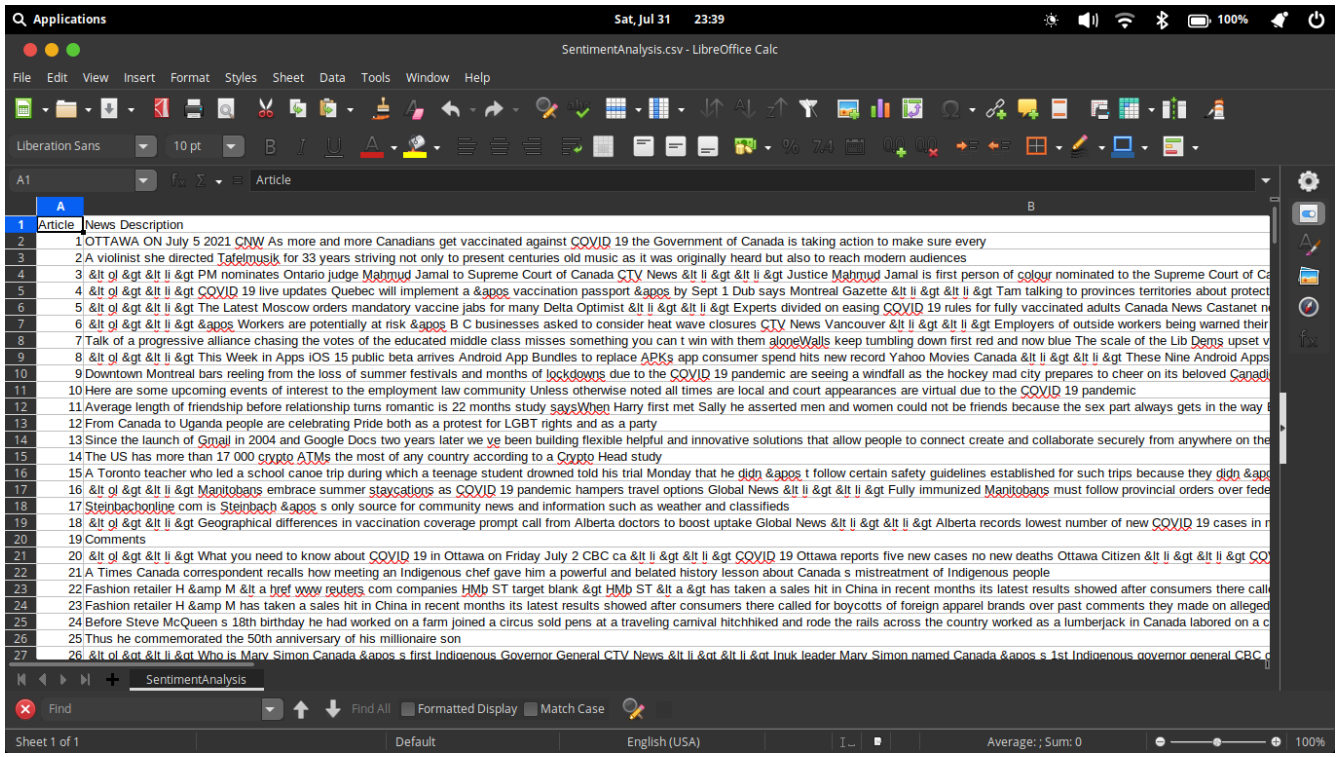
```
        MongoDatabase database = client.getDatabase("CSCI5408");
        MongoCollection<Document> mongocollection = database.getCollection("mongonews");
        FindIterable<Document> iterable_Items = mongocollection.find();

        analysisList.add(new String[]{"Article", "News Description", "Match", "Polarity"});

        for (Document document_loop : iterable_Items) {
            int counter = 0;
            String[] list_of_words = document_loop.get("desc").toString().split(" ");
            Map<String, Integer> bag_of_words = new HashMap<>();
            List<String> matching_word_list = new ArrayList<>();

            for (String word_loop : list_of_words) {
                if (!bag_of_words.containsKey(word_loop)) {
                    bag_of_words.put(word_loop, 1);
                } else {
                    int count = bag_of_words.get(word_loop);
                    bag_of_words.put(word_loop, count + 1);
                }
            }

            for (String key : bag_of_words.keySet()) {
                if (positiveList.contains(key)) {
                    if (!matching_word_list.contains(key)) {
                        matching_word_list.add(key);
                    }
                    counter = counter + bag_of_words.get(key);
                }
                if (negativeList.contains(key)) {
                    if (!matching_word_list.contains(key)) {
                        matching_word_list.add(key);
                    }
                    counter = counter - bag_of_words.get(key);
                }
            }

            String assign_polar;
            if (counter < 0) {
                assign_polar = "negative";
            } else if (counter > 0) {
                assign_polar = "positive";
            } else {
                assign_polar = "netural";
            }
            analysisList.add(new String[]{String.valueOf(count), document_loop.get("desc").toString(),
matching_word_list.toString(), assign_polar});
            count++;
        }


        import_func_write_data.writeData("src/main/resources/Sentiment_Analysis.csv", analysisList);
    }

    public static void setImport_func_read_file(readFile import_func_read_file) {
        Sentiment_Analysis.import_func_read_file = import_func_read_file;
    }
}
```

