

6CS030 – BIG DATA

GROUP ASSESSMENT

Table of Content

# 1. Introduction to Big Data

Gigantic data is an emerging state where immense volume of data is conveyed ordinarily as both coordinated, semi-coordinated and unstructured, which will make an issue for a standard methodology and informational index to manage it. An approach to manage trained decisions subject to correct methodology depicts any data arrangement, and is sufficiently wide to use critical level programming ability and methodologies to change the data into an affiliation's asset. The three DBMS open-source systems and the Oracle, MongoDB and Apache Hadoop programming tools have so far been used to solve problems in a wide range of datasets (Juneja & Das, 2019).

Such colossal information comes from a few distinctive sources with intricacies known as 5Vs i.e.

1. Volume: This is normal for the enormous enlightening lists made at high repeat rates.
2. Variety This applies to different types of documents, i.e., ordered, semi-structured, unstructured and everything.
3. Velocity: This investigates how effectively and consistently information can be made from a solicitation.
4. Veracity: This tends to the exactness, precision, and if credibility of the information.
5. Value: This includes the importance of data from various raw data that are used. It is not feasible merely because of the abundance of information to extract utility from it.

# 2. Introduction to dataset

We have three different datasets for each database which are somehow similar to death. We have taken “suicide rates overview 1985-2016” dataset from other datasets (Human Development Reports, 2020) (World Bank, 2018) and (WHO, 2018) linked by time and place to identify a corresponding signal to raise suicide rates in the various cohorts globally. Another is “Novel corona virus 2019” and last we have dataset for leading cause of death in USA.

## 2.1 Justification for choice

We have chosen the death mortality rate in those three datasets in different way like from the suicide, corona pandemic and leading cause of death in USA. We will do analysis and visualization of those datasets, and figure out the rate of suicide and its key factor for it, for corona virus we will try to visualize and do analysis and at last for Cardiovascular diseases we will try to predict heart failure. Suicide datasets can be found [1][1][1], corona dataset on Johns Hopkins GitHub and leading cause from data.gov.

## 2.2 CSV Dataset

We have suicide csv dataset for the oracle and Hadoop. suicide dataset consists of more than 12 columns, 1000 of records and also leading death dataset has 6 columns and 1000 of record, which will be reduced while cleaning the dataset as required for the project.

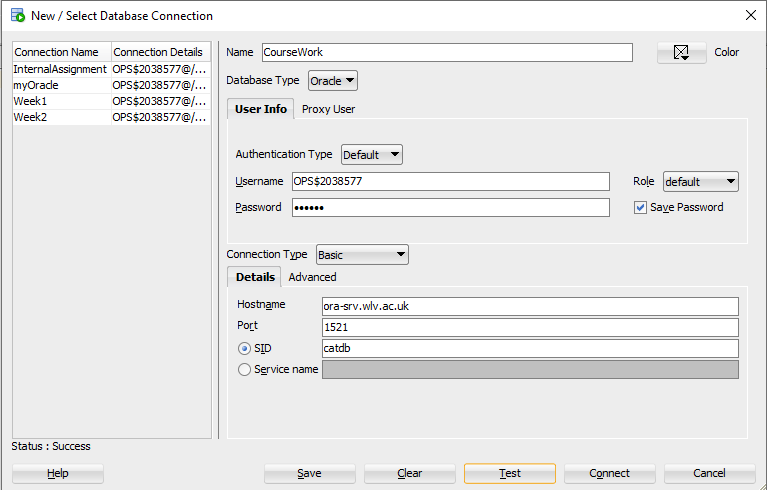
## 2.3 JSON Dataset

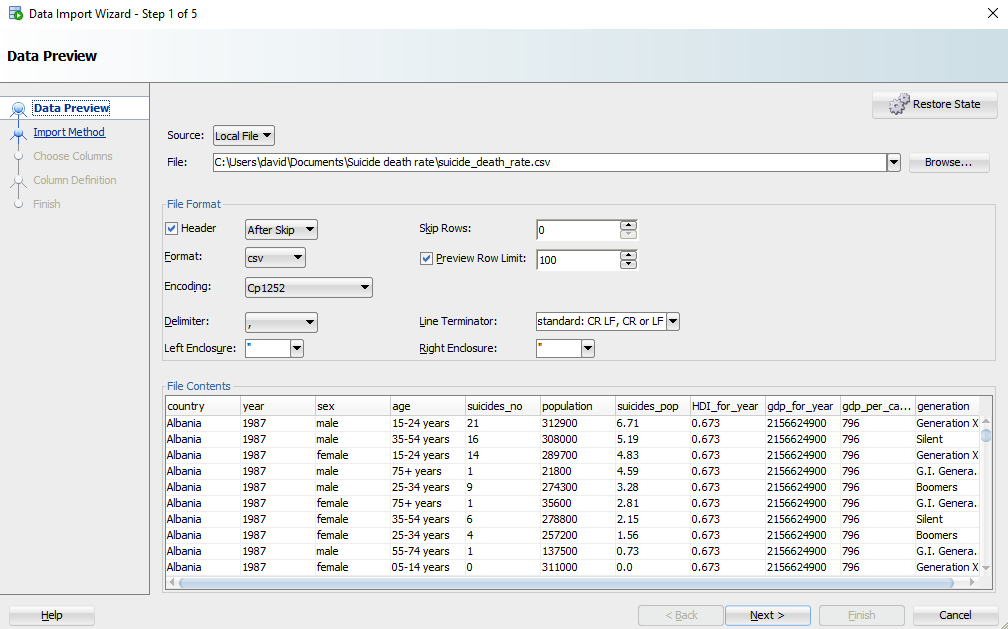
For mongo we have json format dataset of corona virus. It has more then 10000 of record of corona virus of every country that corona virus had affected. We will be analyzing and visualize those datasets according to the requirement. We may reduce or remove some records at time of cleaning the dataset according to its requirements.

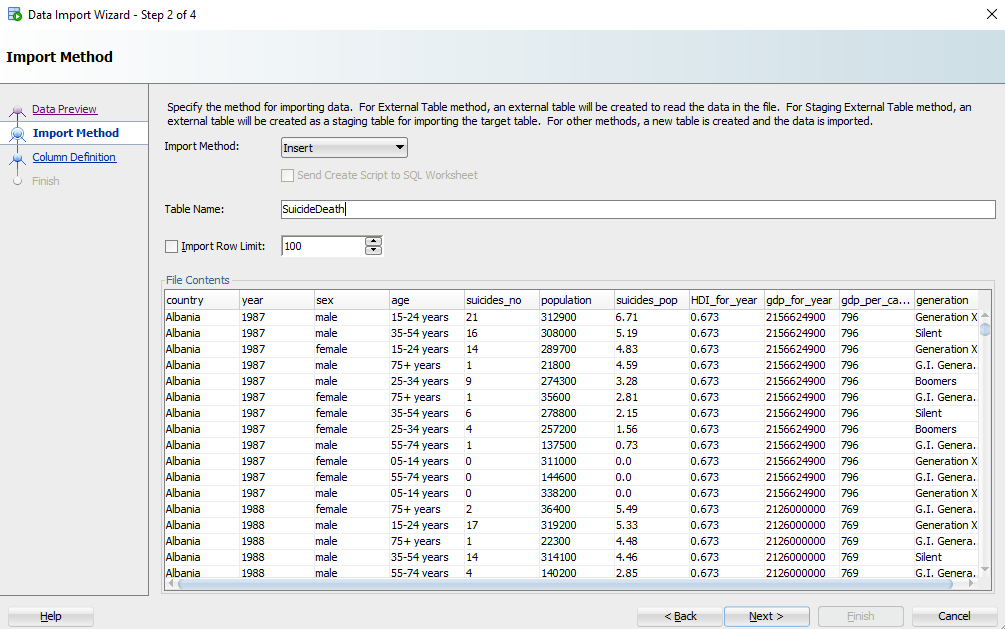
# Import/Cleaning of Data

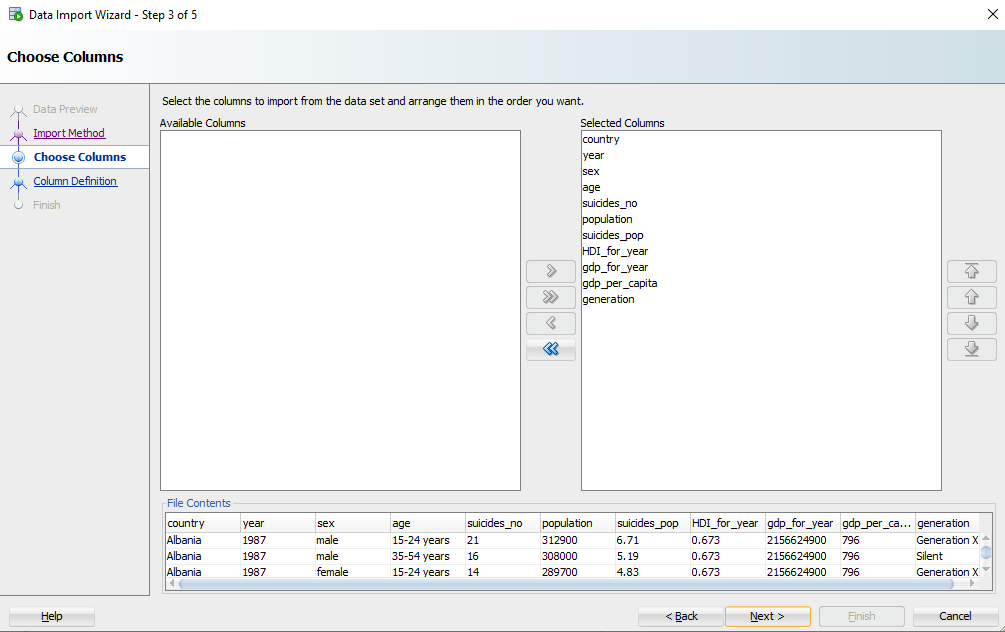
*Note: Cleaning of data is in Appendix for all three datasets*

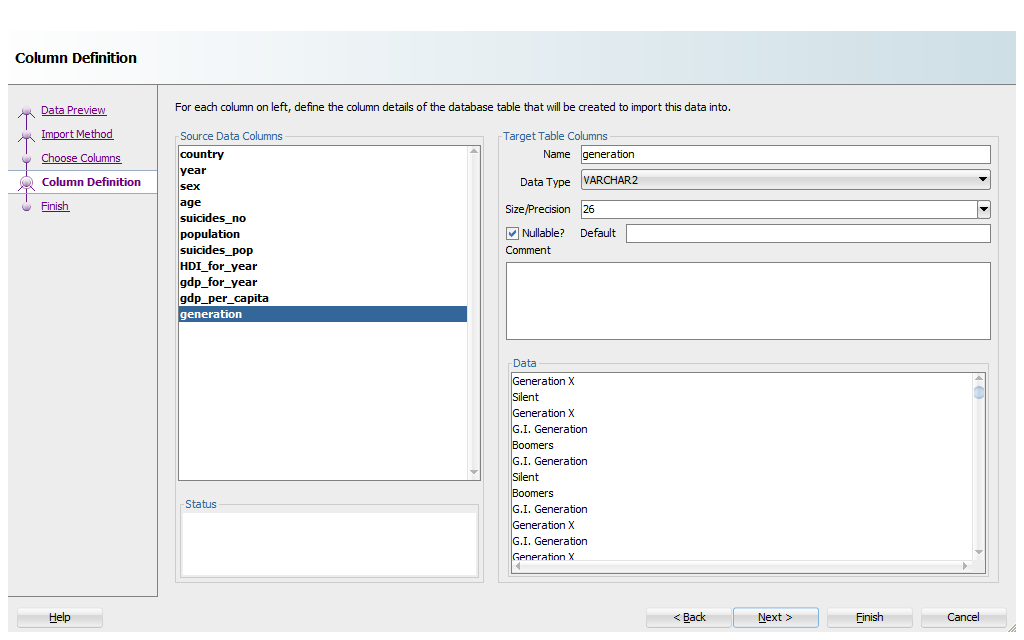
Importing CSV data:

