CLI (Bash) & Data Management for Big Data Project - 1

[Surabhi Agarwal]

[17203535]

An assessment component submitted in part fulfilment for the module

COMP30770

Programming for Big Data

UCD School of Computer Science

University College Dublin

February 2020

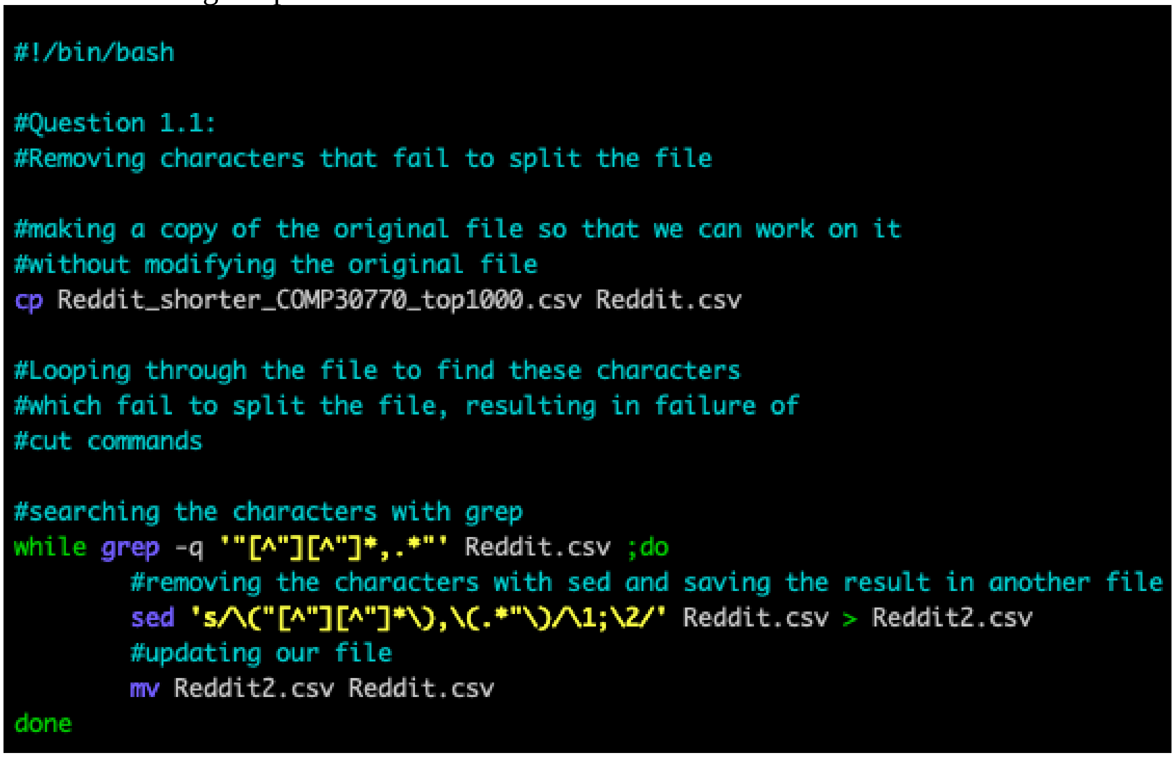
- **Introduction:**

    - This assignment is catered towards understanding CLI and bash and how CLI is used for performing big data tasks in today's world.
    - In the first section, I was given the dataset "Reddit_shorter_COMP30770_top1000.csv and I perform several data preprocessing startegies to clean the dataset and make it suitable for database management tasks.
    - In the second section, I use the cleaned dataset for data management using MySQL and MongoDB database management systems. I then perform serveral SQL and mongo queries to display trends in the dataset.
    - In the third section, I describe how CLI is used for big data tasks even though we have GUI systems like Hadoop. I compare NoSQL and Relational Database Management Systems. This task is very much linked to understanding the first two tasks and explaining them.
    - Finally, my last section is a review report on Google's whitepaper on the Bigtable distributed system.

- **Cleaning a dataset with Bash**

This task is targeted towards cleaning the given dataset containing Reddits so that we are able to use the preprocessed data for data management.
For completing this task, I first login to csserver.ucd.ie via ssh using my terminal and my login details and perform the following commands:

1. **Removing miscellaneous characters that make us unable to split the file.**
   I used the following script for this task:

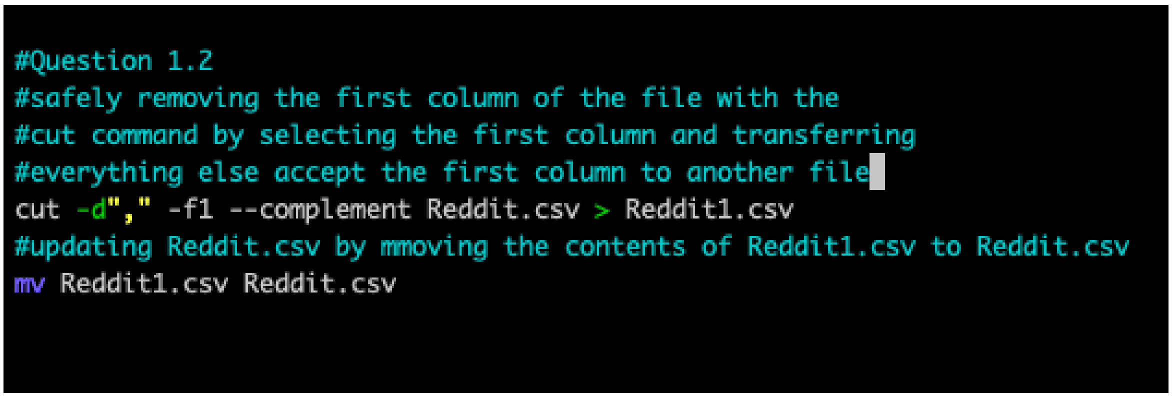


```
#!/bin/bash

#Question 1.1:
#Removing characters that fail to split the file

#making a copy of the original file so that we can work on it
#without modifying the original file
cp Reddit_shorter_COMP30770_top1000.csv Reddit.csv

#Looping through the file to find these characters
#which fail to split the file, resulting in failure of
#cut commands

#searching the characters with grep
while grep -q '"[^"][^"]*,.*"' Reddit.csv ;do
        #removing the characters with sed and saving the result in another file
        sed 's/\("[^"][^"]*\),\(.*"\)/\1;\2/' Reddit.csv > Reddit2.csv
        #updating our file
        mv Reddit2.csv Reddit.csv
done
```

   Firstly, I made a copy of the original file and named it Reddit.csv
   Using a while loop to loop through the file and searching for these characters "[^"][^"]*,.*" using grep, I removed these characters using sed and updating the changes to a temporary file called Reddit2.csv.
   Finally using mv, I updated the Reddit.csv file

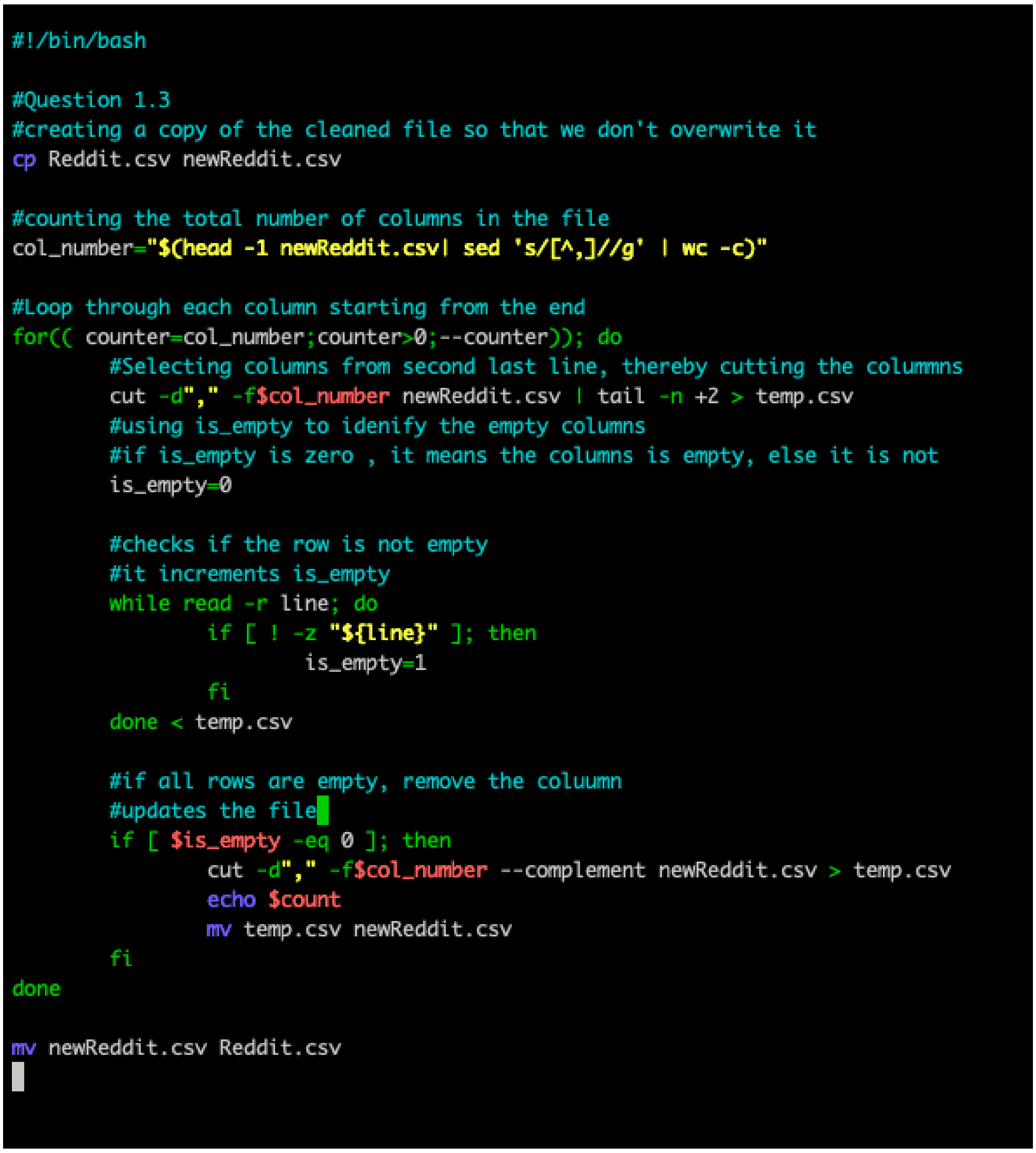
2. **Safely removing the first column with the cut command and the complement option**.

The first column is an index variable and I safely removed this column using cut and the complement option. What complement does over here is that it transfers everything but the first field from one file to another file. So I move the contents to a new file called Reddit1.csv and then update our Reddit.csv file using mv.



```
#Question 1.2
#safely removing the first column of the file with the
#cut command by selecting the first column and transferring
#everything else accept the first column to another file
cut -d"," -f1 --complement Reddit.csv > Reddit1.csv
#updating Reddit.csv by mmoving the contents of Reddit1.csv to Reddit.csv
mv Reddit1.csv Reddit.csv
```