READFILE FUNCTION

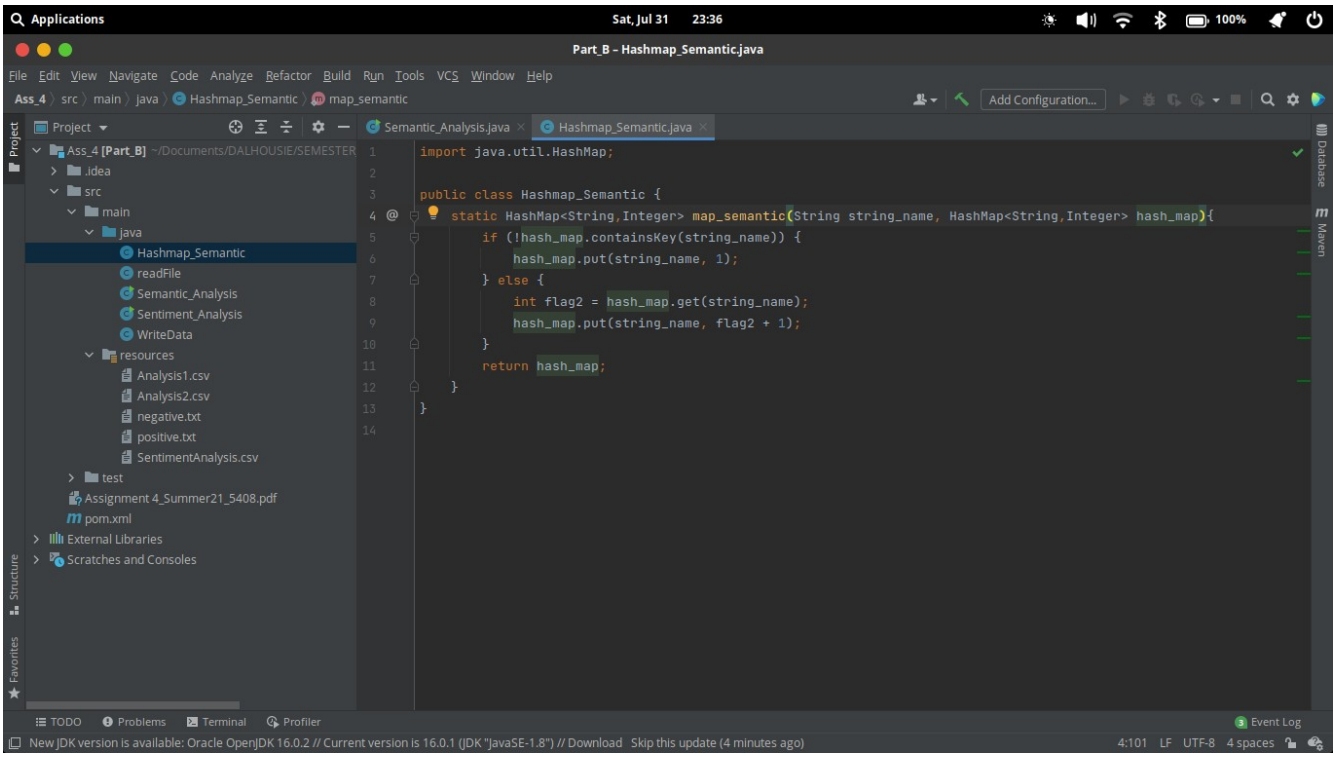
```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;

public class readFile {
    static List<String> readinfFile(String path) throws IOException {
        File file_name = new File(path);
        List<String> info = new ArrayList<>();
        BufferedReader reader = new BufferedReader(new FileReader(file_name));
        String each_line;
        while ((each_line = reader.readLine()) != null) {
            info.add(each_line);
        }
        reader.close();
        return info;
    }
}
```