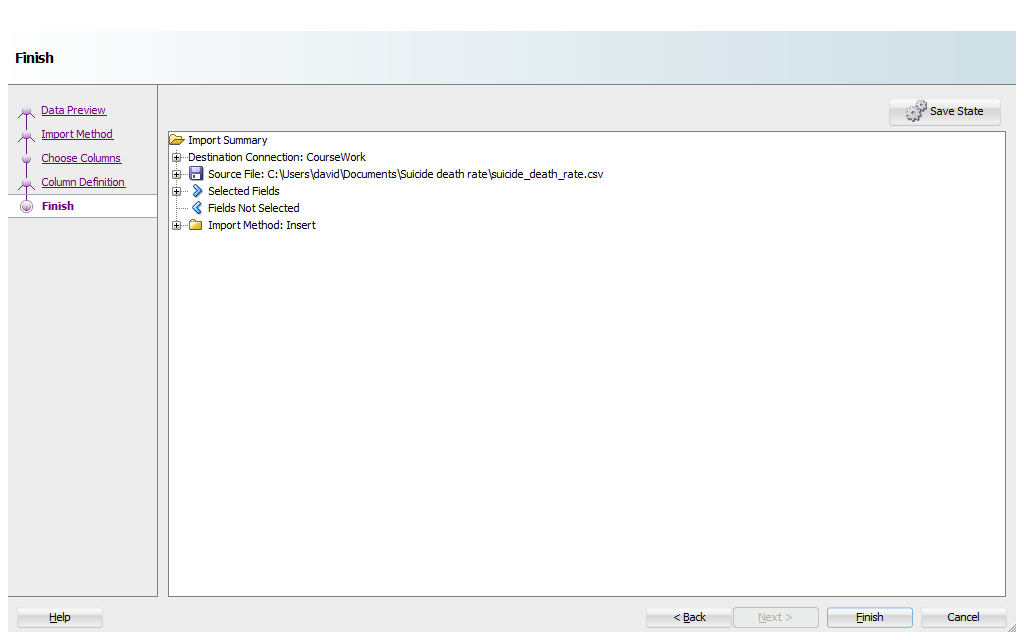
**Creating new connection in oracle:**

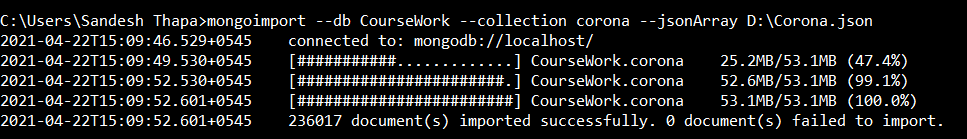
Step 1: Browsing location of CSV file.

Step 2: Creating new table called SuicideDeath which will store all the csv file data.

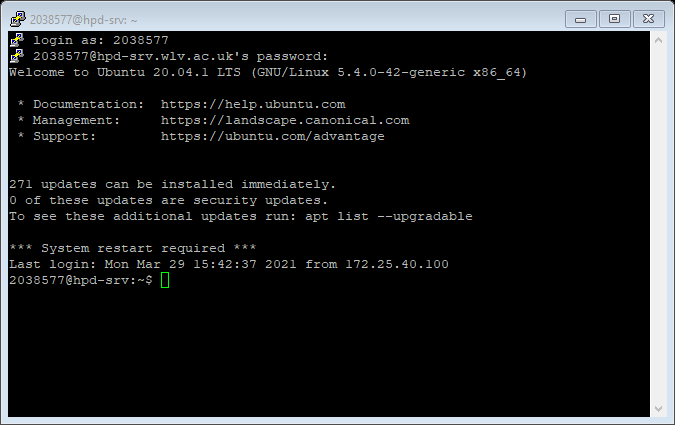
Step 3: Selecting column which is necessary

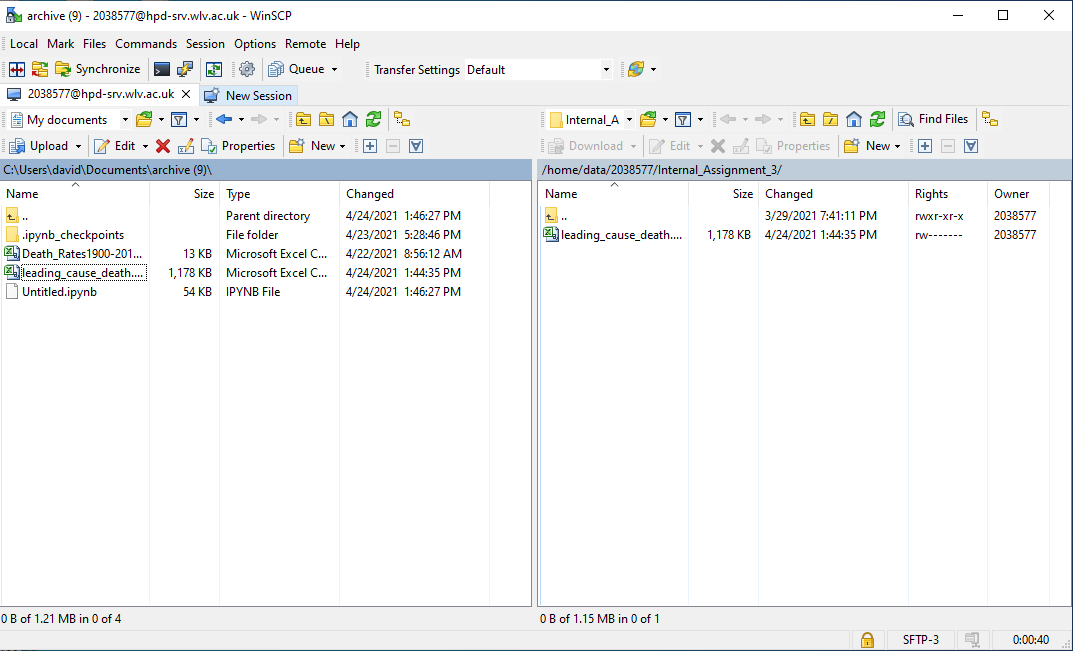
Step 4: Selecting appropriate datatype for each column.

Step 5: Finishing the import:

Importing JSON

**Importing CSV for Hadoop**

****Step 1: Login Using username and Password in putty

Step 2: Upload CSV file in winscp

Step 3: Showing uploading files in terminal

**Appropriateness of dataset in each database**

The findings showed that oracle performs best in csv format than json when the data is zero for rows. Csv is avoided in standard SQL paging routing to achieve quicker results and produce efficiency. For mongo it was seen that json format of data outperformed the csv data because it is complicated for reading and verbose. In Hadoop we need dataset where we can split and encode any data so csv is used because the text of characters are splitable.

# Analysis of the Data and visualizations

*Note: Analysis is done in appendix due to word limit*

**Data Visualization**

**Oracle**

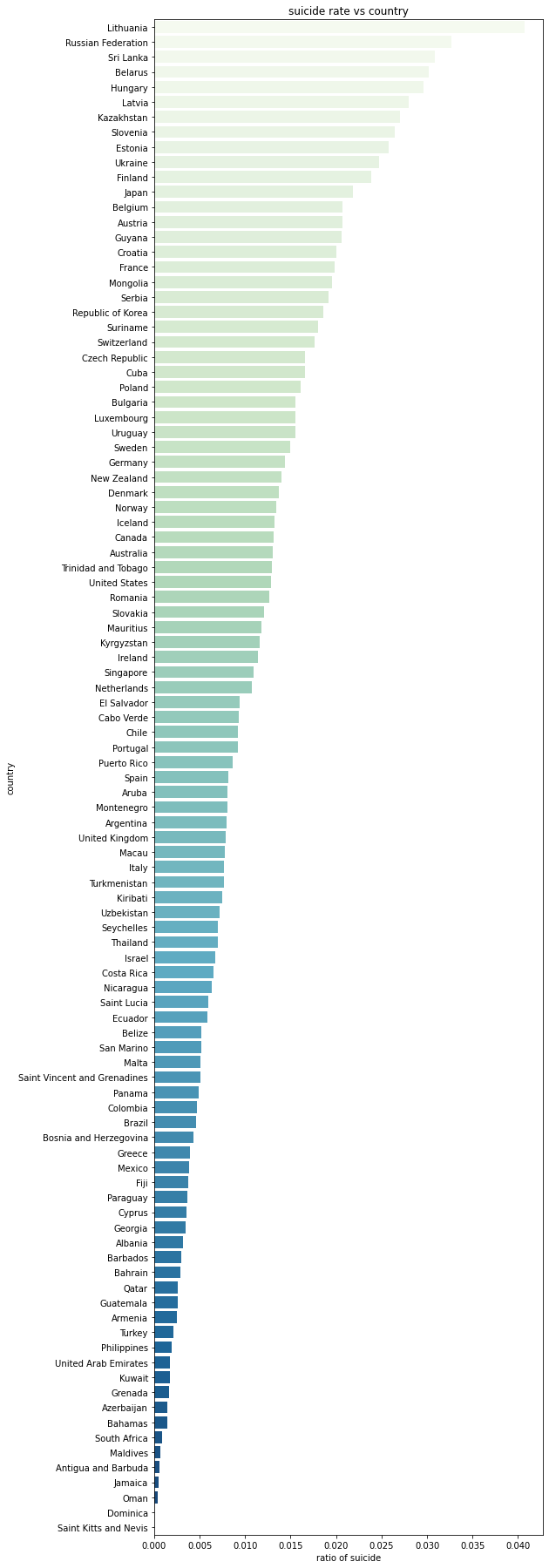
****

Figure 1 Suicide rate of every country

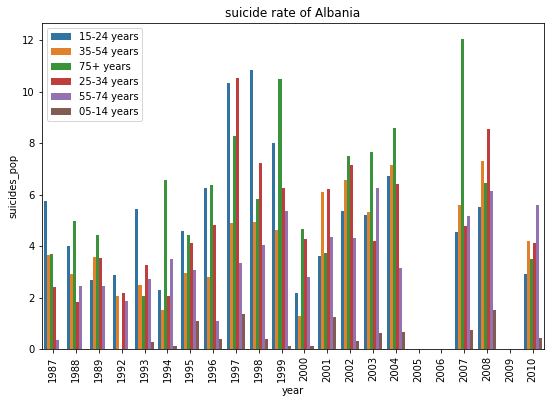
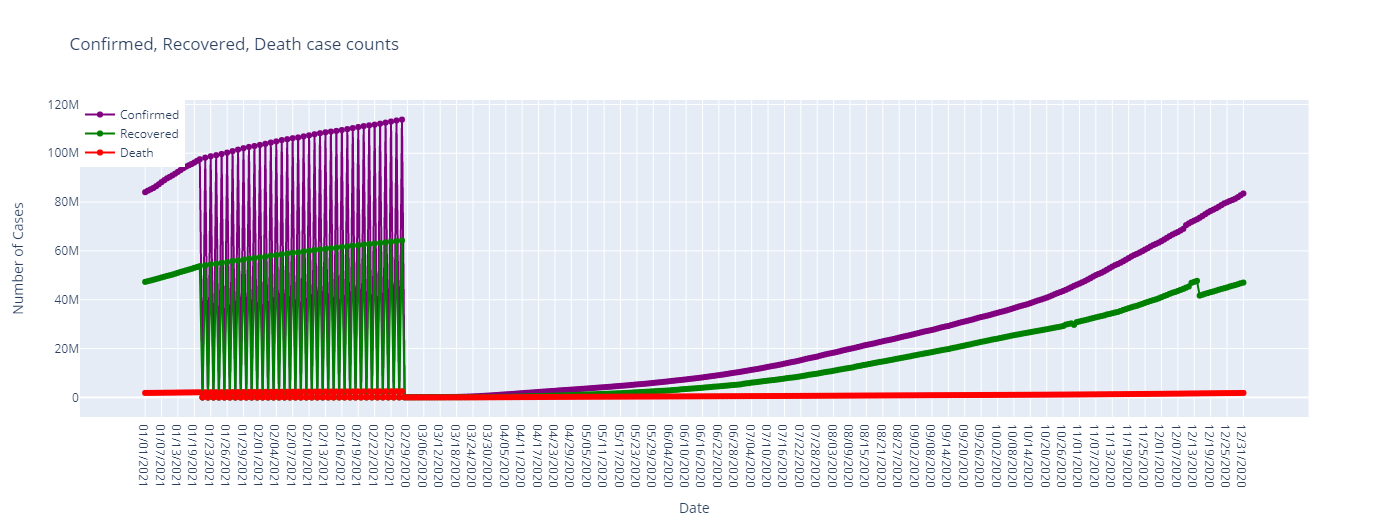
 The above visualization shows the suicide rate on average of each country from lowest rate Saint Kitts and Nevis to highest Lithuania.