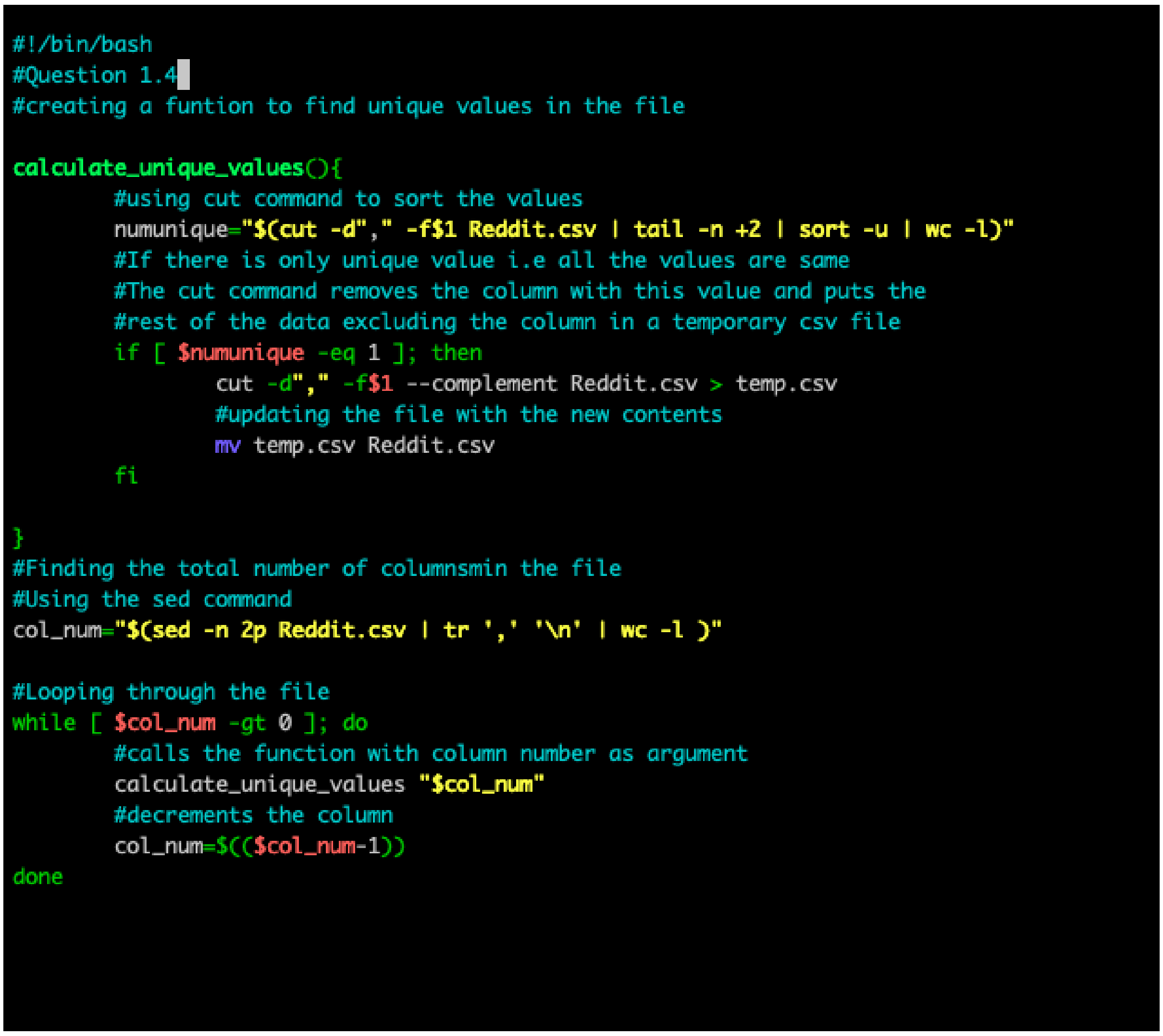
3. **Identifying and removing empty variables using a script:**



```
#!/bin/bash

#Question 1.3
#creating a copy of the cleaned file so that we don't overwrite it
cp Reddit.csv newReddit.csv

#counting the total number of columns in the file
col_number="$(head -1 newReddit.csv| sed 's/[^,]//g' | wc -c)"

#Loop through each column starting from the end
for(( counter=col_number;counter>0;--counter)); do
        #Selecting columns from second last line, thereby cutting the colummns
        cut -d"," -f$col_number newReddit.csv | tail -n +2 > temp.csv
        #using is_empty to idenify the empty columns
        #if is_empty is zero , it means the columns is empty, else it is not
        is_empty=0

        #checks if the row is not empty
        #it increments is_empty
        while read -r line; do
                if [ ! -z "${line}" ]; then
                        is_empty=1
                fi
        done < temp.csv

        #if all rows are empty, remove the coluumn
        #updates the file
        if [ $is_empty -eq 0 ]; then
                cut -d"," -f$col_number --complement newReddit.csv > temp.csv
                echo $count
                mv temp.csv newReddit.csv
        fi
done

mv newReddit.csv Reddit.csv
```

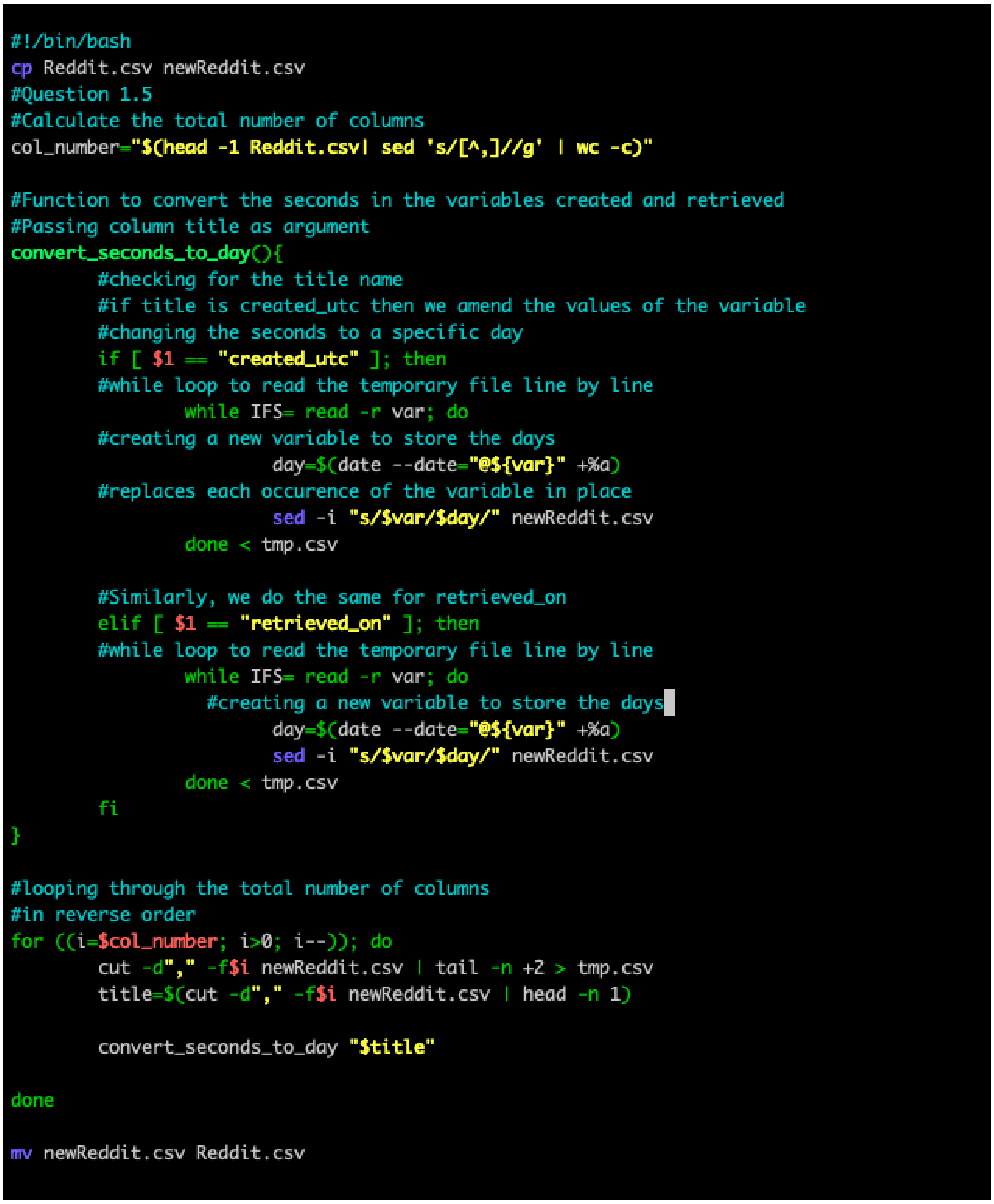
Since some variables in the file are completely empty, we can remove them using a script.
I calculated the total number of columns and stored them in a variable called col_number. Then, I looped through all the columns in the file starting from the end.
Then selecting columns using cut from the second last line onwards (excluding the title).
Now using a while loop I iterated through each column, I checked if the field in the column is empty or not by having a variable called is_empty initialized to 0 at start and incrementing it each time, the row in the particular column is not empty.
Now, if all the rows are found to be empty in the particular column i.e if is_empty is found to be 0, then using cut and complement, I transferred everything excluding that particular column in another file called temp.csv. Then, I updated my original Reddit.csv file.
I used the following script for this task:

4. **Identifying and removing variables which takes only one value**
In this task, I used a function called calculate_unique_values which takes total column number as an argument (the total column number is calculated using sed. Inside the function, I assigned num_unique to the sorted values. Now if numunique is equal to 1, then I performed a cut command where I transferred everything but that particular field into a new temporary, then finally renamed the fle to Reddit.csv. This function is called inside a while loop which loops through the file until the end.
This is how it works:



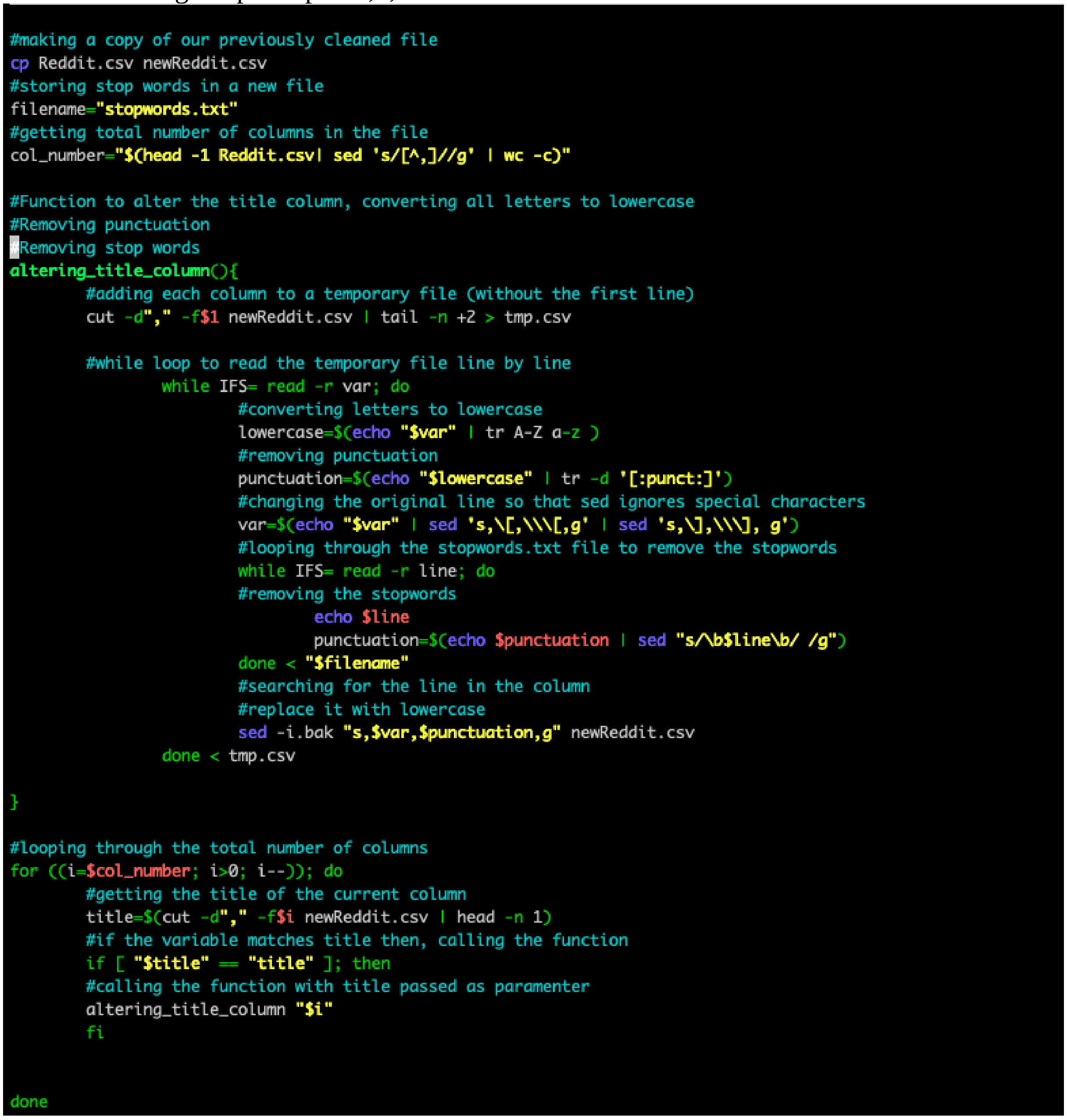
```
#!/bin/bash
#Question 1.4
#creating a funtion to find unique values in the file

calculate_unique_values(){
        #using cut command to sort the values
        numunique="$(cut -d"," -f$1 Reddit.csv | tail -n +2 | sort -u | wc -l)"
        #If there is only unique value i.e all the values are same
        #The cut command removes the column with this value and puts the
        #rest of the data excluding the column in a temporary csv file
        if [ $numunique -eq 1 ]; then
                cut -d"," -f$1 --complement Reddit.csv > temp.csv
                #updating the file with the new contents
                mv temp.csv Reddit.csv
        fi

}
#Finding the total number of columnsmin the file
#Using the sed command
col_num="$(sed -n 2p Reddit.csv | tr ',' '\n' | wc -l )"

#Looping through the file
while [ $col_num -gt 0 ]; do
        #calls the function with column number as argument
        calculate_unique_values "$col_num"
        #decrements the column
        col_num=$(($col_num-1))
done
```