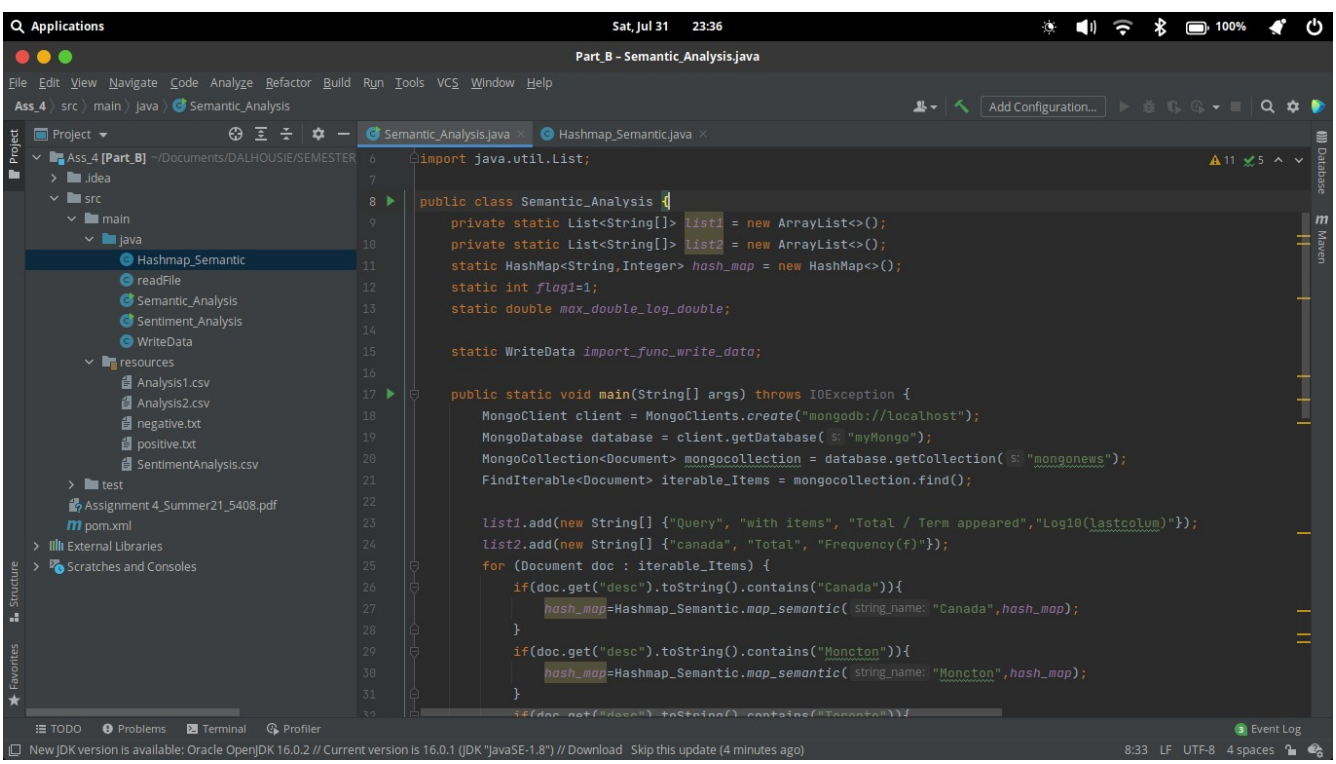
WRITE DATA FUNCTION

```
import com.opencsv.CSVWriter;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.util.List;

public class WriteData {
   public static void writeData(String filePathList, List<String[]> analysisList)
   {
      File file = new File(filePathList);
      try {
         FileWriter fileWriter = new FileWriter(file);
         CSVWriter csvWriter = new CSVWriter(fileWriter);
         csvWriter.writeAll(analysisList);
         csvWriter.close();
      }
      catch (IOException e) {
         e.printStackTrace();
      }
   }
}
```