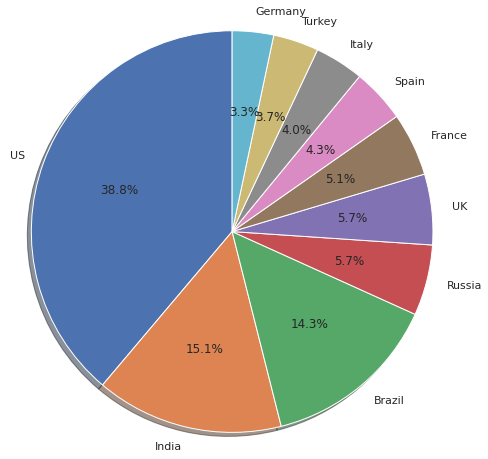
Figure 2 Suicide Rate of Albania

From the above visualization figure, we can see the suicide rate of Albania between different age group. If we see suicide rate between 1987 to 1996 age group of 15-24 have highest death rate and followed by 35-54 and 75 age group in second place. After 1966 we can see the peak in suicide rate from 1997 to 1999. In those time period 15-24, 25-34 have highest suicide rate.

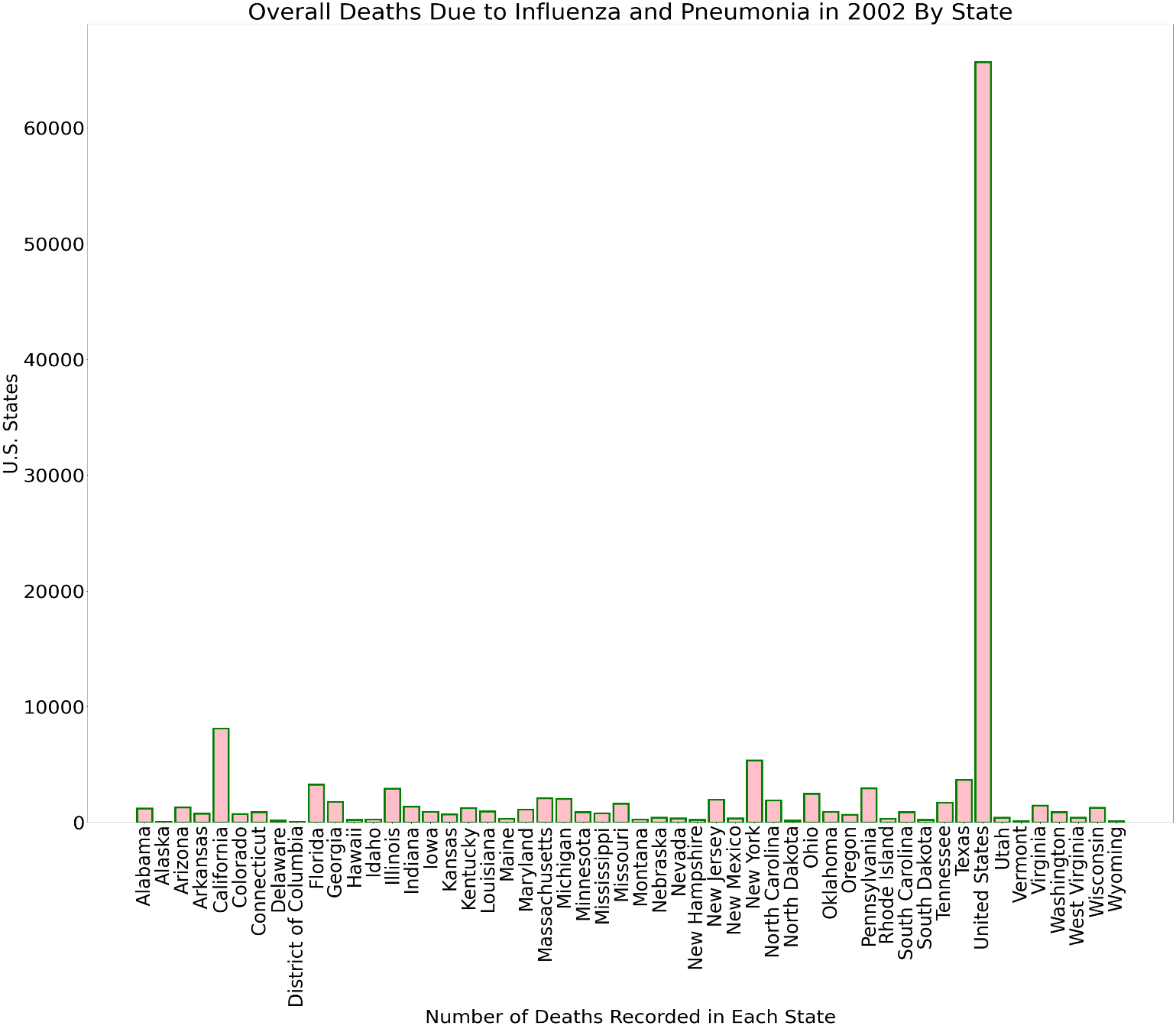
The prevalence of suicide is linked to many factors such as national per capita. We found that the suicide rate is strongly associated with GDP in the world in certain nations, showing a certain degree of decrease throughout the suicide rate in the degree of national income.

Mongo

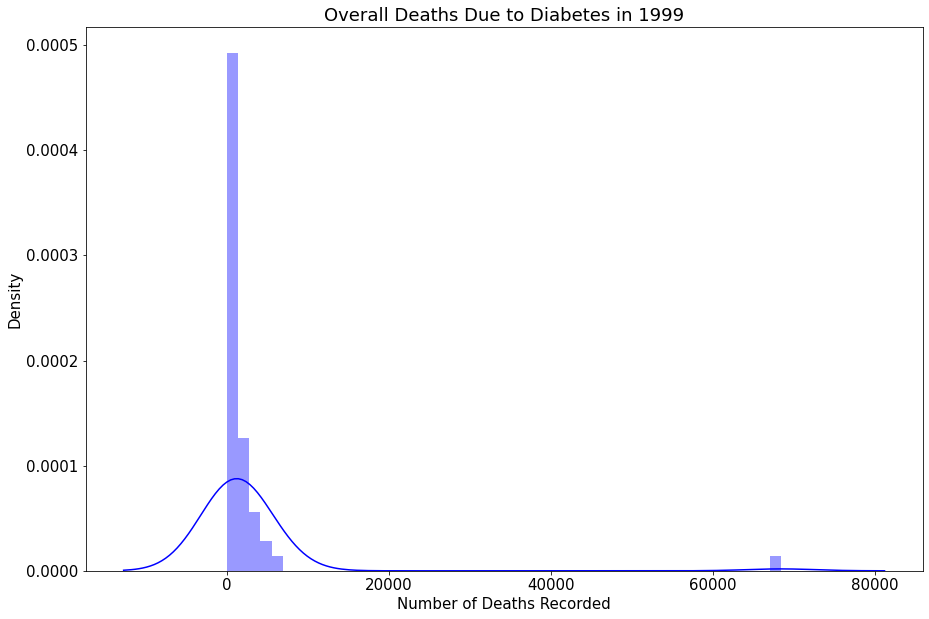
*Note: Figure date Starts from 03/06/2020*

Above figure shows that at when corona was first confimred there was more 84 thousand confimred case with 36 thousand recovered and 2872 deaths per day worldwide. Those confimred case keep increasing by the end of 2020 and reach upto 83.52186M confirmed case and death of more then 1.8125578M of people and it is still increasing.

From this pie chart we can see that US was badly affected by the corona virus. Out of 100% corona virus US had 38.8% of virus following closely to each other by India 15.1% and 14.3%. Those three counties almost carried the more then half of the world corona virus.

Hadoop

In this bar chart, the number of deaths reported in 2002 by nation, and in the US as a whole, as a result of flu and pneumonia complications is seen.



From the above histogram, it shows that number of death records in 1999 from diabetes-realted complications

# Comparison

**Advantage of using three different database management system**

## Oracle

* Versatility:

Oracle can operate on any kind of operating system, giving users a lot of versatility.

* Functionality:

It coordinates undeniable level innovations with cutting edge undertaking arrangements that can be utilized to oversee enormous volumes of information in practically all venture applications.

* Reliability:

Oracle is a trustworthy data set framework and can likewise be a data framework that performs well when extraction errands are allotted to it.

* Data Recovery:

Oracle has a feature called Flashback that helps you to retrieve data that has been removed or destroyed, which speeds up data recovery and simplifies management.

* Performance:

Oracle provides techniques such as using several servers to figure out constant information with actual Application cluster function that can increase computing speed (Malavika, 2019).

## Mongo

* Versatility:

It provides tremendous flexibility in the collection and access to data types, since it is a schema less database.

* Accessibility:

MongoDB has many community drivers to serve in lesser-known programming languages all main programming languages.

* Speed:

Data access and query is very speedy since the RAM is used for storing data.

* Cost

It's highly inexpensive because it eliminates data storage and hardware costs.

* Scalability

It is really convenient to scale the database. It offers the chance to sharpen horizontally (DataFlair, 2019).

## Hadoop

* Flexibility:

In better places, Hadoop can be utilized for methods like information stockpiling, extortion recognizable proof, and so on.

* Speed:

The methods used are mostly located on the same computer where the data processes are done very quickly.

* Scalability:

The computing platforms are very versatile, and they can save and provide massive volumes of data over different parallel servers.

* Safer

In case of a malfunction, information is copied to different hubs with the end goal that another duplicate is prepared for the clients.

* Cost effective

It is rendered as an infrastructure for scale-outs to hold all the big data of businesses and can be later used for computation and storage (Stevepaul, 2015).

**Disadvantage of using three different database management system.**

## Oracle

* The machine identity of Oracle is limited to 8.
* Oracle has a special network setup to function correctly.
* It lacks looping functions like and which loops so that remedial processing is not carried out.
* Contrasted with other SQLs, the expense of oracle is extremely high.
* Oracle does not provide integrated functions for data cleaning. For the cleaning of the data resources of third parties are therefore used (Singh, 2020).

## MongoDB

* Joins not Supported:

Like a relationship data set MongoDB doesn't uphold join. In any case, you can utilize the associations highlight by physically coding it. Be that as it may, execution can back off and results can change.

* High Memory Usage:

For each pair of values, MongoDB stores key names. There is also no data redundancy due to no link capability. This leads to inefficient memory consumption.

* Limited Data Size:

It does not support document size more than 16mb.

* Limited Nesting:

There is a limitation that we cannot perform nesting for more than 100 levels (DataFlair, 2019).

## Hadoop

* Security Concerns:

It can be a challenge to only manage complicated systems like Hadoop. In the Hadoop protection model, a basic example can be used, which is disabled due to the complexity. If the manager of the network does not know how to do so, the data might be at enormous risk and it lacks the encryption.

* Vulnerable by Nature:

With respect to defense, Hadoop's own make-up puts it at risk. The code is almost exclusively written in Java, one of the most commonly used and divisive programs. Java has been extensively abused and has also been embroiled in several abuses of confidentiality.

* Not fit for small Data

Albeit huge information isn't delivered carefully for enormous associations, not all large information frameworks are proper for little information necessities. Unfortunately, it happens that Hadoop is one of them. The Hadoop Disseminated Document Framework does not have the capacity to viably oblige the change of little records because of its high-volume nature. Thus, restricted information volumes are not suggested for organizations.

* Potential Stability Issues

Hadoop has a good extent of strength issues like all open-Source applications. Organizations must ensure that they run the new stable update or run it under a third-Party Provider who is prepared to deal with those issues in order to prevent these problems.