5. **Converting seconds (epoch) to day format for the created_utc and retrieved_on variables**
   In this case, again I calculated the total number of columns using sed and then I used a function called convert_seconds_to_day which takes title column as argument. I then search for the columns "created_utc" and "retrieved_on". For both of them, I created a new variable called day to store the days (using the command –date). This line is used to convert the epoch variable to the day format.
   Then using sed, I replaced each occurrence of the varible in place. This function is called inside a while loop which loops through the file.
   Then finally I renamed the file to Reddit.csv. This is how it works:



```
#!/bin/bash
cp Reddit.csv newReddit.csv
#Question 1.5
#Calculate the total number of columns
col_number="$(head -1 Reddit.csv| sed 's/[^,]//g' | wc -c)"

#Function to convert the seconds in the variables created and retrieved
#Passing column title as argument
convert_seconds_to_day(){
        #checking for the title name
        #if title is created_utc then we amend the values of the variable
        #changing the seconds to a specific day
        if [ $1 == "created_utc" ]; then
        #while loop to read the temporary file line by line
                while IFS= read -r var; do
        #creating a new variable to store the days
                        day=$(date --date="@${var}" +%a)
        #replaces each occurence of the variable in place
                        sed -i "s/$var/$day/" newReddit.csv
                done < tmp.csv

        #Similarly, we do the same for retrieved_on
        elif [ $1 == "retrieved_on" ]; then
        #while loop to read the temporary file line by line
                while IFS= read -r var; do
                  #creating a new variable to store the days
                        day=$(date --date="@${var}" +%a)
                        sed -i "s/$var/$day/" newReddit.csv
                done < tmp.csv
        fi
}

#looping through the total number of columns
#in reverse order
for ((i=$col_number; i>0; i--)); do
        cut -d"," -f$i newReddit.csv | tail -n +2 > tmp.csv
        title=$(cut -d"," -f$i newReddit.csv | head -n 1)

        convert_seconds_to_day "$title"

done

mv newReddit.csv Reddit.csv
```

6. **a. Converting letters to lowercase, b. removing stopwords, c. removing punctuation, d. reducing the words to their basic stem words using a stemming technique**
   I used the following script for part a,b,c

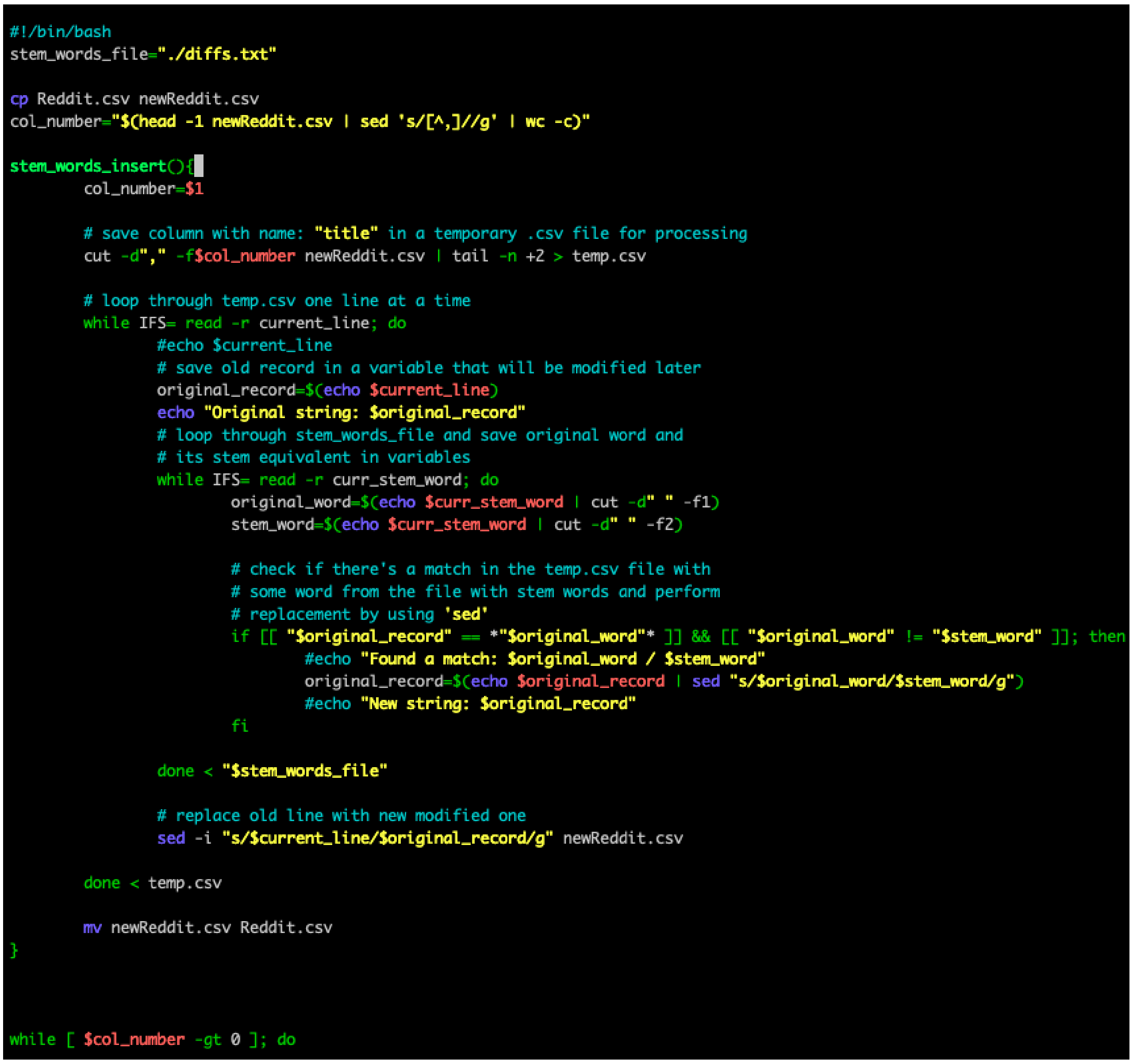


```
#making a copy of our previously cleaned file
cp Reddit.csv newReddit.csv
#storing stop words in a new file
filename="stopwords.txt"
#getting total number of columns in the file
col_number="$(head -1 Reddit.csv| sed 's/[^,]//g' | wc -c)"

#Function to alter the title column, converting all letters to lowercase
#Removing punctuation
#Removing stop words
altering_title_column(){
        #adding each column to a temporary file (without the first line)
        cut -d"," -f$1 newReddit.csv | tail -n +2 > tmp.csv

        #while loop to read the temporary file line by line
                while IFS= read -r var; do
                        #converting letters to lowercase
                        lowercase=$(echo "$var" | tr A-Z a-z )
                        #removing punctuation
                        punctuation=$(echo "$lowercase" | tr -d '[:punct:]')
                        #changing the original line so that sed ignores special characters
                        var=$(echo "$var" | sed 's,\[,\\\[,g' | sed 's,\],\\\], g')
                        #looping through the stopwords.txt file to remove the stopwords
                        while IFS= read -r line; do
                        #removing the stopwords
                                echo $line
                                punctuation=$(echo $punctuation | sed "s/\b$line\b/ /g")
                        done < "$filename"
                        #searching for the line in the column
                        #replace it with lowercase
                        sed -i.bak "s,$var,$punctuation,g" newReddit.csv
                done < tmp.csv

}

#looping through the total number of columns
for ((i=$col_number; i>0; i--)); do
        #getting the title of the current column
        title=$(cut -d"," -f$i newReddit.csv | head -n 1)
        #if the variable matches title then, calling the function
        if [ "$title" == "title" ]; then
        #calling the function with title passed as paramenter
        altering_title_column "$i"
        fi


done
```

For this task, I used a function called altering_title_column. Firstly, I added each column to a temporary file (without the title). Then I used a while loop to read the file line by line and then convert each letter to lowercase using tr A-Z a-z and store it in a lowercase variable, remove the punctuation using tr -d '[:punct:]'(storing it in the punctuation variable) and used a variable to amend the original line so that sed ignores special characters.
To remove the stop words, I stored them in a txt file called stopwords.txt. I then used a while loop to loop through this txt file and stored each stop words in the same variable called punctuation which we used to remove the punctuation symbols. Then finally removing these variables using sed.
I called this function by looping through the total number of columns in the file. I assign the title column to the first row in the dataset and if the value of title matched "title", I call the function with the ith column as an argument.

In the next task, we are supposed to reduce the words to the step words using a stemming technique and a disctionary. I used this as my dictionary and saved it in a text file on my server called “diffs.txt”. I used a function called stem_words_insert where I stored the first field of the file diffs.txt in a variable called original_word and the second field in a variable called stem_word. Now I check if the original record i.e original field in the Reddit.csv matches the original word in the stemming dictionary or not. If it matches, then I replace the original record with it’s stem word.
To call this function, I loop through the file and pass column number as argument to the function.

This is how the script works:

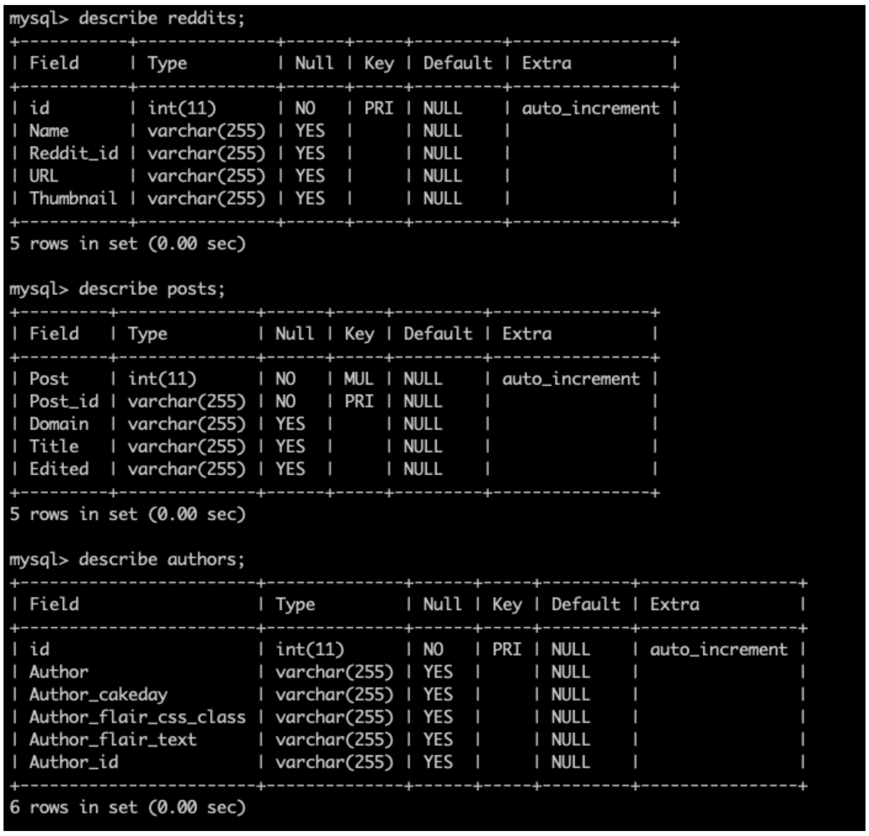


```
#!/bin/bash
stem_words_file="./diffs.txt"

cp Reddit.csv newReddit.csv
col_number="$(head -1 newReddit.csv | sed 's/[^,]//g' | wc -c)"

stem_words_insert(){
        col_number=$1

        # save column with name: "title" in a temporary .csv file for processing
        cut -d"," -f$col_number newReddit.csv | tail -n +2 > temp.csv

        # loop through temp.csv one line at a time
        while IFS= read -r current_line; do
                #echo $current_line
                # save old record in a variable that will be modified later
                original_record=$(echo $current_line)
                echo "Original string: $original_record"
                # loop through stem_words_file and save original word and
                # its stem equivalent in variables
                while IFS= read -r curr_stem_word; do
                        original_word=$(echo $curr_stem_word | cut -d" " -f1)
                        stem_word=$(echo $curr_stem_word | cut -d" " -f2)

                        # check if there's a match in the temp.csv file with
                        # some word from the file with stem words and perform
                        # replacement by using 'sed'
                        if [[ "$original_record" == *"$original_word"* ]] && [[ "$original_word" != "$stem_word" ]]; then
                                #echo "Found a match: $original_word / $stem_word"
                                original_record=$(echo $original_record | sed "s/$original_word/$stem_word/g")
                                #echo "New string: $original_record"
                        fi

                done < "$stem_words_file"

                # replace old line with new modified one
                sed -i "s/$current_line/$original_record/g" newReddit.csv

        done < temp.csv

        mv newReddit.csv Reddit.csv
}


while [ $col_number -gt 0 ]; do
```

- **Data Management**

Uploading the cleaned dataset in the database management systems.