PART – C
SEMANTIC ANALYSIS





Applications Sat, Jul 31 23:37 100%

Part_B – Semantic_Analysis.java

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Ass_4 〉 src 〉 main 〉 java 〉 Semantic_Analysis

Add Configuration...

Project

Ass_4 [Part_B] ~/Documents/DALHOUSIE/SEMESTER
- .idea
- src
  - main
    - java
      - Hashmap_Semantic
      - readFile
      - Semantic_Analysis
      - Sentiment_Analysis
      - WriteData
    - resources
      - Analysis1.csv
      - Analysis2.csv
      - negative.txt
      - positive.txt
      - SentimentAnalysis.csv
  - test
- Assignment 4_Summer21_5408.pdf
- pom.xml
- External Libraries
- Scratches and Consoles

Semantic_Analysis.java | Hashmap_Semantic.java

```
                hash_map=Hashmap_Semantic.map_semantic( string_name: "Toronto",hash_map);
            }
            String[] wordList = doc.get("desc").toString().split( regex: " ");

            int flag2=0;
            for (String word : wordList) {
                if (word.equals("Canada")) {
                    flag2++;
                }
            }
            if(flag2>0){
                list2.add(new String[]{String.valueOf(flag1),String.valueOf(wordList.length),String.valueOf(flag2)
                flag1++;
                String str=String.format("%.5f",(double)flag2/wordList.length);
                double freq= Double.valueOf(str);
                if(flag1==1){
                    max_double_log_double=freq;
                }
                if(max_double_log_double<freq){
                    max_double_log_double=freq;
                }
            }
        }
        for (String key : hash_map.keySet() ) {
            list1.add(new String[]{key,String.valueOf(hash_map.get(key)),"500/"+String.valueOf(hash_map.get(key)),
        }
```

TODO Problems Terminal Profiler Event Log

New JDK version is available: Oracle OpenJDK 16.0.2 // Current version is 16.0.1 (JDK "JavaSE-1.8") // Download Skip this update (4 minutes ago) 8:33 LF UTF-8 4 spaces

Applications Sat, Jul 31 23:37 100%

Part_B – Semantic_Analysis.java

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Ass_4 〉 src 〉 main 〉 java 〉 Semantic_Analysis

Add Configuration...

Project

Ass_4 [Part_B] ~/Documents/DALHOUSIE/SEMESTER
- .idea
- src
  - main
    - java
      - Hashmap_Semantic
      - readFile
      - Semantic_Analysis
      - Sentiment_Analysis
      - WriteData
    - resources
      - Analysis1.csv
      - Analysis2.csv
      - negative.txt
      - positive.txt
      - SentimentAnalysis.csv
  - test
- Assignment 4_Summer21_5408.pdf
- pom.xml
- External Libraries
- Scratches and Consoles

Semantic_Analysis.java | Hashmap_Semantic.java

```
                list2.add(new String[]{String.valueOf(flag1),String.valueOf(wordList.length),String.
                flag1++;
                String str=String.format("%.5f",(double)flag2/wordList.length);
                double freq= Double.valueOf(str);
                if(flag1==1){
                    max_double_log_double=freq;
                }
                if(max_double_log_double<freq){
                    max_double_log_double=freq;
                }
            }
        }
        for (String key : hash_map.keySet() ) {
            list1.add(new String[]{key,String.valueOf(hash_map.get(key)),"500/"+String.valueOf(hash_map.get(key)),
        }
        System.out.println("highest frequency "+max_double_log_double);
        import_func_write_data.writeData( filePathList: "src/main/resources/Analysis1.csv",list1);
        import_func_write_data.writeData( filePathList: "src/main/resources/Analysis2.csv",list2);

    }
}
```