* General limitations

The article presents as potential alternatives Apache Flume, MillWheel and Google's cloud dataflow. Both of the platforms have in common an opportunity to increase data storage, aggregate and integration quality and reliability. The key argument in the essay is that businesses will skip great advantages when using Hadoop alone [6].

# Conclusion and Recommendations

So, in a nutshell we had gone through the practical work on all three databases, it was challenging and interesting. When we operate on these databases, we know that they process vast volumes of data, which makes these databases an irreplaceable part of the big data world. We may use the sql command to view the data in oracles. Because MongoDB is a NoSQL, the document uses json and schema, but supports all csv and json formats. It is also supported for Hadoop.

Different queries were carried out at the time of doing coursework on these databases and we can see clear visualization of data in the form of diagram. Therefore, to maintain accuracy we should feed the data of complete error free by cleaning it. These will lead to much faster and more accurate data needed. Also, while selecting the dataset the user should know the dataset properly, as a result user can take maximum advantage of dataset while analyzing, running quires and visualizing the data.

**Appendix**

## Oracle

Oracle database was made by oracle corporation in 1997, as a connected data set administration framework. It is a multi-model social information base administration framework, especially for the calculation of organization network and information stockpiling. It is one of the principal alternatives for practical arrangements and information stockpiling for organizations. It upholds SQL for communicating with the data set as a query language (Educba, 2020).

## MongoDB

MongoDB is a NoSQL information base for the report arranged capacity of high-volume documents. MongoDB utilizes records and records as opposed to utilizing tables and lines as in standard social data sets. Reports are comprised of essential worth combines that are a crucial MongoDB information unit. Assortments incorporate records and capabilities that are equivalent to connection information base tables. MongoDB is a mid-2000s site that has materialized (Guru99, 2021).

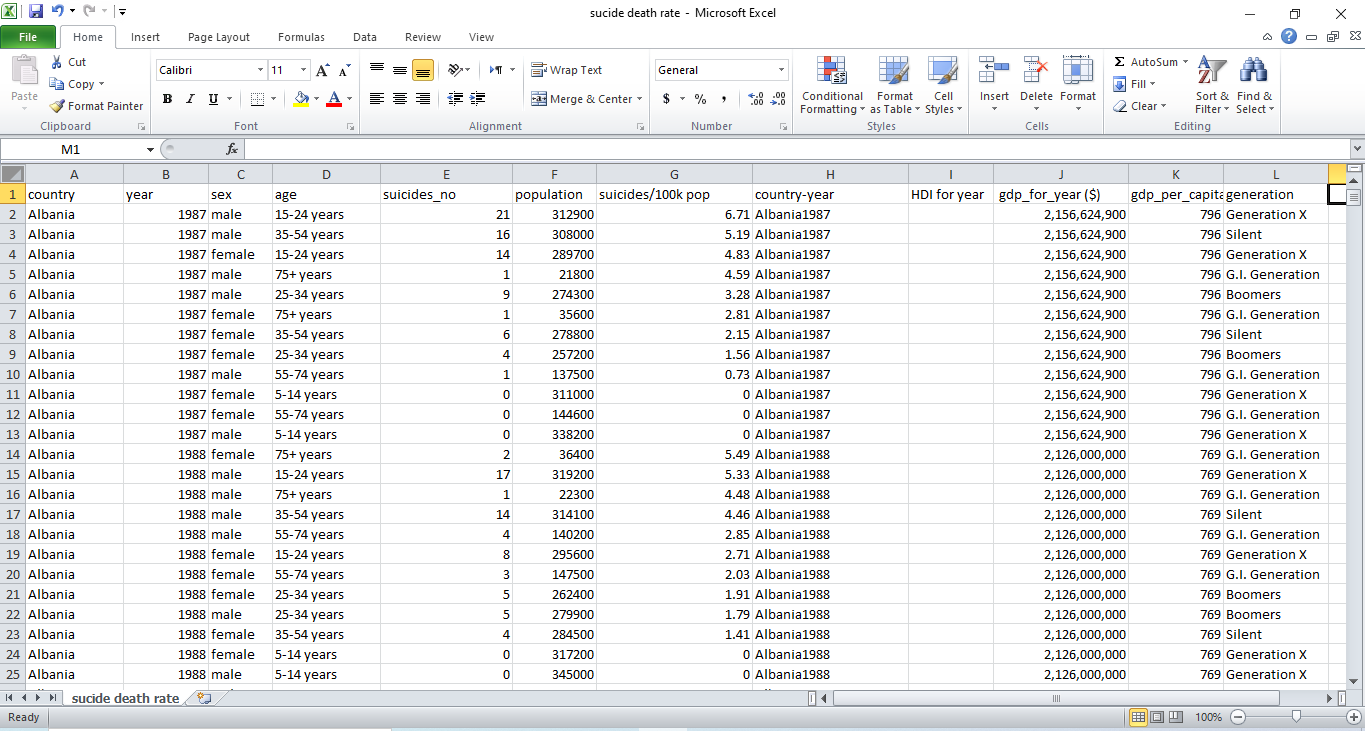
## Hadoop

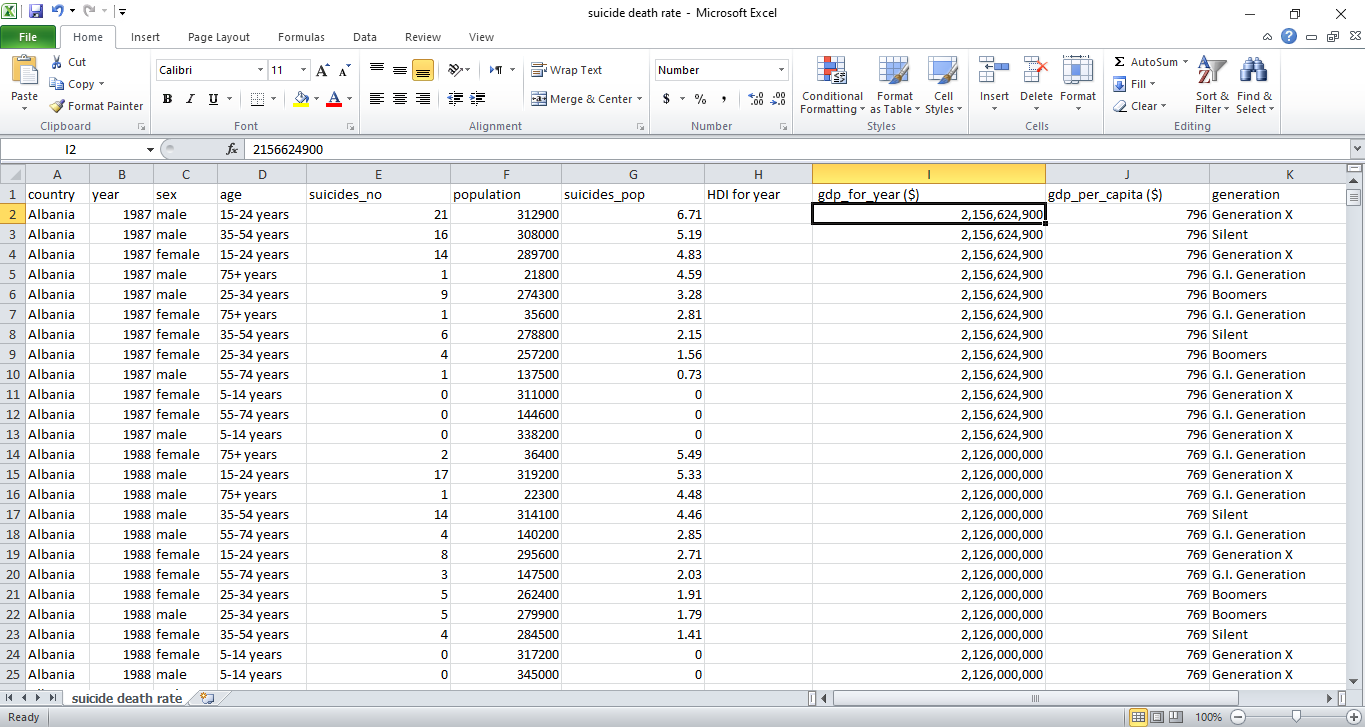
Hadoop is a software open-source architecture for data storage and operation on commodity clusters. It has vast data capacity, immense computing power and the capacity to perform almost unlimited simultaneous tasks or tasks.

## Cleaning of data

Oracle

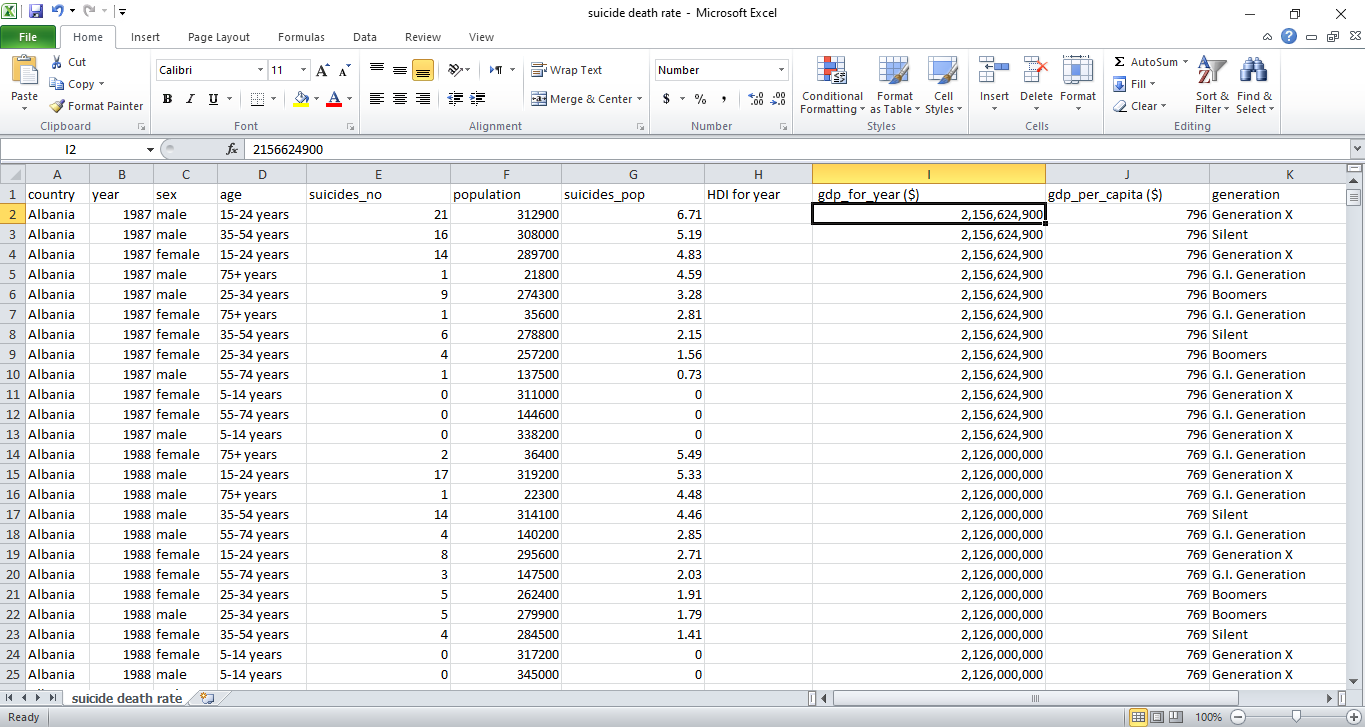
**Step 1: Dropping Unnecessary columns**