1. Creating a database in MySQL with 3 tables i.e authors, posts and reddits. The tables I created have the following models:

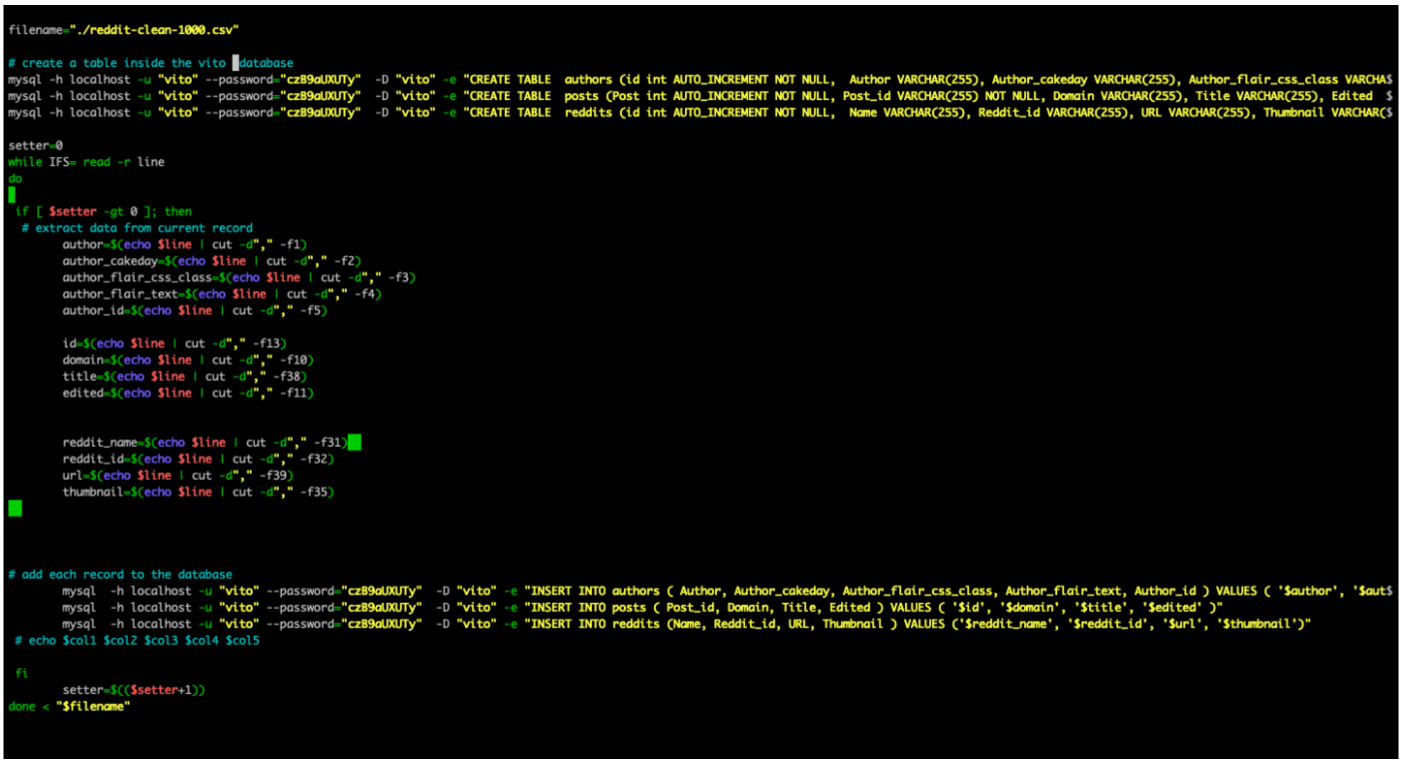


```
mysql> describe reddits;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| id        | int(11)      | NO   | PRI | NULL    | auto_increment |
| Name      | varchar(255) | YES  |     | NULL    |                |
| Reddit_id | varchar(255) | YES  |     | NULL    |                |
| URL       | varchar(255) | YES  |     | NULL    |                |
| Thumbnail | varchar(255) | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)

mysql> describe posts;
+---------+--------------+------+-----+---------+----------------+
| Field   | Type         | Null | Key | Default | Extra          |
+---------+--------------+------+-----+---------+----------------+
| Post    | int(11)      | NO   | MUL | NULL    | auto_increment |
| Post_id | varchar(255) | NO   | PRI | NULL    |                |
| Domain  | varchar(255) | YES  |     | NULL    |                |
| Title   | varchar(255) | YES  |     | NULL    |                |
| Edited  | varchar(255) | YES  |     | NULL    |                |
+---------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)

mysql> describe authors;
+-----------------------+--------------+------+-----+---------+----------------+
| Field                 | Type         | Null | Key | Default | Extra          |
+-----------------------+--------------+------+-----+---------+----------------+
| id                    | int(11)      | NO   | PRI | NULL    | auto_increment |
| Author                | varchar(255) | YES  |     | NULL    |                |
| Author_cakeday        | varchar(255) | YES  |     | NULL    |                |
| Author_flair_css_class| varchar(255) | YES  |     | NULL    |                |
| Author_flair_text     | varchar(255) | YES  |     | NULL    |                |
| Author_id             | varchar(255) | YES  |     | NULL    |                |
+-----------------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

Firstly, I used my database ‘vito’ to create the above tables.

- The table reddits contains the subreddit id, the subreddit name, url and thumbnail. It also contains id auto_increment which creates an id for each of the reddits as an additional column in the table. This attribute is unique and acts as our primary key.
- Similarly the table posts contains the attributes post (auto_increment) as our foreign key referring to the authors id from the table authors. The table posts also contain the post_id, domain, title and an attribute to check if the post was edited or not.
- The table authors contains attributes id (primary key), author name, author cake day, author_flair_css-class, author_flair_text and the author_id.
- Over here, for the purpose of this assignment, I have chosen the post table and the author table to have a 1-1 relationship having the id as the foreign key.

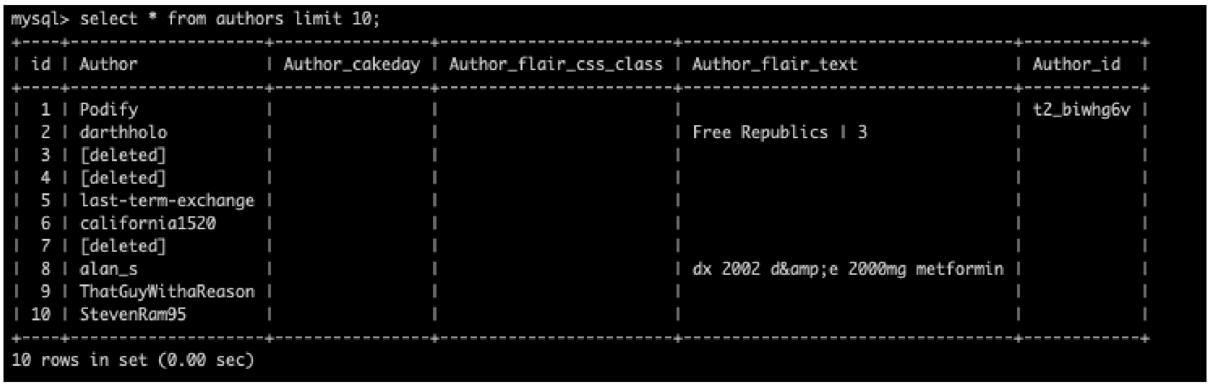
2. The script below creates and populates the database for each value in the file.
   Firslty, I used my database vito and created tables: authors, posts and reddits.
   Then using a while loop to loop through the while, I extracted data from the file and assigned values corresponding to each of my fields in the tables.
   Thereafter, I populated my tables using these fields and the insert commands.
   I used a variable called setter to make sure that we populate everything but not the title row.



```
filename="./reddit-clean-1000.csv"

# create a table inside the vito database
mysql -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "CREATE TABLE  authors (id int AUTO_INCREMENT NOT NULL,  Author VARCHAR(255), Author_cakeday VARCHAR(255), Author_flair_css_class VARCHA$
mysql -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "CREATE TABLE  posts (Post int AUTO_INCREMENT NOT NULL, Post_id VARCHAR(255) NOT NULL, Domain VARCHAR(255), Title VARCHAR(255), Edited  $
mysql -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "CREATE TABLE  reddits (id int AUTO_INCREMENT NOT NULL,  Name VARCHAR(255), Reddit_id VARCHAR(255), URL VARCHAR(255), Thumbnail VARCHAR($

setter=0
while IFS= read -r line
do

 if [ $setter -gt 0 ]; then
  # extract data from current record
        author=$(echo $line | cut -d"," -f1)
        author_cakeday=$(echo $line | cut -d"," -f2)
        author_flair_css_class=$(echo $line | cut -d"," -f3)
        author_flair_text=$(echo $line | cut -d"," -f4)
        author_id=$(echo $line | cut -d"," -f5)

        id=$(echo $line | cut -d"," -f13)
        domain=$(echo $line | cut -d"," -f10)
        title=$(echo $line | cut -d"," -f38)
        edited=$(echo $line | cut -d"," -f11)


        reddit_name=$(echo $line | cut -d"," -f31)
        reddit_id=$(echo $line | cut -d"," -f32)
        url=$(echo $line | cut -d"," -f39)
        thumbnail=$(echo $line | cut -d"," -f35)




# add each record to the database
        mysql  -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "INSERT INTO authors ( Author, Author_cakeday, Author_flair_css_class, Author_flair_text, Author_id ) VALUES ( '$author', '$aut$
        mysql  -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "INSERT INTO posts ( Post_id, Domain, Title, Edited ) VALUES ( '$id', '$domain', '$title', '$edited' )"
        mysql  -h localhost -u "vito" --password="czB9aUXUTy"  -D "vito" -e "INSERT INTO reddits (Name, Reddit_id, URL, Thumbnail ) VALUES ('$reddit_name', '$reddit_id', '$url', '$thumbnail')"
 # echo $col1 $col2 $col3 $col4 $col5