TODO Problems Terminal Profiler Event Log

New JDK version is available: Oracle OpenJDK 16.0.2 // Current version is 16.0.1 (JDK "JavaSE-1.8") // Download Skip this update (5 minutes ago) 8:33 LF UTF-8 4 spaces

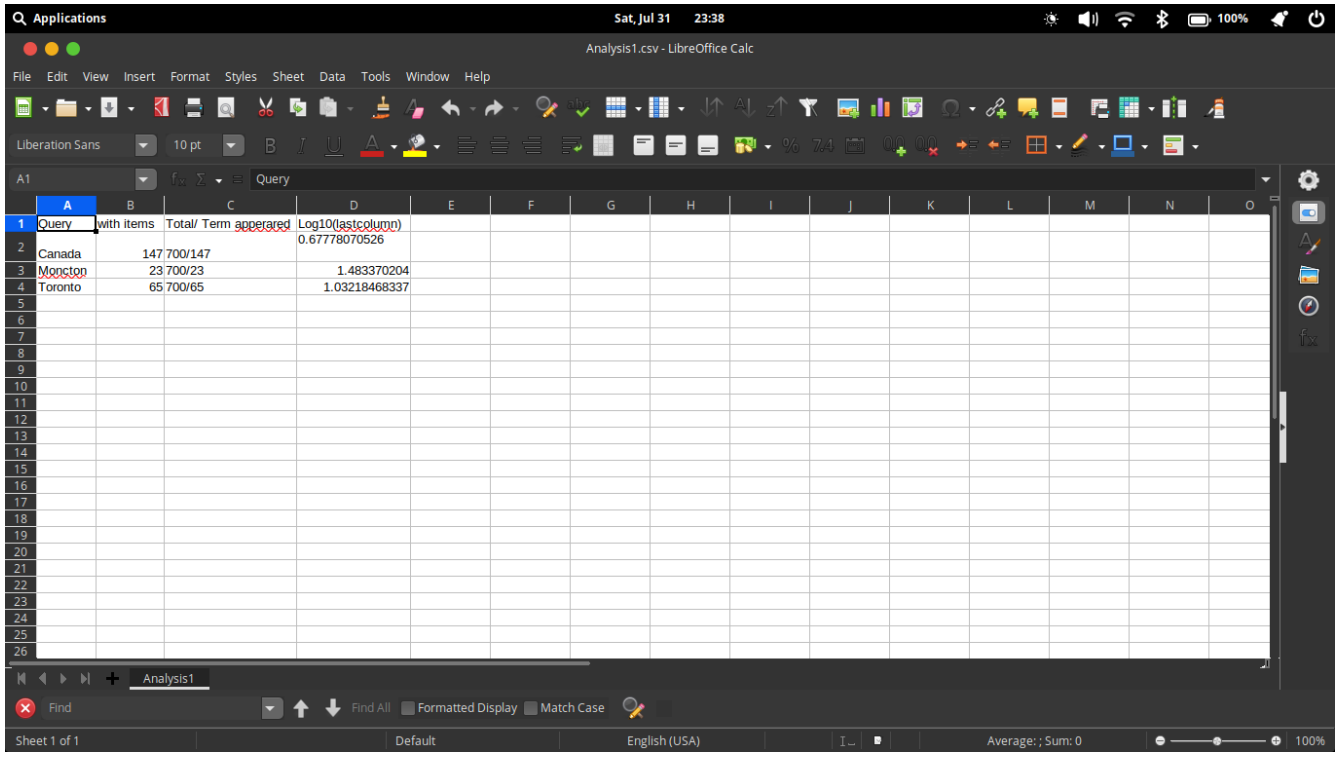
CODE FILES

```
import com.mongodb.client.*;
import org.bson.Document;
import java.io.IOException;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;

public class Semantic_Analysis {
    private static List<String[]> list1 = new ArrayList<>();
    private static List<String[]> list2 = new ArrayList<>();
    static HashMap<String,Integer> hash_map = new HashMap<>();
    static int flag1=1;
    static double max_double_log_double;

    static WriteData import_func_write_data;

    public static void main(String[] args) throws IOException {
        MongoClient client = MongoClients.create("mongodb://localhost");
        MongoDatabase database = client.getDatabase("myMongo");
        MongoCollection<Document> mongocollection = database.getCollection("mongonews");
        FindIterable<Document> iterable_Items = mongocollection.find();

        list1.add(new String[] {"Query", "with items", "Total / Term appeared","Log10(lastcolum)"});
        list2.add(new String[] {"canada", "Total", "Frequency(f)"});
        for (Document doc : iterable_Items) {
            if(doc.get("desc").toString().contains("Canada")){
                hash_map=Hashmap_Semantic.map_semantic("Canada",hash_map);
            }
            if(doc.get("desc").toString().contains("Moncton")){
                hash_map=Hashmap_Semantic.map_semantic("Moncton",hash_map);
            }
            if(doc.get("desc").toString().contains("Toronto")){
                hash_map=Hashmap_Semantic.map_semantic("Toronto",hash_map);
            }
            String[] wordList = doc.get("desc").toString().split(" ");

            int flag2=0;
            for (String word : wordList) {
                if (word.equals("Canada")) {
                    flag2++;
                }
            }
            if(flag2>0){
                list2.add(new String[]{String.valueOf(flag1),String.valueOf(wordList.length),String.valueOf(flag2)});
                flag1++;
                String str=String.format("%.5f",(double)flag2/wordList.length);
                double freq= Double.valueOf(str);
                if(flag1==1){
                    max_double_log_double=freq;
                }
                if(max_double_log_double<freq){
                    max_double_log_double=freq;
                }
            }
```

```
        }
        for (String key : hash_map.keySet() ) {
            list1.add(new String[]
{key,String.valueOf(hash_map.get(key)),"500/"+String.valueOf(hash_map.get(key)),String.valueOf(Math.log10(500/
hash_map.get(key)))});
        }
        System.out.println("highest frequency "+max_double_log_double);
        import_func_write_data.writeData("src/main/resources/Analysis1.csv",list1);
        import_func_write_data.writeData("src/main/resources/Analysis2.csv",list2);


    }
}
```