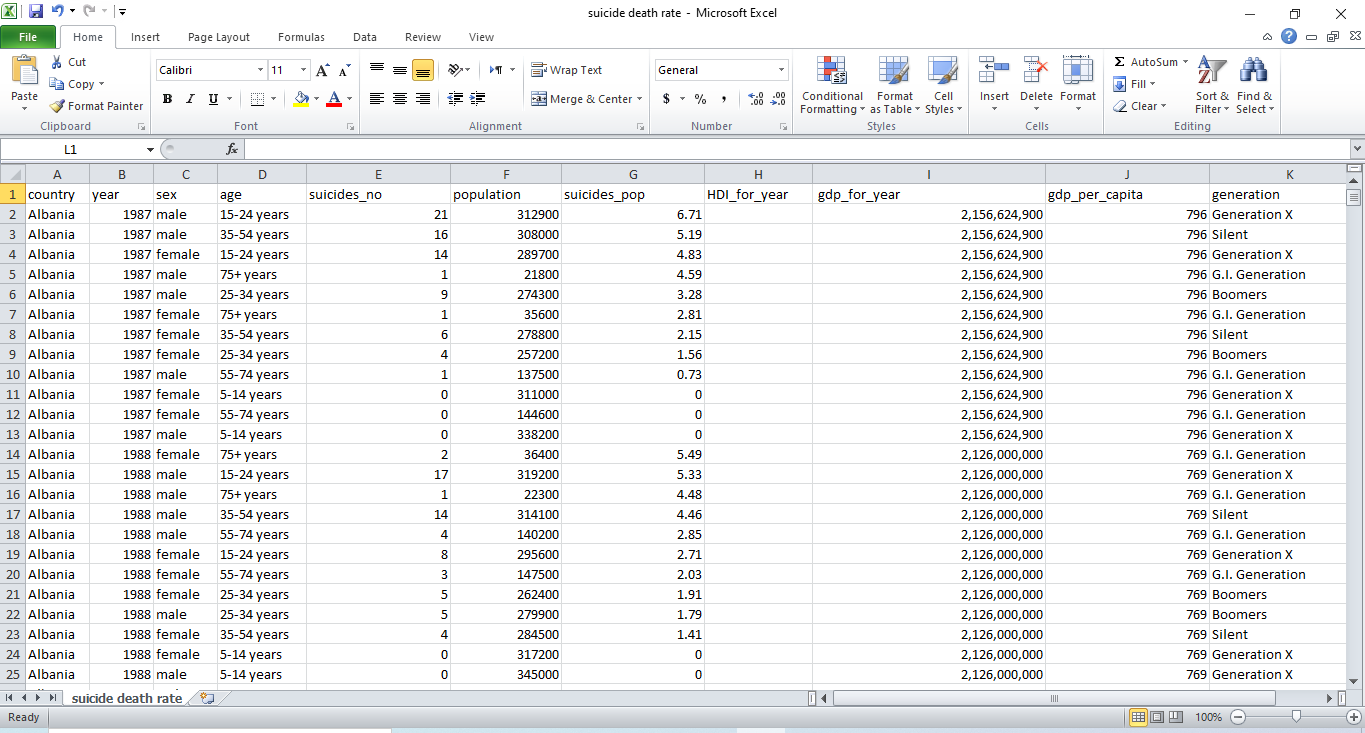
**Before**

**After**

In this step column **country-year** is removed because data are not necessary for analysis.

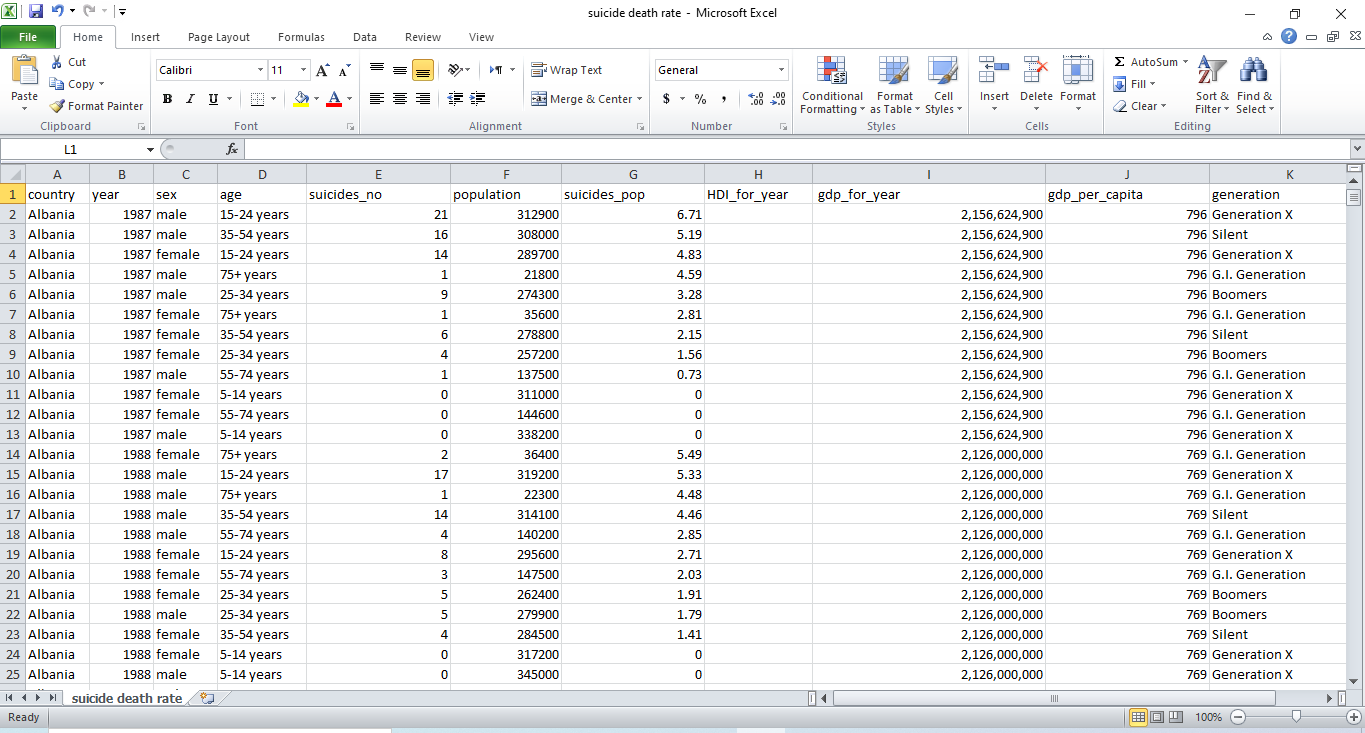
**Step 2: Modifying columns name**

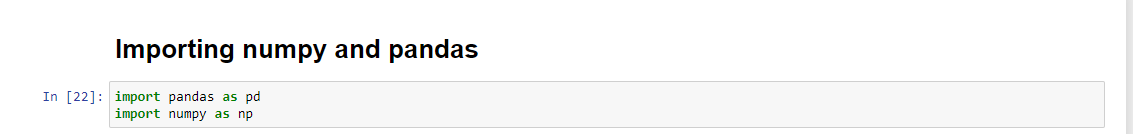
**Before**

**After**

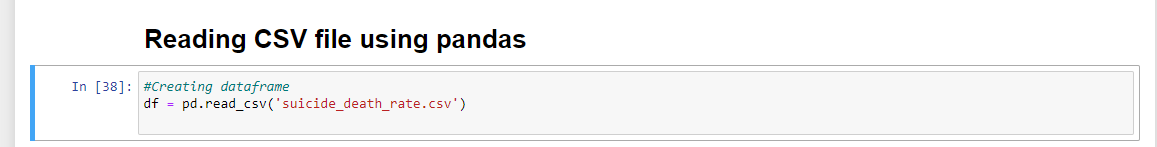
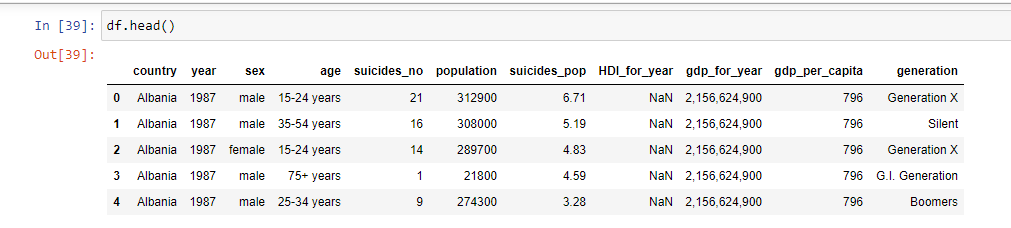
In this step symbols like ($) in **gdp\_per\_capita ($),  gdp\_for\_year ($)** columns are removed**.** In **suicides/100k pop** column name ‘**/100k**’ is replaced with underscore ( **\_** ). Same as replacement in **suicides/100k pop** column for **HDI for year** column also space between each column name words is replaced with ( **\_** ).

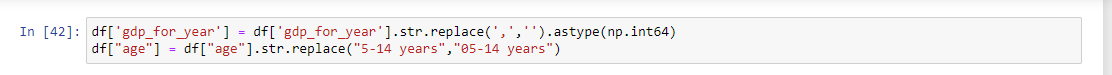
**Step 3: Modifying row values**

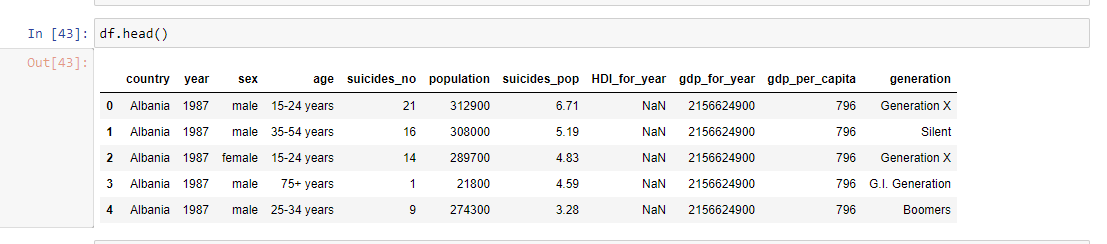
**Before:**

**After**

In this step pandas python library is used because a csv file contains a lot of data so it is very time consuming to clean data of each cell of the csv file. The pandas is a python library which helps to manipulate data.

In pandas library read\_csv method is used to read a csv\_file.

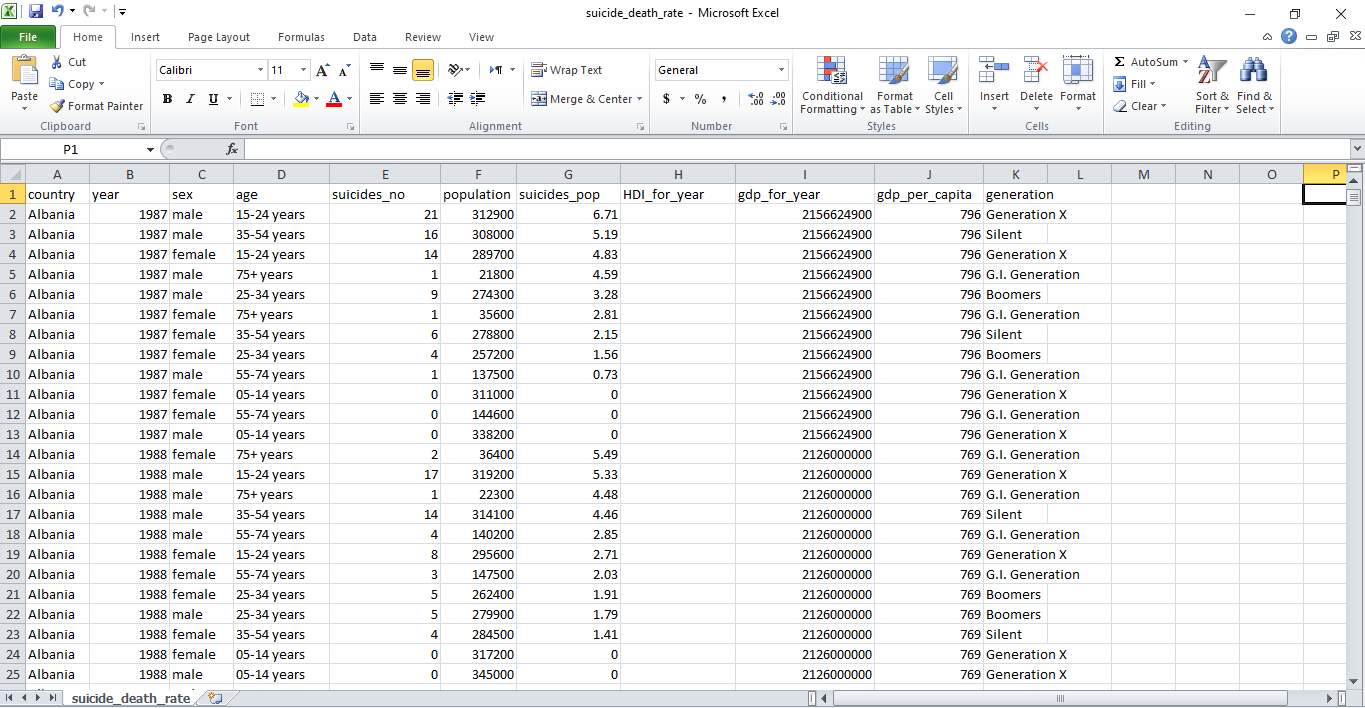
The head function displays the first five row data of csv file. It also takes parameter to display how many data’s the default value is five.  As we can see in gdp\_for\_year columns data there is comma included between numbers.