 fi
        setter=$(($setter+1))
done < "$filename"
```

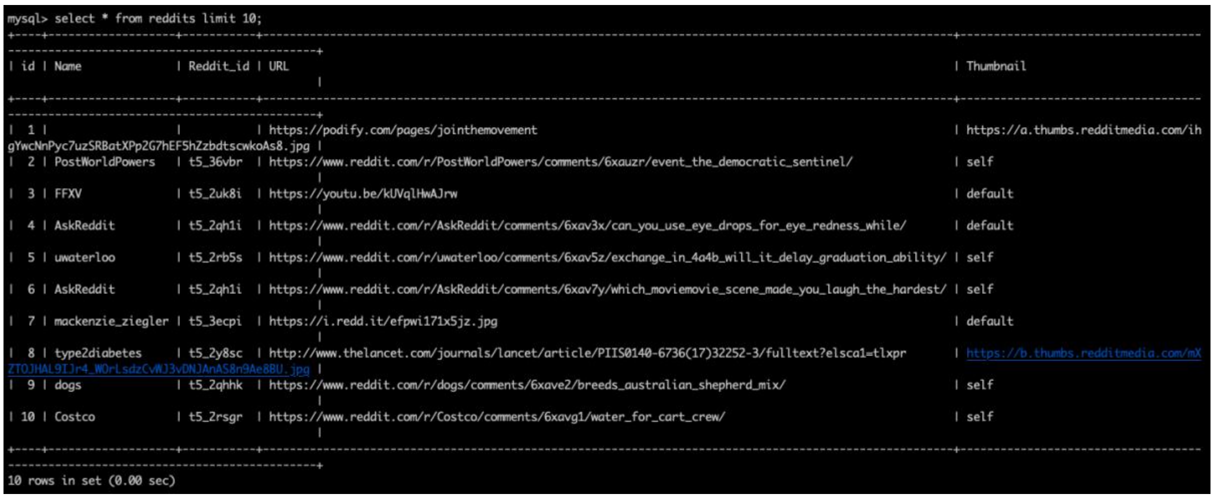
Now performing some SQL queries to see trends in our tables.

3. SQL Queries
   - To select everything from the table authors, displaying top 10 columns:
     Query: select * from authors limit 10;



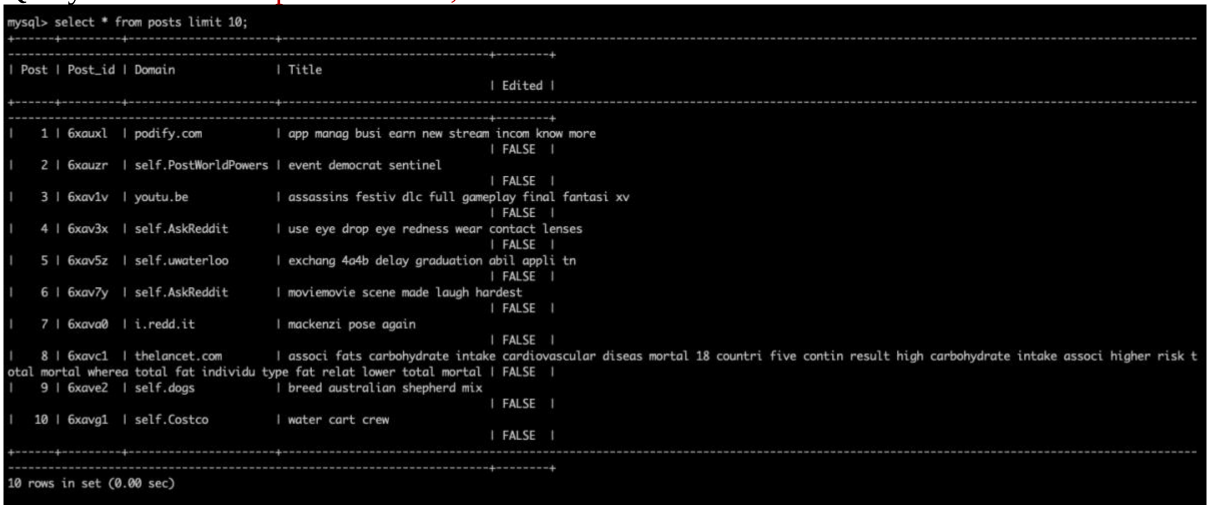
```
mysql> select * from authors limit 10;
+----+--------------------+----------------+------------------------+------------------------------------+------------+
| id | Author             | Author_cakeday | Author_flair_css_class | Author_flair_text                  | Author_id  |
+----+--------------------+----------------+------------------------+------------------------------------+------------+
|  1 | Podify             |                |                        |                                    | t2_biwhg6v |
|  2 | darthholo          |                |                        | Free Republics | 3                 |            |
|  3 | [deleted]          |                |                        |                                    |            |
|  4 | [deleted]          |                |                        |                                    |            |
|  5 | last-term-exchange |                |                        |                                    |            |
|  6 | california1520     |                |                        |                                    |            |
|  7 | [deleted]          |                |                        |                                    |            |
|  8 | alan_s             |                |                        | dx 2002 d&amp;e 2000mg metformin   |            |
|  9 | ThatGuyWithaReason |                |                        |                                    |            |
| 10 | StevenRam95        |                |                        |                                    |            |
+----+--------------------+----------------+------------------------+------------------------------------+------------+
10 rows in set (0.00 sec)
```

- To select everything from the table reddits, displaying top 10 columns
  Query: select * from reddits limit 10;



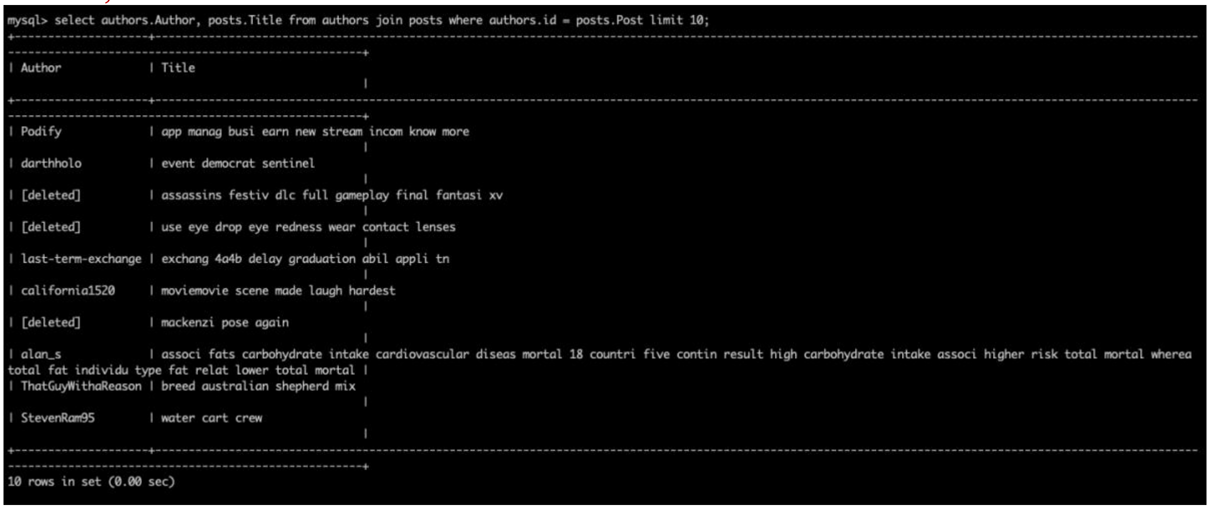
```
mysql> select * from reddits limit 10;
+----+------------------+----------+----------------------------------------------------------------------------------------------------+--------------------------------
-----------------------------------------------+
| id | Name             | Reddit_id | URL                                                                                               | Thumbnail
                                               |
+----+------------------+----------+----------------------------------------------------------------------------------------------------+--------------------------------
-----------------------------------------------+
|  1 |                  |          | https://podify.com/pages/jointhemovement                                                           | https://a.thumbs.redditmedia.com/ih
gYwcNnPyc7uzSRBatXPp2G7hEF5hZzbdtscwkoAs8.jpg |
|  2 | PostWorldPowers  | t5_36vbr | https://www.reddit.com/r/PostWorldPowers/comments/6xauzr/event_the_democratic_sentinel/            | self
                                               |
|  3 | FFXV             | t5_2uk8i | https://youtu.be/kUVqlHwAJrw                                                                       | default
                                               |
|  4 | AskReddit        | t5_2qh1i | https://www.reddit.com/r/AskReddit/comments/6xav3x/can_you_use_eye_drops_for_eye_redness_while/    | default
                                               |
|  5 | uwaterloo        | t5_2rb5s | https://www.reddit.com/r/uwaterloo/comments/6xav5z/exchange_in_4a4b_will_it_delay_graduation_ability/ | self
                                               |
|  6 | AskReddit        | t5_2qh1i | https://www.reddit.com/r/AskReddit/comments/6xav7y/which_moviemovie_scene_made_you_laugh_the_hardest/ | self
                                               |
|  7 | mackenzie_ziegler | t5_3ecpi | https://i.redd.it/efpwi171x5jz.jpg                                                                | default
                                               |
|  8 | type2diabetes    | t5_2y8sc | http://www.thelancet.com/journals/lancet/article/PIIS0140-6736(17)32252-3/fulltext?elsca1=tlxpr   | https://b.thumbs.redditmedia.com/mX
ZTOJHAL9IJr4_WOrLsdzCvWJ3vDNJAnAS8n9Ae8BU.jpg |
|  9 | dogs             | t5_2qhhk | https://www.reddit.com/r/dogs/comments/6xave2/breeds_australian_shepherd_mix/                     | self
                                               |
| 10 | Costco           | t5_2rsgr | https://www.reddit.com/r/Costco/comments/6xavg1/water_for_cart_crew/                              | self
                                               |
+----+------------------+----------+----------------------------------------------------------------------------------------------------+--------------------------------
-----------------------------------------------+
10 rows in set (0.00 sec)
```

- To select everything from the table posts
- Query: select * from posts limit 10;



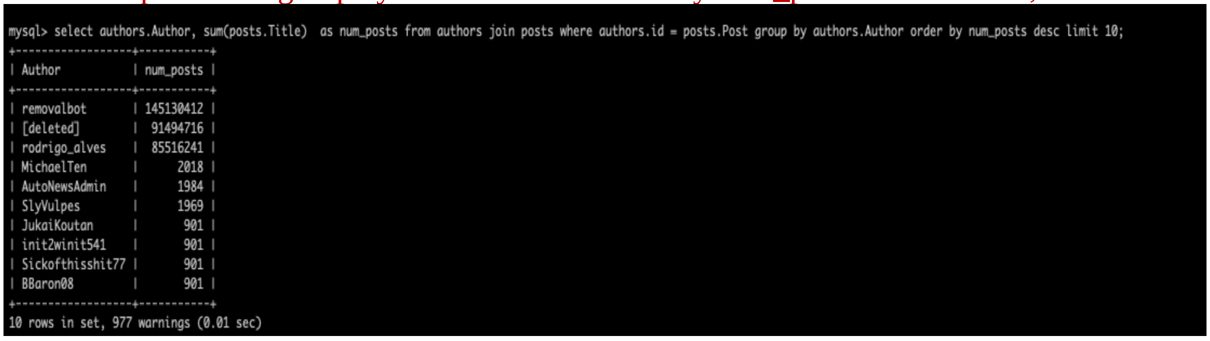
```
mysql> select * from posts limit 10;
+------+---------+----------------------+--------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------+--------+
| Post | Post_id | Domain               | Title
                                                             | Edited |
+------+---------+----------------------+--------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------+--------+
|    1 | 6xauxl  | podify.com           | app manag busi earn new stream incom know more
                                                             | FALSE  |
|    2 | 6xauzr  | self.PostWorldPowers | event democrat sentinel
                                                             | FALSE  |
|    3 | 6xav1v  | youtu.be             | assassins festiv dlc full gameplay final fantasi xv
                                                             | FALSE  |
|    4 | 6xav3x  | self.AskReddit       | use eye drop eye redness wear contact lenses
                                                             | FALSE  |
|    5 | 6xav5z  | self.uwaterloo       | exchang 4a4b delay graduation abil appli tn
                                                             | FALSE  |
|    6 | 6xav7y  | self.AskReddit       | moviemovie scene made laugh hardest
                                                             | FALSE  |
|    7 | 6xava0  | i.redd.it            | mackenzi pose again
                                                             | FALSE  |
|    8 | 6xavc1  | thelancet.com        | associ fats carbohydrate intake cardiovascular diseas mortal 18 countri five contin result high carbohydrate intake associ higher risk t
otal mortal wherea total fat individu type fat relat lower total mortal | FALSE  |
|    9 | 6xave2  | self.dogs            | breed australian shepherd mix
                                                             | FALSE  |
|   10 | 6xavg1  | self.Costco          | water cart crew
                                                             | FALSE  |
+------+---------+----------------------+--------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------+--------+
10 rows in set (0.00 sec)
```

- To perform a join between the table authors and posts and return the author name
  and the title of his post
  Query: select authors.Author, posts.Title from authors join posts where authors.id = posts.Post limit 10;



```
mysql> select authors.Author, posts.Title from authors join posts where authors.id = posts.Post limit 10;
+-------------------+--------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------+
| Author            | Title
                                                  |
+-------------------+--------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------+
| Podify            | app manag busi earn new stream incom know more
                                                  |
| darthholo         | event democrat sentinel
                                                  |
| [deleted]         | assassins festiv dlc full gameplay final fantasi xv
                                                  |
| [deleted]         | use eye drop eye redness wear contact lenses
                                                  |
| last-term-exchange | exchang 4a4b delay graduation abil appli tn
                                                  |
| california1520    | moviemovie scene made laugh hardest
                                                  |
| [deleted]         | mackenzi pose again
                                                  |
| alan_s            | associ fats carbohydrate intake cardiovascular diseas mortal 18 countri five contin result high carbohydrate intake associ higher risk total mortal wherea
total fat individu type fat relat lower total mortal |
| ThatGuyWithaReason | breed australian shepherd mix
                                                  |
| StevenRam95       | water cart crew
                                                  |
+-------------------+--------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------+
10 rows in set (0.00 sec)
```

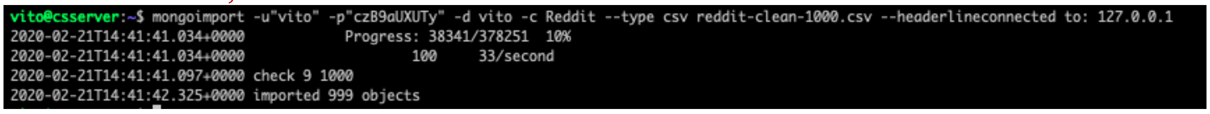
- To return the number of posts by each author in descending order
- Query: select authors.Author, sum(posts.Title) as num_posts from authors join posts where authors.id = posts.Post group by authors.Author order by num_posts desc limit 10;