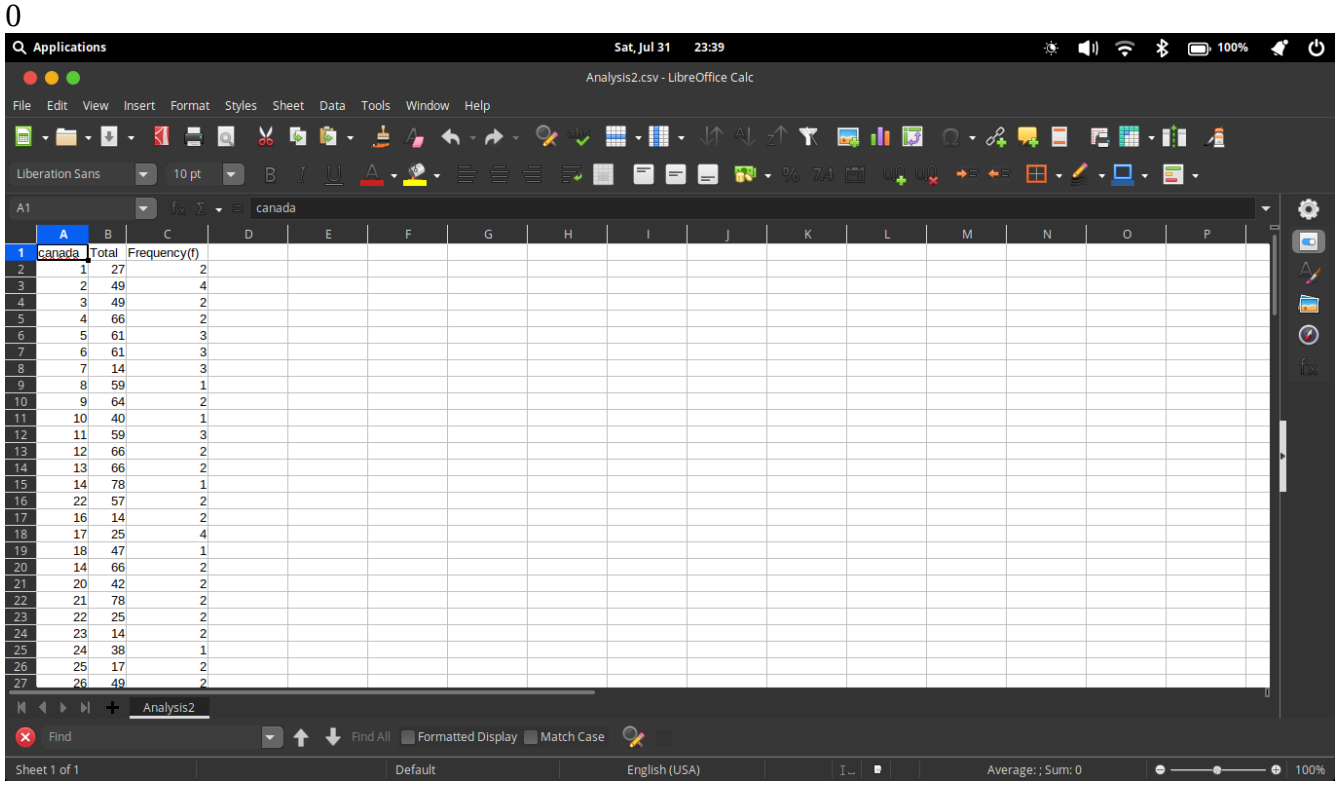
HASHMAP SEMANTIC

```
import java.util.HashMap;

public class Hashmap_Semantic {
    static HashMap<String,Integer> map_semantic(String string_name, HashMap<String,Integer> hash_map){
        if (!hash_map.containsKey(string_name)) {
            hash_map.put(string_name, 1);
        } else {
            int flag2 = hash_map.get(string_name);
            hash_map.put(string_name, flag2 + 1);
        }
        return hash_map;
    }
}
```