The above figure shows code that converts numbers into string and replaces ‘,’ with empty string. Then it again converts string into numbers.

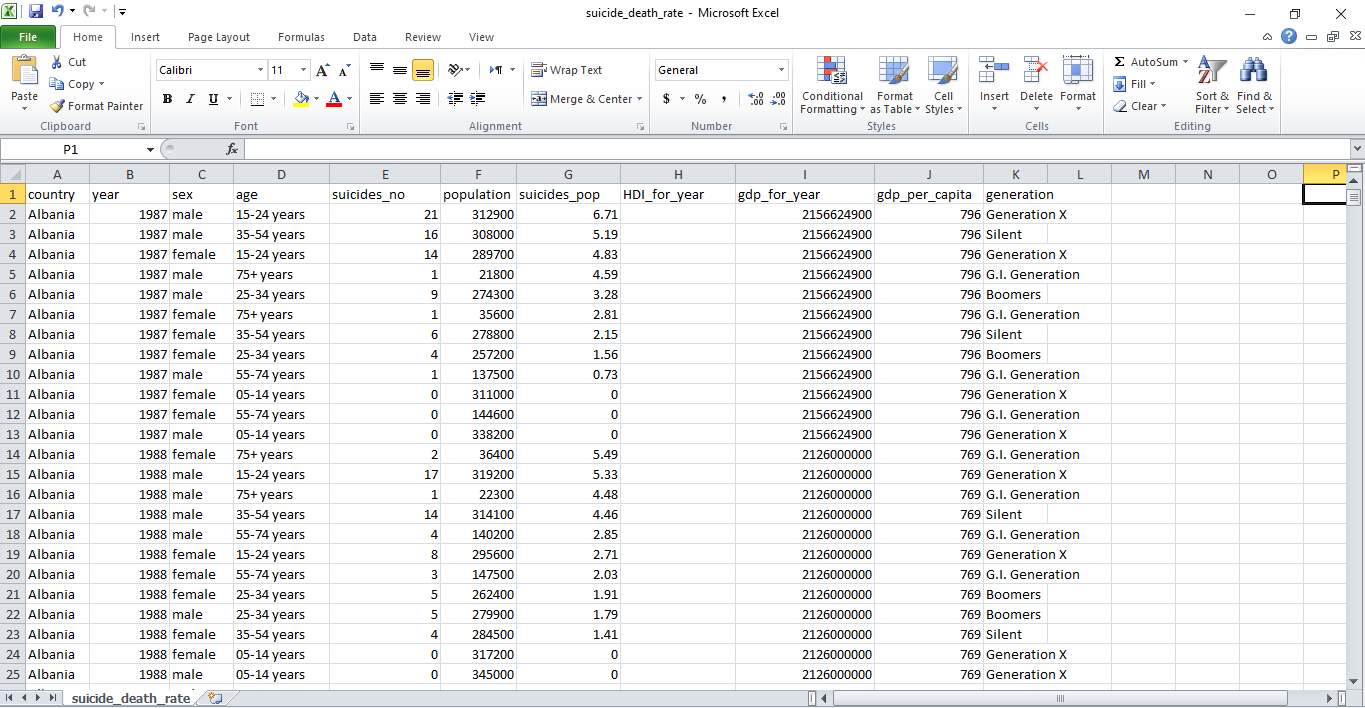
The above figure shows after removing ‘,’ from the data of gdp\_for\_year column.

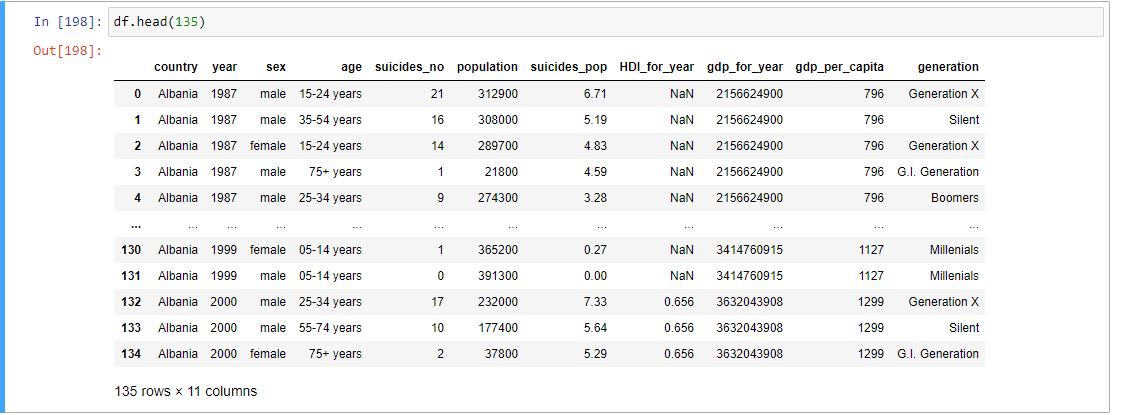


After changes are made in the data frame the data frame will be converted in a csv file and replaced with an old csv file. To replace an old csv file with a new one the to\_csv method is used which takes two parameters. The first parameter is for a new csv file name which is the same as the old one and another parameter is an option for including index inside of a csv file or not.

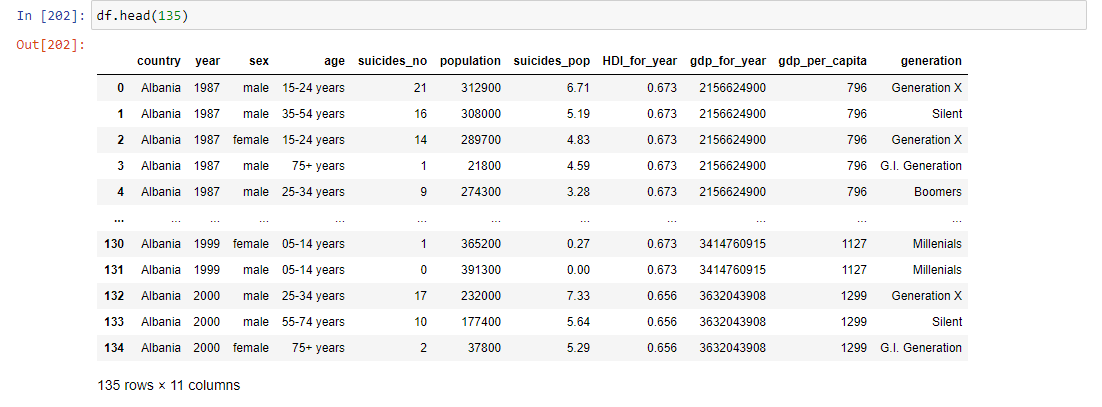
The above figure shows that after replacing the old csv file with a new one.

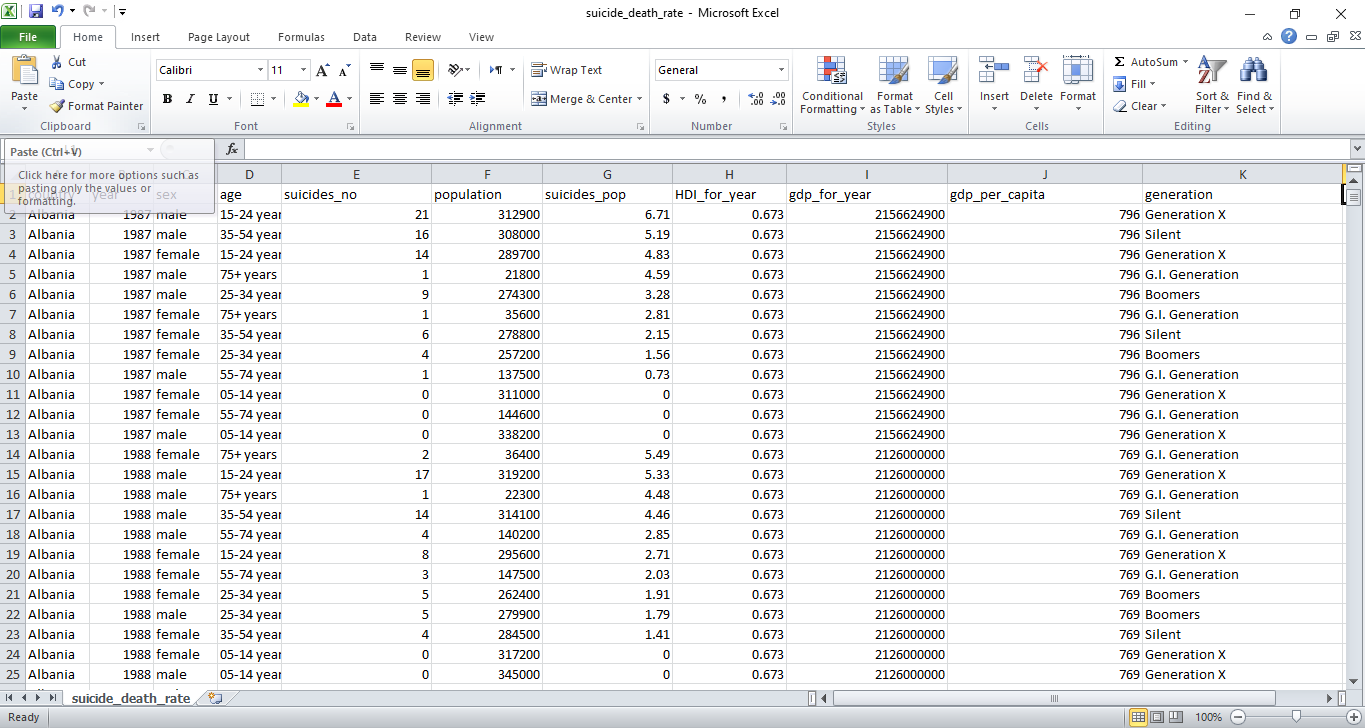
**Step 4: Filling value in empty cell**

**Before:**

**After:**

As shown in above code it fills nan value which is in HDI\_for\_year column with the mean of a HDI of a country.