```
mysql> select authors.Author, sum(posts.Title)  as num_posts from authors join posts where authors.id = posts.Post group by authors.Author order by num_posts desc limit 10;
+-----------------+-----------+
| Author          | num_posts |
+-----------------+-----------+
| removalbot      | 145130412 |
| [deleted]       |  91494716 |
| rodrigo_alves   |  85516241 |
| MichaelTen      |      2018 |
| AutoNewsAdmin   |      1984 |
| SlyVulpes       |      1969 |
| JukaiKoutan     |       901 |
| init2winit541   |       901 |
| Sickofthisshit77 |      901 |
| BBaron08        |       901 |
+-----------------+-----------+
10 rows in set, 977 warnings (0.01 sec)
```

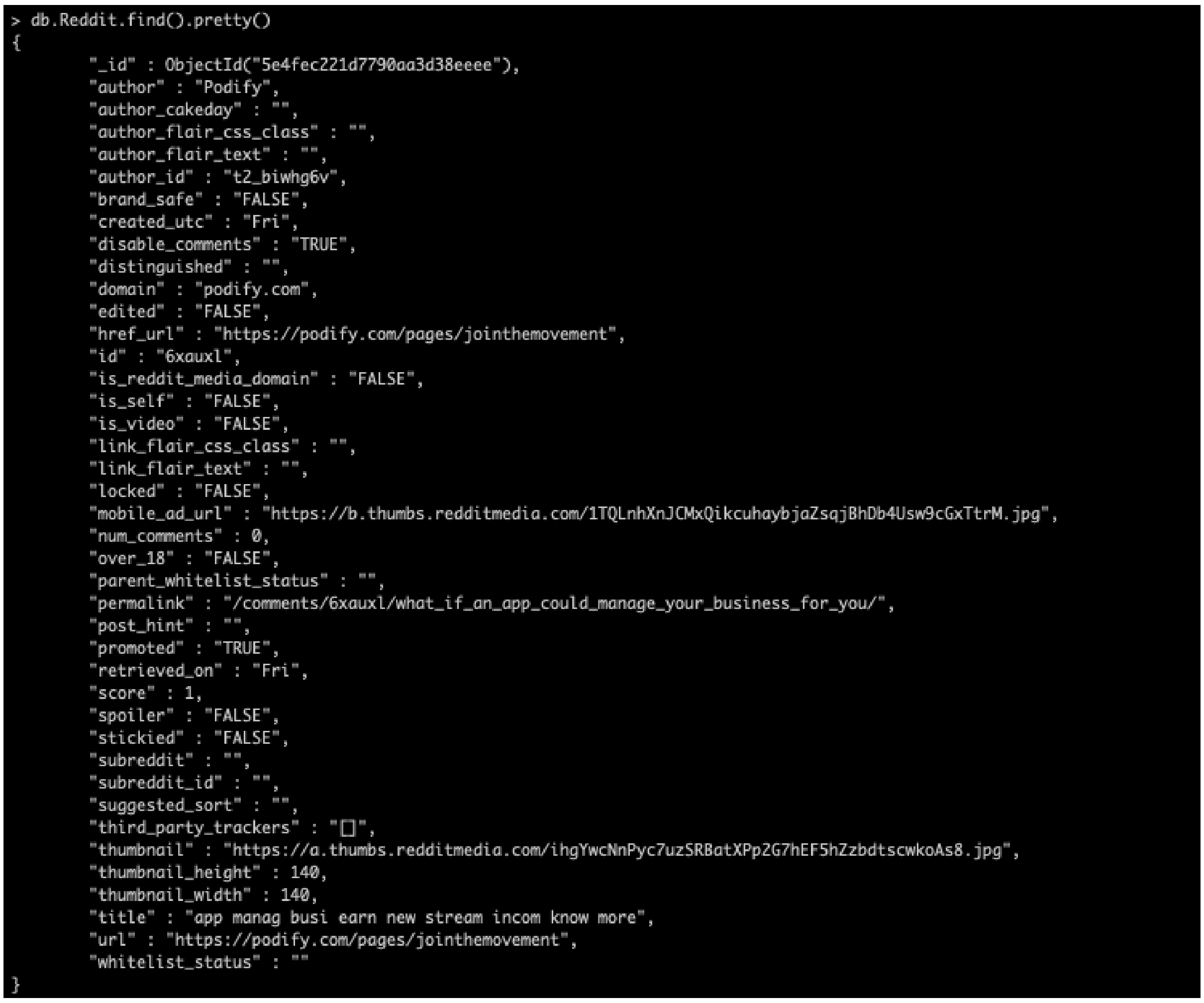
-

4. Creating and populating a MongoDB collection called ‘Reddit’ with every line in the CSV file as an entry/document.
   Using the query: mongoimport -u”vito” -p”czB9aUXUTy” -d vito -c Reddiy –type csv reddit-clean-1000.csv –headerline;



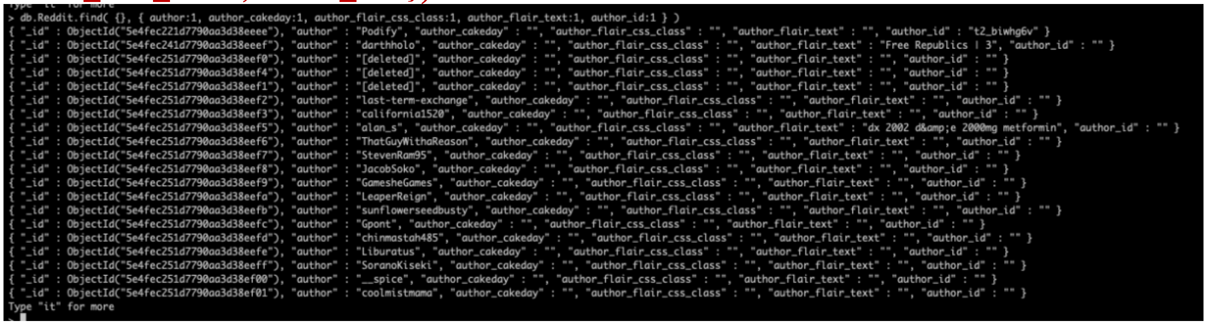
```
vito@csserver:~$ mongoimport -u"vito" -p"czB9aUXUTy" -d vito -c Reddit --type csv reddit-clean-1000.csv --headerlineconnected to: 127.0.0.1
2020-02-21T14:41:41.034+0000            Progress: 38341/378251  10%
2020-02-21T14:41:41.034+0000                    100     33/second
2020-02-21T14:41:41.097+0000 check 9 1000
2020-02-21T14:41:42.325+0000 imported 999 objects
```

Now printing the collection Reddit:



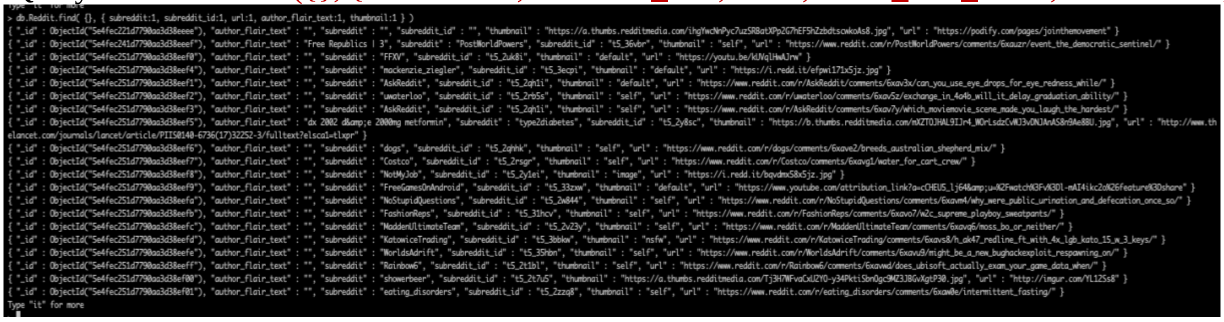
```
> db.Reddit.find().pretty()
{
        "_id" : ObjectId("5e4fec221d7790aa3d38eeee"),
        "author" : "Podify",
        "author_cakeday" : "",
        "author_flair_css_class" : "",
        "author_flair_text" : "",
        "author_id" : "t2_biwhg6v",
        "brand_safe" : "FALSE",
        "created_utc" : "Fri",
        "disable_comments" : "TRUE",
        "distinguished" : "",
        "domain" : "podify.com",
        "edited" : "FALSE",
        "href_url" : "https://podify.com/pages/jointhemovement",
        "id" : "6xauxl",
        "is_reddit_media_domain" : "FALSE",
        "is_self" : "FALSE",
        "is_video" : "FALSE",
        "link_flair_css_class" : "",
        "link_flair_text" : "",
        "locked" : "FALSE",
        "mobile_ad_url" : "https://b.thumbs.redditmedia.com/1TQLnhXnJCMxQikcuhaybjaZsqjBhDb4Usw9cGxTtrM.jpg",
        "num_comments" : 0,
        "over_18" : "FALSE",
        "parent_whitelist_status" : "",
        "permalink" : "/comments/6xauxl/what_if_an_app_could_manage_your_business_for_you/",
        "post_hint" : "",
        "promoted" : "TRUE",
        "retrieved_on" : "Fri",
        "score" : 1,
        "spoiler" : "FALSE",
        "stickied" : "FALSE",
        "subreddit" : "",
        "subreddit_id" : "",
        "suggested_sort" : "",
        "third_party_trackers" : "[]",
        "thumbnail" : "https://a.thumbs.redditmedia.com/ihgYwcNnPyc7uzSRBatXPp2G7hEF5hZzbdtscwkoAs8.jpg",
        "thumbnail_height" : 140,
        "thumbnail_width" : 140,
        "title" : "app manag busi earn new stream incom know more",
        "url" : "https://podify.com/pages/jointhemovement",
        "whitelist_status" : ""
}
```

5. Writing the previous SQL queries again with. Mongo
   - To select author related information as done in the SQL query
     Query: db.Reddit.find({}, { author:1, author_cakeday:1, author_flair_css_class:1, author_flair_text:1, author_id:1})



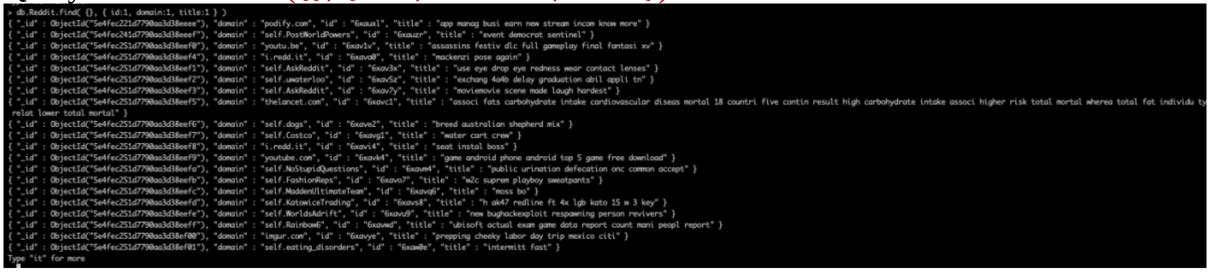
```
> db.Reddit.find( {}, { author:1, author_cakeday:1, author_flair_css_class:1, author_flair_text:1, author_id:1 } )
{ "_id" : ObjectId("5e4fec221d7790aa3d38eeee"), "author" : "Podify", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "t2_biwhg6v" }
{ "_id" : ObjectId("5e4fec241d7790aa3d38eeef"), "author" : "darthholo", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "Free Republics | 3", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef0"), "author" : "[deleted]", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef4"), "author" : "[deleted]", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef1"), "author" : "[deleted]", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef2"), "author" : "last-term-exchange", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef3"), "author" : "california1520", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef5"), "author" : "alan_s", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "dx 2002 d&amp;e 2000mg metformin", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef6"), "author" : "ThatGuyWithaReason", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef7"), "author" : "StevenRam95", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef8"), "author" : "JacobSoko", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef9"), "author" : "GamesheGames", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefa"), "author" : "LeaperReign", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefb"), "author" : "sunflowerseedbusty", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefc"), "author" : "Gpont", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefd"), "author" : "chinmastah485", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefe"), "author" : "Liburatus", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eeff"), "author" : "SoranoKiseki", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef00"), "author" : "__spice", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef01"), "author" : "coolmistmama", "author_cakeday" : "", "author_flair_css_class" : "", "author_flair_text" : "", "author_id" : "" }
Type "it" for more
```

   - To select information for the subreddits
     Query: db.Reddit.find({}, { subreddit:1, subreddit_id:1, url: 1,author_flair_text:1, thumbnail:1})



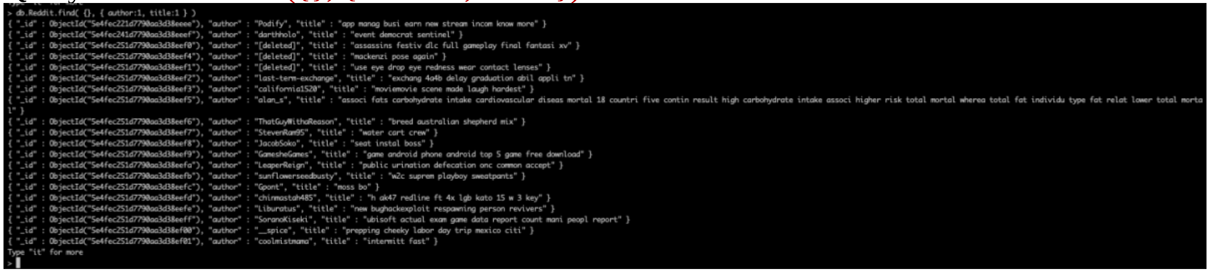
```
> db.Reddit.find( {}, { subreddit:1, subreddit_id:1, url:1, author_flair_text:1, thumbnail:1 } )
{ "_id" : ObjectId("5e4fec221d7790aa3d38eeee"), "author_flair_text" : "", "subreddit" : "", "subreddit_id" : "", "thumbnail" : "https://a.thumbs.redditmedia.com/ihgYwcNnPyc7uz5RBatXPp2G7hEF5hZzbdtscwkoAs8.jpg", "url" : "https://podify.com/pages/jointhemovement" }
{ "_id" : ObjectId("5e4fec241d7790aa3d38eeef"), "author_flair_text" : "Free Republics | 3", "subreddit" : "PostWorldPowers", "subreddit_id" : "t5_36vbr", "thumbnail" : "self", "url" : "https://www.reddit.com/r/PostWorldPowers/comments/6xauzr/event_the_democratic_sentinel/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef0"), "author_flair_text" : "", "subreddit" : "FFXV", "subreddit_id" : "t5_2uk8i", "thumbnail" : "default", "url" : "https://youtu.be/kUVqlHwAJrw" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef4"), "author_flair_text" : "", "subreddit" : "mackenzie_ziegler", "subreddit_id" : "t5_3ecpi", "thumbnail" : "default", "url" : "https://i.redd.it/efpwi171x5jz.jpg" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef1"), "author_flair_text" : "", "subreddit" : "AskReddit", "subreddit_id" : "t5_2qh1i", "thumbnail" : "default", "url" : "https://www.reddit.com/r/AskReddit/comments/6xav3x/can_you_use_eye_drops_for_eye_redness_while/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef2"), "author_flair_text" : "", "subreddit" : "uwaterloo", "subreddit_id" : "t5_2rb5s", "thumbnail" : "self", "url" : "https://www.reddit.com/r/uwaterloo/comments/6xav5z/exchange_in_4a4b_will_it_delay_graduation_ability/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef3"), "author_flair_text" : "", "subreddit" : "AskReddit", "subreddit_id" : "t5_2qh1i", "thumbnail" : "self", "url" : "https://www.reddit.com/r/AskReddit/comments/6xav7y/which_moviemovie_scene_made_you_laugh_the_hardest/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef5"), "author_flair_text" : "dx 2002 d&amp;e 2000mg metformin", "subreddit" : "type2diabetes", "subreddit_id" : "t5_2y8sc", "thumbnail" : "https://b.thumbs.redditmedia.com/mXZTOJHAL9IJr4_WOrLsdzCvWJ3vDNJAnAS8n9Ae8BU.jpg", "url" : "http://www.th
elancet.com/journals/lancet/article/PIIS0140-6736(17)32252-3/fulltext?elsca1=tlxpr" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef6"), "author_flair_text" : "", "subreddit" : "dogs", "subreddit_id" : "t5_2qhhk", "thumbnail" : "self", "url" : "https://www.reddit.com/r/dogs/comments/6xave2/breeds_australian_shepherd_mix/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef7"), "author_flair_text" : "", "subreddit" : "Costco", "subreddit_id" : "t5_2rsgr", "thumbnail" : "self", "url" : "https://www.reddit.com/r/Costco/comments/6xavg1/water_for_cart_crew/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef8"), "author_flair_text" : "", "subreddit" : "NotMyJob", "subreddit_id" : "t5_2y1ei", "thumbnail" : "image", "url" : "https://i.redd.it/bqvdmx58x5jz.jpg" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef9"), "author_flair_text" : "", "subreddit" : "FreeGamesOnAndroid", "subreddit_id" : "t5_33zxw", "thumbnail" : "default", "url" : "https://www.youtube.com/attribution_link?a=cCHEU5_lj64&amp;u=%2Fwatch%3Fv%3Dl-mAI4ikc2o%26feature%3Dshare" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefa"), "author_flair_text" : "", "subreddit" : "NoStupidQuestions", "subreddit_id" : "t5_2w844", "thumbnail" : "self", "url" : "https://www.reddit.com/r/NoStupidQuestions/comments/6xavm4/why_were_public_urination_and_defecation_once_so/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefb"), "author_flair_text" : "", "subreddit" : "FashionReps", "subreddit_id" : "t5_31hcv", "thumbnail" : "self", "url" : "https://www.reddit.com/r/FashionReps/comments/6xavo7/w2c_supreme_playboy_sweatpants/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefc"), "author_flair_text" : "", "subreddit" : "MaddenUltimateTeam", "subreddit_id" : "t5_2v23y", "thumbnail" : "self", "url" : "https://www.reddit.com/r/MaddenUltimateTeam/comments/6xavq6/moss_bo_or_neither/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefd"), "author_flair_text" : "", "subreddit" : "KatowiceTrading", "subreddit_id" : "t5_3bbkw", "thumbnail" : "nsfw", "url" : "https://www.reddit.com/r/KatowiceTrading/comments/6xavs8/h_ak47_redline_ft_with_4x_lgb_kato_15_w_3_keys/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefe"), "author_flair_text" : "", "subreddit" : "WorldsAdrift", "subreddit_id" : "t5_35hbn", "thumbnail" : "self", "url" : "https://www.reddit.com/r/WorldsAdrift/comments/6xavu9/might_be_a_new_bughackexploit_respawning_on/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eeff"), "author_flair_text" : "", "subreddit" : "Rainbow6", "subreddit_id" : "t5_2t1bl", "thumbnail" : "self", "url" : "https://www.reddit.com/r/Rainbow6/comments/6xavwd/does_ubisoft_actually_exam_your_game_data_when/" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef00"), "author_flair_text" : "", "subreddit" : "showerbeer", "subreddit_id" : "t5_2t7u5", "thumbnail" : "https://a.thumbs.redditmedia.com/Tj3H7WFvaCxU2YO-y34PktiSbnOgc9MZ3JBGvXgtP30.jpg", "url" : "http://imgur.com/YL12Ss8" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef01"), "author_flair_text" : "", "subreddit" : "eating_disorders", "subreddit_id" : "t5_2zzq8", "thumbnail" : "self", "url" : "https://www.reddit.com/r/eating_disorders/comments/6xaw0e/intermittent_fasting/" }
Type "it" for more
```

   - To select everything from the table posts
   - Query: db.Reddit.find({}, { id:1, domain:1, title:1})



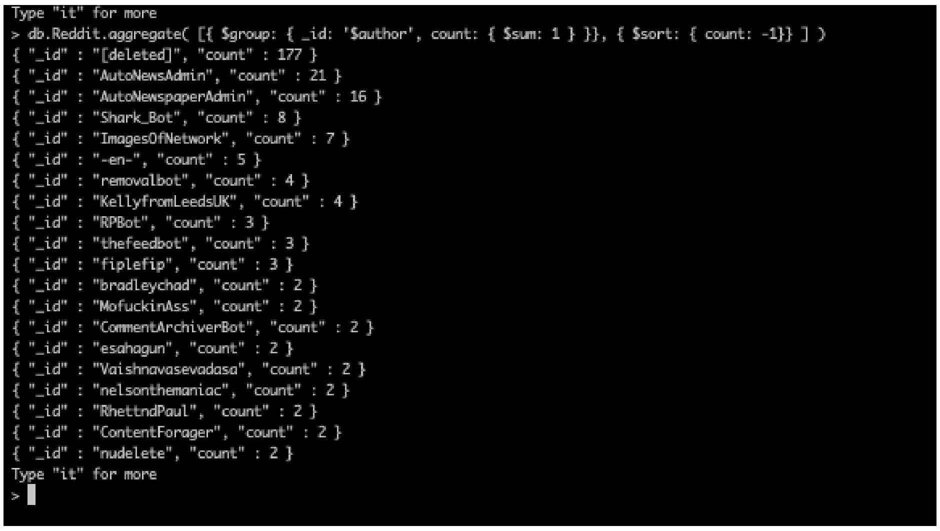
```
> db.Reddit.find( {}, { id:1, domain:1, title:1 } )
{ "_id" : ObjectId("5e4fec221d7790aa3d38eeee"), "domain" : "podify.com", "id" : "6xauxl", "title" : "app manag busi earn new stream incom know more" }
{ "_id" : ObjectId("5e4fec241d7790aa3d38eeef"), "domain" : "self.PostWorldPowers", "id" : "6xauzr", "title" : "event democrat sentinel" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef0"), "domain" : "youtu.be", "id" : "6xav1v", "title" : "assassins festiv dlc full gameplay final fantasi xv" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef4"), "domain" : "i.redd.it", "id" : "6xava0", "title" : "mackenzi pose again" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef1"), "domain" : "self.AskReddit", "id" : "6xav3x", "title" : "use eye drop eye redness wear contact lenses" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef2"), "domain" : "self.uwaterloo", "id" : "6xav5z", "title" : "exchang 4a4b delay graduation abil appli tn" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef3"), "domain" : "self.AskReddit", "id" : "6xav7y", "title" : "moviemovie scene made laugh hardest" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef5"), "domain" : "thelancet.com", "id" : "6xavc1", "title" : "associ fats carbohydrate intake cardiovascular diseas mortal 18 countri five contin result high carbohydrate intake associ higher risk total mortal wherea total fat individu ty
 relat lower total mortal" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef6"), "domain" : "self.dogs", "id" : "6xave2", "title" : "breed australian shepherd mix" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef7"), "domain" : "self.Costco", "id" : "6xavg1", "title" : "water cart crew" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef8"), "domain" : "i.redd.it", "id" : "6xavi4", "title" : "seat instal boss" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef9"), "domain" : "youtube.com", "id" : "6xavk4", "title" : "game android phone android top 5 game free download" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefa"), "domain" : "self.NoStupidQuestions", "id" : "6xavm4", "title" : "public urination defecation onc common accept" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefb"), "domain" : "self.FashionReps", "id" : "6xavo7", "title" : "w2c suprem playboy sweatpants" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefc"), "domain" : "self.MaddenUltimateTeam", "id" : "6xavq6", "title" : "moss bo" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefd"), "domain" : "self.KatowiceTrading", "id" : "6xavs8", "title" : "h ak47 redline ft 4x lgb kato 15 w 3 key" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefe"), "domain" : "self.WorldsAdrift", "id" : "6xavu9", "title" : "new bughackexploit respawning person revivers" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eeff"), "domain" : "self.Rainbow6", "id" : "6xavwd", "title" : "ubisoft actual exam game data report count mani peopl report" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef00"), "domain" : "imgur.com", "id" : "6xavye", "title" : "prepping cheeky labor day trip mexico citi" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef01"), "domain" : "self.eating_disorders", "id" : "6xaw0e", "title" : "intermitt fast" }
Type "it" for more
```

   - To return the posts associated with each author
     Query: db.Reddit.find({}, {author:1, title:1})