Query	with items	Total/ Term apperared	Log10(lastcolumn)
Canada	147	700/147	0.67778070526
Moncton	23	700/23	1.483370204
Toronto	65	700/65	1.03218468337

0



canada	Total	Frequency(f)
1	27	2
2	49	4
3	49	2
4	66	2
5	61	3
6	61	3
7	14	3
8	59	1
9	64	2
10	40	1
11	59	3
12	66	2
13	66	2
14	78	1
22	57	2
16	14	2
17	25	4
18	47	1
14	66	2
20	42	2
21	78	2
22	25	2
23	14	2
24	38	1
25	17	2
26	49	2

REFERENCES:

[1] “IBM Cognos Analytics,” *IBM Cognos Analytics*, 2020. [Online]. Available: https://www.ibm.com/ca-en/products/cognos-analytics. [Accessed: 01-Aug-2021]

[2] “IBM Docs,” *Ibm.com*, 03-Mar-2021. [Online]. Available: https://www.ibm.com/docs/en/cognos-analytics/10.2.2?topic=SSEP7J_10.2.2/com.ibm.swg.ba.cognos.wig_cr.10.2.2.doc/c_gtstd_c8_bi.html. [Accessed: 01-Aug-2021]

[3]Wikipedia Contributors, “IBM Cognos Analytics,” *Wikipedia*, 13-Jul-2021. [Online]. Available: https://en.wikipedia.org/wiki/IBM_Cognos_Analytics. [Accessed: 01-Aug-2021]

[4]“Kaggle: Your Home for Data Science,” *Kaggle.com*, 2021. [Online]. Available: https://www.kaggle.com/. [Accessed: 01-Aug-2021]

[5] “opencsv –,” *Sourceforge.net*, 2013. [Online]. Available: http://opencsv.sourceforge.net/. [Accessed: 01-Aug-2021]

[6] “Calc | LibreOffice - Free Office Suite - Based on OpenOffice - Compatible with Microsoft,” *Libreoffice.org*, 2021. [Online]. Available: https://www.libreoffice.org/discover/calc/. [Accessed: 01-Aug-2021]