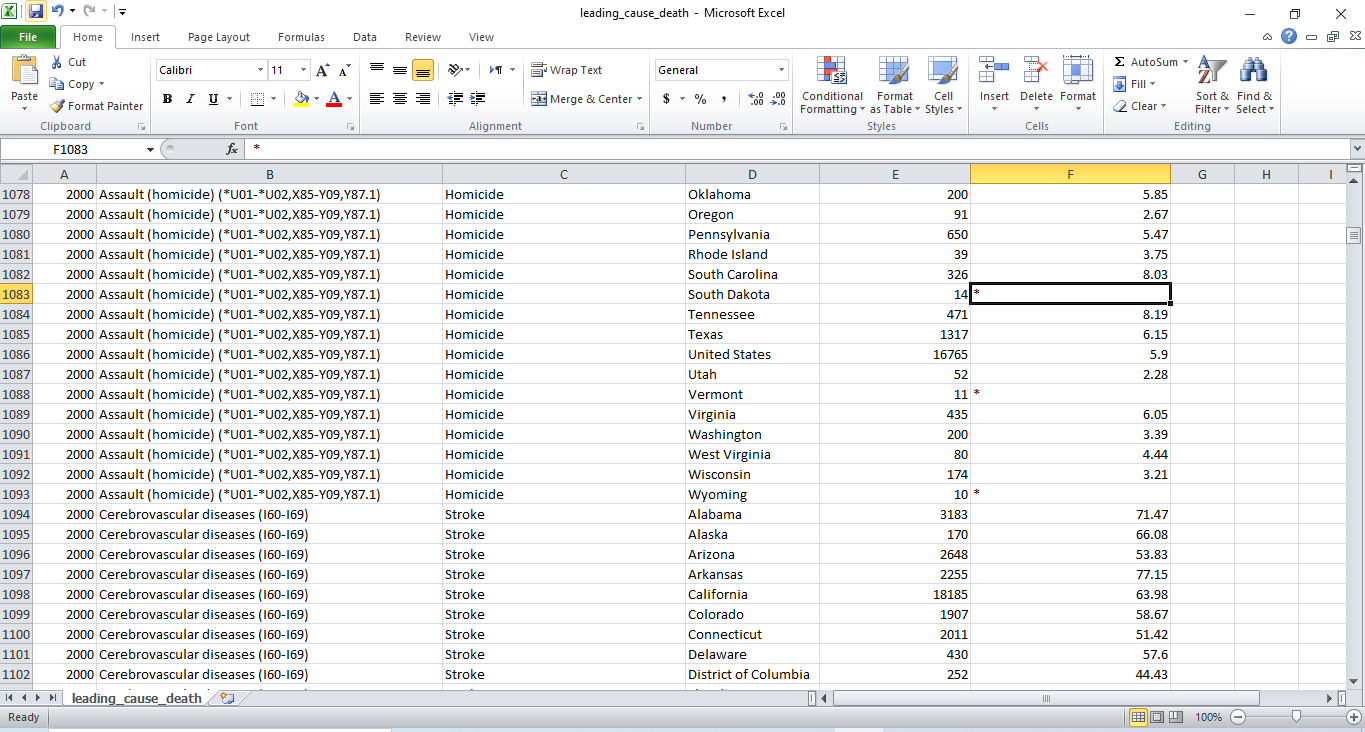
****

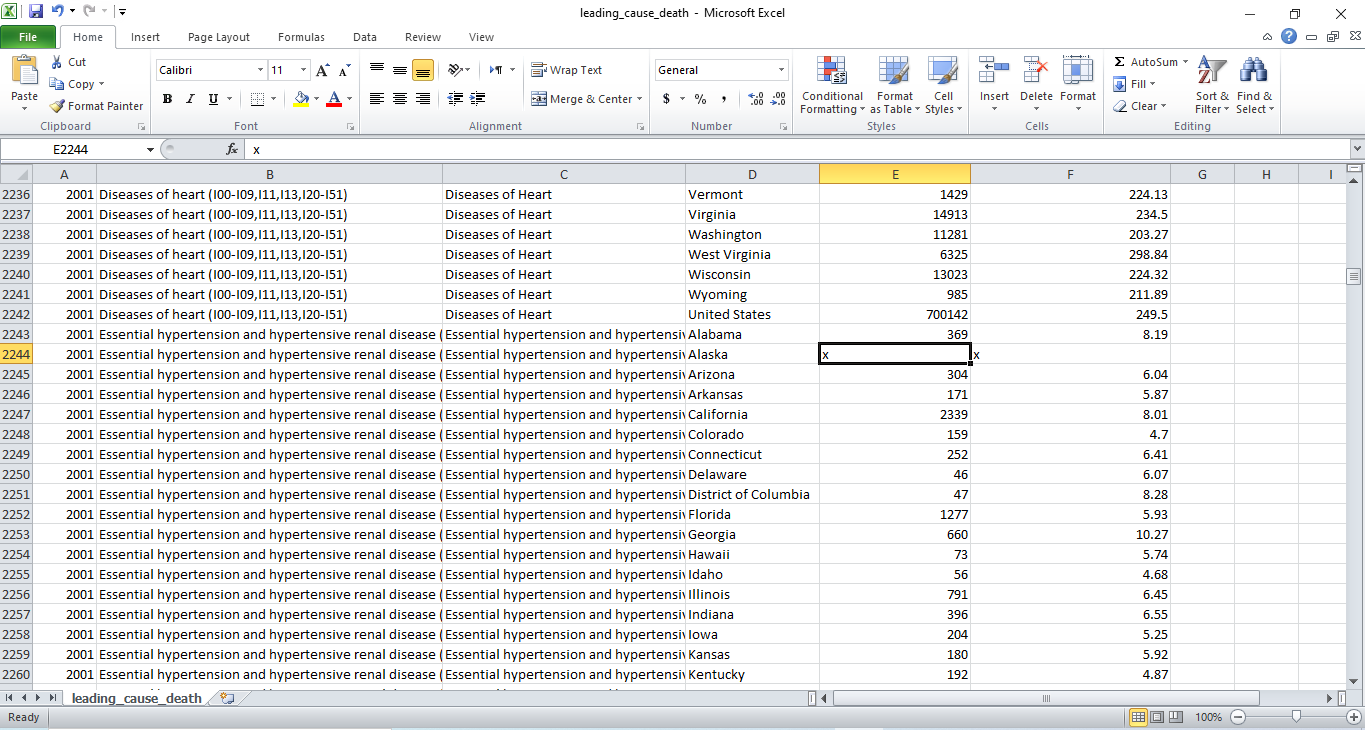
****

**Cleaning dataset for Hadoop**

**Removing (\*, x) from csv file**

Before:





After:

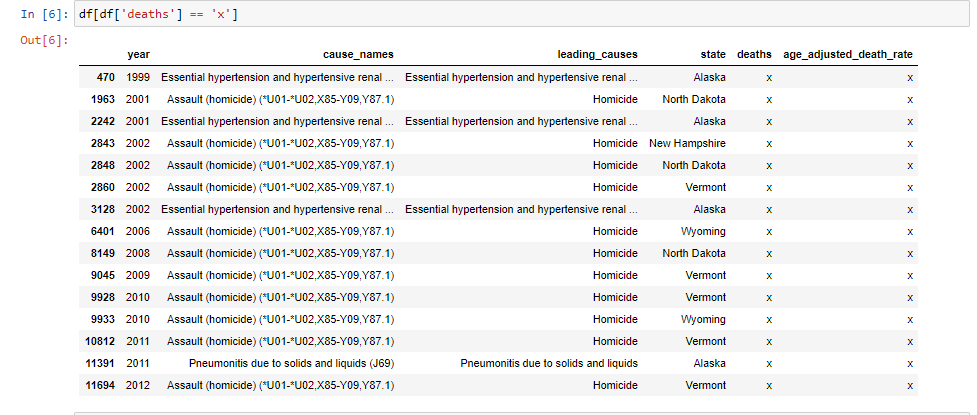
Importing pandas:



Reading CSV file



Displaying data frame rows where death column contains ‘x’:

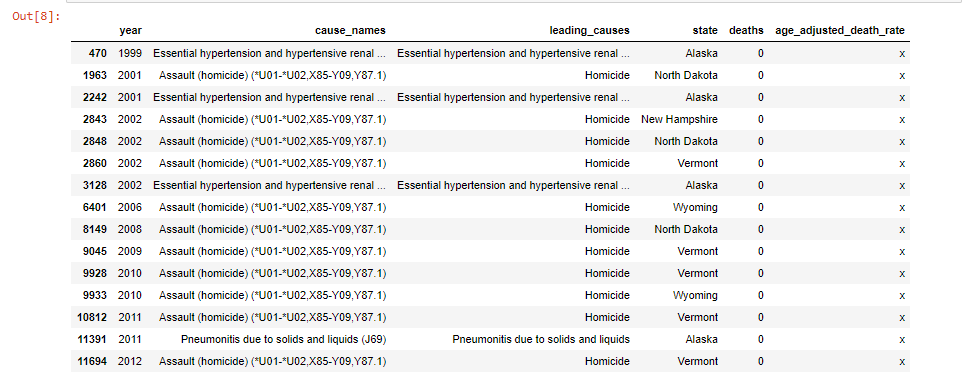


Replacing ‘x’ with 0 in deaths columns data:

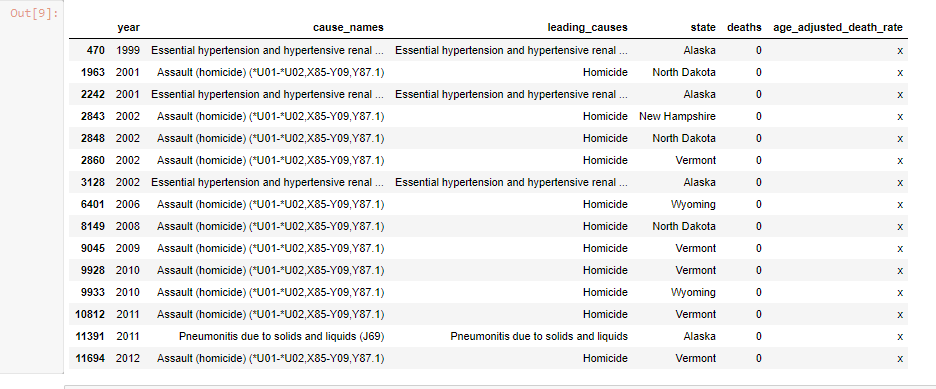


Displaying data frame rows where deaths columns rows contain value 0.

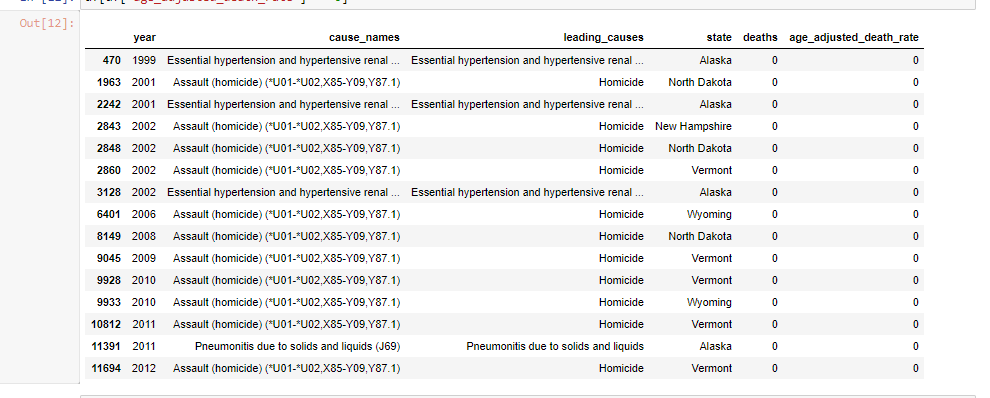


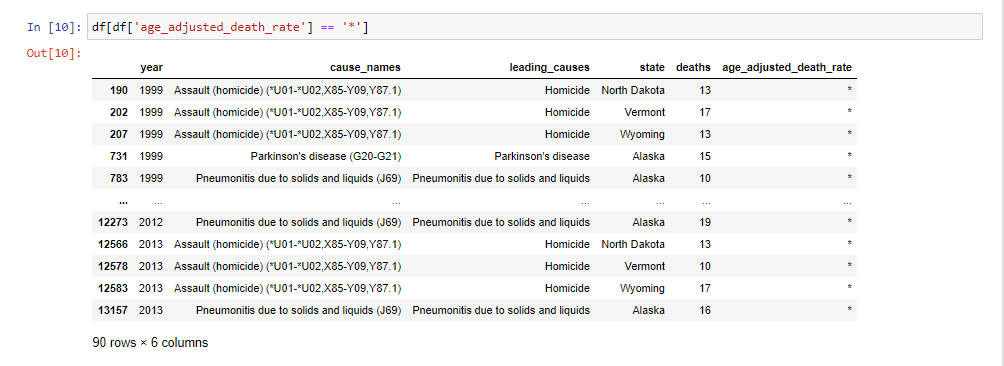


Displaying data frame rows where age\_adjusted\_death\_rate column contains ‘x’:



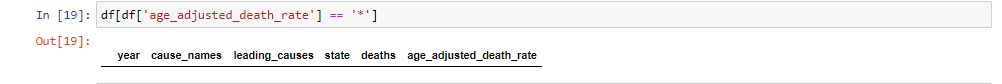
Replacing ‘x’ with 0 in age\_adjusted\_death\_rate column:

After removing ‘x’ from deaths and age\_adjusted\_death\_rate column.