```
> db.Reddit.find( {}, { author:1, title:1 } )
{ "_id" : ObjectId("5e4fec221d7790aa3d38eeee"), "author" : "Podify", "title" : "app manag busi earn new stream incom know more" }
{ "_id" : ObjectId("5e4fec241d7790aa3d38eeef"), "author" : "darthholo", "title" : "event democrat sentinel" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef0"), "author" : "[deleted]", "title" : "assassins festiv dlc full gameplay final fantasi xv" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef4"), "author" : "[deleted]", "title" : "mackenzi pose again" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef1"), "author" : "[deleted]", "title" : "use eye drop eye redness wear contact lenses" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef2"), "author" : "last-term-exchange", "title" : "exchang 4a4b delay graduation abil appli tn" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef3"), "author" : "california1520", "title" : "moviemovie scene made laugh hardest" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef5"), "author" : "alan_s", "title" : "associ fats carbohydrate intake cardiovascular diseas mortal 18 countri five contin result high carbohydrate intake associ higher risk total mortal wherea total fat individu type fat relat lower total morta
l" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef6"), "author" : "ThatGuyWithaReason", "title" : "breed australian shepherd mix" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef7"), "author" : "StevenRam95", "title" : "water cart crew" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef8"), "author" : "JacobSoko", "title" : "seat instal boss" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eef9"), "author" : "GamesheGames", "title" : "game android phone android top 5 game free download" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefa"), "author" : "LeaperReign", "title" : "public urination defecation onc common accept" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefb"), "author" : "sunflowerseedbusty", "title" : "w2c suprem playboy sweatpants" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefc"), "author" : "Gpont", "title" : "moss bo" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefd"), "author" : "chinmastah485", "title" : "h ak47 redline ft 4x lgb kato 15 w 3 key" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eefe"), "author" : "Liburatus", "title" : "new bughackexploit respawning person revivers" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38eeff"), "author" : "SoranoKiseki", "title" : "ubisoft actual exam game data report count mani peopl report" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef00"), "author" : "__spice", "title" : "prepping cheeky labor day trip mexico citi" }
{ "_id" : ObjectId("5e4fec251d7790aa3d38ef01"), "author" : "coolmistmama", "title" : "intermitt fast" }
Type "it" for more
>
```

- To return the number of posts by each author in descending order
  Query: db.Reddit.aggregate([{$group:{_id:'$author', count: { $sum: 1}}}, { $sort: { count: -1}}])



```
Type "it" for more
> db.Reddit.aggregate( [{ $group: { _id: '$author', count: { $sum: 1 } }}, { $sort: { count: -1}} ] )
{ "_id" : "[deleted]", "count" : 177 }
{ "_id" : "AutoNewsAdmin", "count" : 21 }
{ "_id" : "AutoNewspaperAdmin", "count" : 16 }
{ "_id" : "Shark_Bot", "count" : 8 }
{ "_id" : "ImagesOfNetwork", "count" : 7 }
{ "_id" : "-en-", "count" : 5 }
{ "_id" : "removalbot", "count" : 4 }
{ "_id" : "KellyfromLeedsUK", "count" : 4 }
{ "_id" : "RPBot", "count" : 3 }
{ "_id" : "thefeedbot", "count" : 3 }
{ "_id" : "fiplefip", "count" : 3 }
{ "_id" : "bradleychad", "count" : 2 }
{ "_id" : "MofuckinAss", "count" : 2 }
{ "_id" : "CommentArchiverBot", "count" : 2 }
{ "_id" : "esahagun", "count" : 2 }
{ "_id" : "Vaishnavasevadasa", "count" : 2 }
{ "_id" : "nelsonthemaniac", "count" : 2 }
{ "_id" : "RhettndPaul", "count" : 2 }
{ "_id" : "ContentForager", "count" : 2 }
{ "_id" : "nudelete", "count" : 2 }
Type "it" for more
>
```

6. Modifying the structure of the database to add new subreddits per post:
   - Is there a need to alter previous records stored in order to apply the new proposed structure:
     MySQL – Yes, In this case of having many subreddits per post, we should have a 1 to many relationship between the table posts and reddits. Therefore, we would have to edit the structure wherein the sub reddit id acts as a foreign key in the table posts.
     We can also, alternatively create another table called PostToSubreddits and put up post id and subreddit id as the primary and foreign keys.
     MongoDB – Yes, we need to have an array storing subreddit id per post.

   - Are the previous queries still working?
     MySQL – No, since we edited our foreign key to subreddit id in posts, we would need to alter the join query so that it works for subreddits
     MongoDB – No, since the field has changed from a single value to an array of values.

- ## Reflection

1. Suitability of CLI for some big data tasks:

   CLI stands for Command Line Interface, it is a program that allows users to type text commands intructing the computer to do specific tasks. CLI has been a tool that comes from the 70s and one can say that CLI was the first tool for an interactive communication between a human and a machine.
   Often people use Hadoop and other Big Data tools for data processing and analysis tasks, but these tasks can be done using simpler and faster tools like CLI because of it's variety of benefits like:
   - CLI is an extremely stable environment i.e creating a data pipeline out of shell commands means that all processing steps can be done in parallel. One can easily construct a stream processing pipeline with basic commands that will have great performance when compared to many modern Big Data tools.
   - Using CLI is simple, agile and highly modular by design
   - CLI is interactive. It is an REPL (read-eval-print loop) environment, as opposed to the edit-compile-run-debug systems. And this makes it an ideal tool for data exploration and for quickly designing operations and building data flows
   - CLI integrates really well with other programming languages and technology
   - CLI is closer to the file system, hence closer to our data, this means it is much faster than big data technologies like Hadoop because of it's simplicity, speed and a reduced line of communication. Streaming analysis requires no memory all.
   - NoSQL database systems like MongoDB can be used in a Command Line Interface which use a document oriented system instead of a relational database management system. Whereas, modern Big data tools make use of a relational database management system.

   Hence, we can conclude that simpler tools like CLI can be used to perform Big Data tasks giving faster results. However, tools like Hadoop can be used for huge amount of data or when distributed processing is required.

2. The following table compares RDBMS and NoSQL and gives instances on which is better suited in different cases:

RDBMS	NoSQL
RDMS database is a relational database and is standard language for relational database management systems. The relations among tables are stored in the form of a table. SQL is used in an RDBMS to perform tasks on a given database. Examples of RDBMS include Oracle, Access, Microsoft SQL Server etc	NoSQL means "Not Only SQL". Unstructured, schema less data can be put together in multiple collections and nodes and it does not require fixed table schemas. Examples of NoSQL include CouchDB, mongoDB and RavenDB
1. RDBMS has ACID properties of a transaction i.e atomicity, consistency, isolation and durability	NoSQL does not contain ACID properties but is based on the CAP theorem i.e consistency, availability and partition tolerance.

2. SQL databases are table based containing schemas and constraints. SQL databases because of its structure could be a good option for applications requiring multi row transcations.	NoSQL database is schema less so data can be inserted in a NoSQL database without any predefined schema i.e it can be changed anytime.
3. Scalability in a relational database is expensive and difficult to handle. It is based on vertical scaling. A relational database needs to be distributed on multiple servers if it needs to be scaled. This is difficult to manage.	NoSQL is hugely scalable because of horizontal scaling that means scaling by adding more machines into the pool of your resources. Each node is independent and self sufficient i.e there is no single point of contention across the system
4. SQL has more complexity in terms of fitting data into tables. If the data doesn't fit into tables, database structure will be more complex and difficult to handle. 5. RDBMS databases are expensive as they make use of larger servers and storage systems.	NoSQL on the other hand is more flexible, easier to replicate, easier to partition and join free. NoSQL databases are cheap and opensource with easier implementations and cheaper servers

We conclude that using NoSQL databases has a variety of benefits over using RDBMS databases. However, GUI mode tools to access NoSQL databases are harder to find in the market while readily available for RDBMS databases.
However, when we talk about applications requiring multi-row transactions, SQL databases score better.

3. Reaserch Report:

Bigtable: A distributed storage system for structured data

Motivation for the research: Problems faced

We know that data processing and storage in Google grew at a very large size in petabytes scale since the past twenty years. Since Google's primary business focussed mainly on web search, the ability to quickly access huge amounts of data distributed across a wide array of servers became vital for its functioning. Also, as Google grew over the years, most of it's applications required a rapid reaction time which required the database to achieve low latency.

Solving the problem using the Google Bigtable

In order to fit the data storage demand of Google services and solve the problems mentioned above, the author's team implemented and deployed Bigtable, a NoSQL database for managing structured data at Google.

In this white paper, the authors describe the simple data model, design, implementation and refinement of Google's Bigtable database.

Bigtable is a distributed storage system for managing structured data that is designed to scale to a very large size: petabytes of data across thousands of commodity servers.

Due to its wide applicability, scalability, high performance and high availability, Bigtable is used by more than sixty Google products and projects, including Google Earth, Google Finance, Analytics as well as Personalized search.

As the authors note, Bigtable shares many characteristics with database systems, with important differences. For example, it does not support a full relational data model, rather opting for a simple data model that gives users control over data layout and format, as well as work with the data locality in the underlying storage.

Outlining the Data Model of Bigtable

The authors describe a Bigtable as a "sparse, distributed, persistent multi-dimensional sorted map". The map is indexed by a row key, column key, and a timestamp; each value in the map is an uninterpreted array of bytes.

The authors came up with this model after examining a variety of potential uses of a Bigtable like system.

Rows: The row keys in Bigtable are stored in string format, allowing rows to be sorted in lexicographical (or alphabetical) order which helps the overall efficiency of the system especially in regard to load distribution. Since Bigtable is widely used in web-based search applications, this design allows information from the same domain to be stored close to each other in the database, with the URLs being used as row keys.

A range of rows in Bigtable is known as a "tablet." Bigtable tablets are stored on the same server, to enhance overall performance and load distribution.

Columns: Column keys in Bigtable are grouped together as column families. Usually data within a column family share the same data type. Google uses column families in their Webtable implementation of Bigtable to store all the anchors that refer to a web page. Hence, this design makes reads and writes more efficient in a distributed environment.

Timestamp: Each cell in a Bigtable can contain multiple versions of the data indexed by time with the help of a Timestamp. The Webtable stores the time Google crawled a specific URL in its timestamp index.

API: Bigtable is accessed by an API that allows for creating and deleting tables and column families, as well as other functionality such as modifying metadata and access control rights. Bigtable is also compatible with MapReduce and it is built on top of the distributed storage system GFS.

Scientific discussions: Structure & Implementation of Bigtable

Bigtable is built on several other basic component from google.

1. Bigtable adopts Google File System to storage logs and data format files. It depends on a cluster management system for scheduling jobs, managing resources on shared machines, dealing with machine

failures, and monitoring machine status.
2. Bigtable is also built upon the Google SSTable which is a persistent, ordered immutable map from keys to values.
3. A root tablet is stored in Chubby which contains a METADATA table with the location of all the other tablets. Chubby keeps track of all of the tablet servers and when a server starts, it acquires a lock on some data. The persistent state of the data system is stored in GFS and updates to the data are committed to a commit log.
Bigtable uses Chubby for ensuring that there is at most one active master at any time, storing the bootstrap location of Bigtable data, discovering tablet servers and finalizing tablet server deaths, storing Bigtable schema information and access control lists.

Improvements

To ensure the high performance, availability, and reliability, there are a couple of optimizations implemented in Bigtable. These include locality groups, compression over locality groups, caching for read performance, Bloom filters, commit-log implementation, speeding up tablet recovery and exploiting immutability.

Strengths & Contributions of the paper

The main strength of this paper is that it introduces a successful implementation of a distributed data management system that is very scalable and reliable. This is well explained by its widespread applications across most of the Google's services, like Google Analytics, Personalized Search, Google Finance, Google Earth and many more.
It is well able to handle such a wide array of requirements and workloads which adds to the adaptability of the Bigtable system.
Being able to implement and deploy a system like Bigtable on such a large scale is definitely a great achievement for any company.
A few weaknesses and areas of improvements of this paper include Bigtable's speedup when it comes to concurrent execution across multiple servers. By increasing the number of servers, we are impacting the random read performance by a huge factor which results in other benchmarks showing less favourable results which eventually lead to loss of performace. The paper also throws light on factors which affect the performance such as suboptimal load balancing. A downside of Bigtable itself is that it does not provide the consistency of a traditional RDBMS, or a recognizable query language. Additionally, many optimization decisions seem to be left up to the programmer. This makes it significantly harder to adopt than other systems.

Other related work

The Boxwood project has components that overlap in some ways with Chubby, GFS, and Bigtable, since it provides for distributed agreement, locking, distributed chunk storage, and distributed B-tree storage.
Recent projects like CAN, Chord, Tapestry and Pastry have tackled the problem of providing distributed storage or higher-level services over wide area networks, often at "Internet scale."
Projects like C-Store and commercial products such as Sybase IQ, SenSage, KDB+ are other examples of other related works similar to the Bigtable.

Conclusions & Thoughts

I enjoyed reading this paper because it introduces the design motivations and implementations very well and clearly.
Even though this paper was pretty long, the authors have humbly described their works in designing and implementing such a vast distributed system like the Bigtable.
This paper also links to other studies on GFS, MapReduce and C-Store which are highly relevant to our module course content.
For me, personally I felt that the authors focussed very well on the problem they were trying to solve with Bigtable which resulted in Bigtable being one of the largest NoSQL subsets known as Bigtable databases. Google's influence in technology has gone far beyond search algorithms, search advertisements and mobile operating

systems and with examples like Bigtable, Chubby, GFS, MapReduce, we can already see their impact.
I would also like this paper better if it gave more details on how Bigtable interacts with MapReduce. The paper provides a brief idea, and it's interesting to learn more about it relevant to our course.

• Conclusion

Through this assignment, I got a better understanding of how CLI and bash can be used to clean datasets, for database management and what are the practical benefits of using CLI over the modern big data tools.
The research paper on Bigtable also gave me a very good understanding of the a large scale implementation of NoSQL and the amazing improvements it can give in terms of performace and scalability.

• References

1. https://www.quantmetry.com/data-science-at-the-command-line/
2. https://adamdrake.com/command-line-tools-can-be-235x-faster-than-your-hadoop-cluster.html
3. https://www.loginradius.com/engineering/relational-database-management-system-rdbms-vs-nosql/
4. https://huu.la/blog/review-of-bigtable-a-distributed-storage-system-for-structured-data
5. https://static.googleusercontent.com/media/research.google.com/en//archive/bigtable-osdi06.pdf
6. https://www.cs.rochester.edu/courses/261/spring2017/termpaper/16/paper.pdf