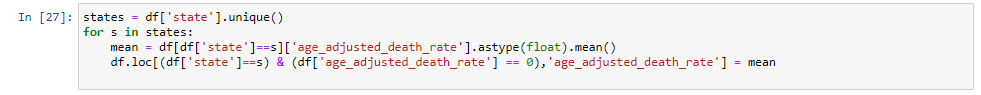
Displaying rows from data frame where age\_adjusted\_death\_rate contains ‘\*’: 

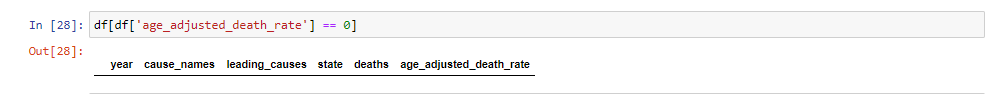
Replacing ‘\*’ with 0 in age\_adjusted\_death\_rate column:



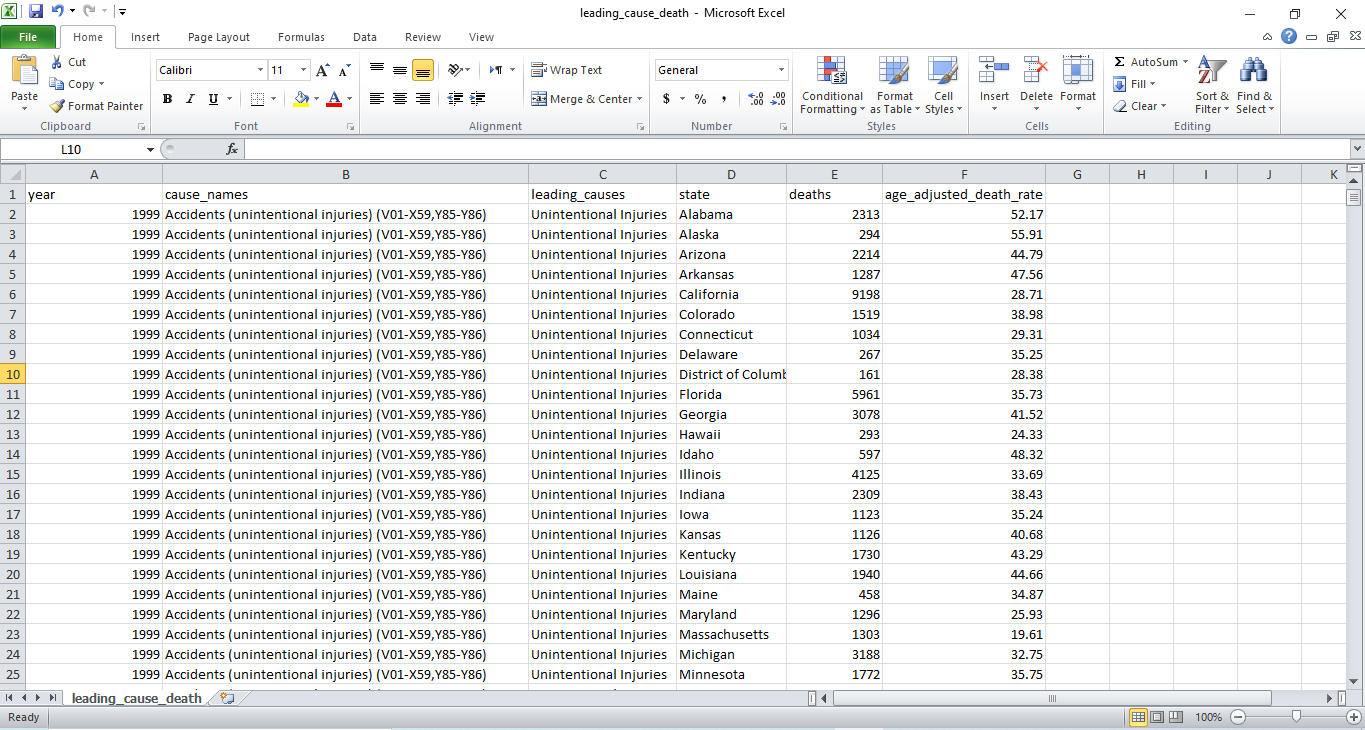
After removing ‘\*’ from age\_adjusted\_death\_rate column:

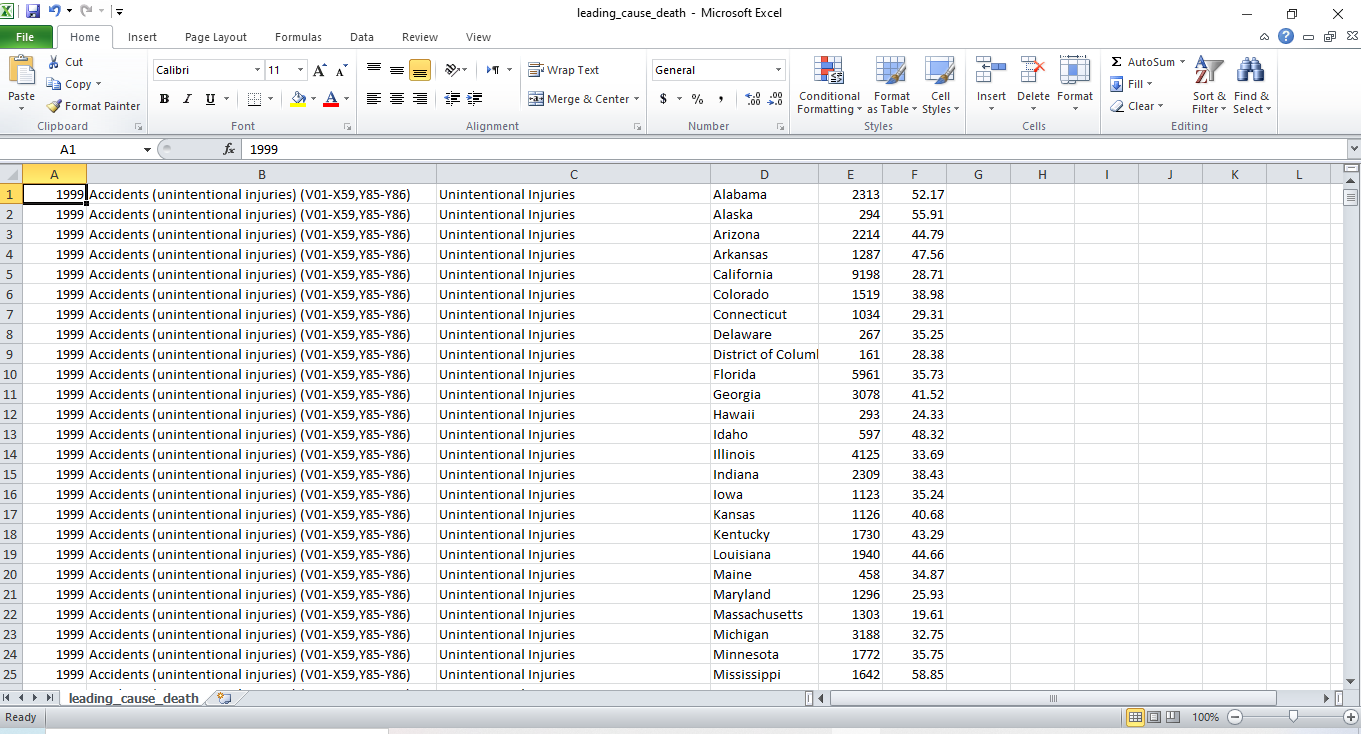
Replacing 0 with mean value in age\_adjusted\_death\_rate column:



After replacing 0 with mean value in age\_adjusted\_death\_rate column:

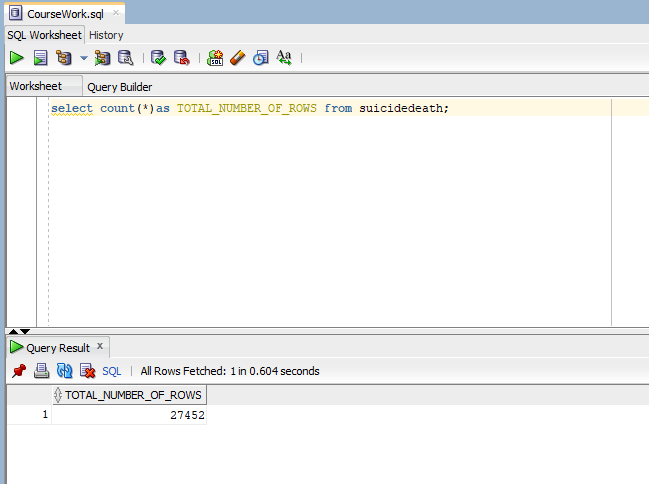
**Removing all the columns**

Before:

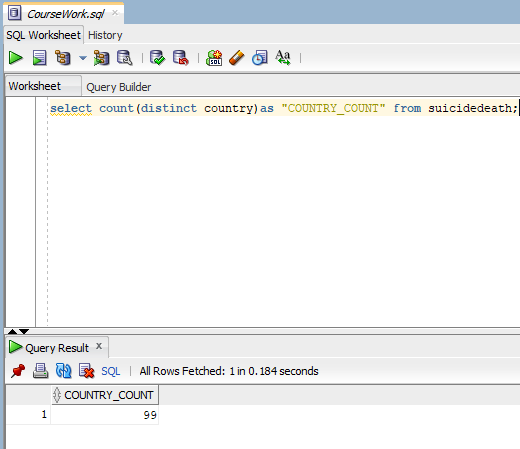
After:

# Analysis of Data

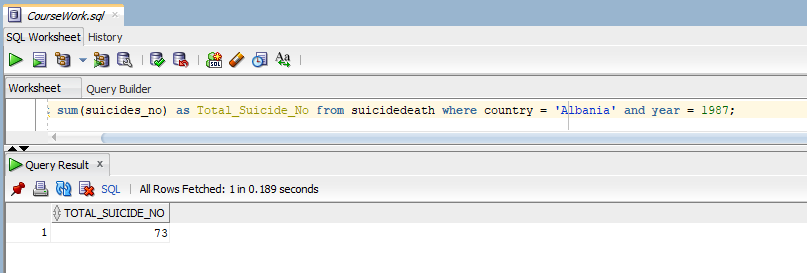
## Query for oracle

**Counting total number of rows present in table.**

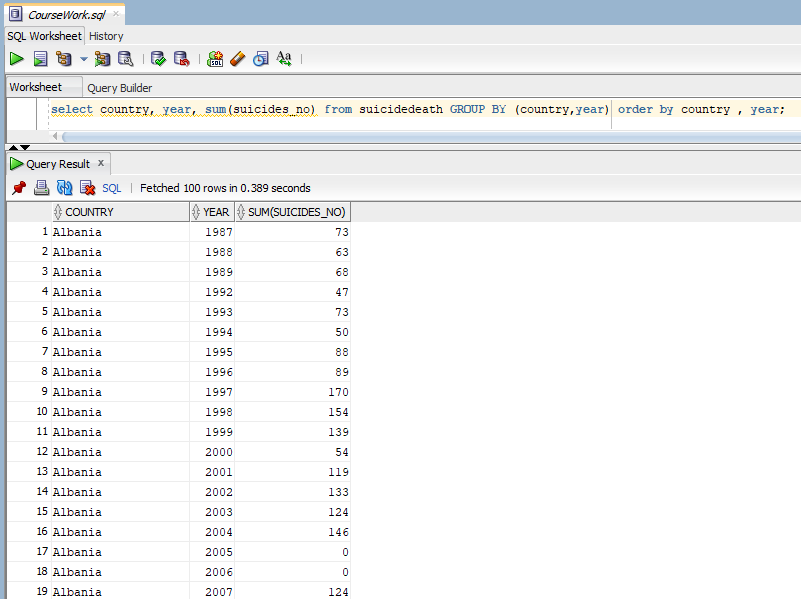
**Counting total distinct countries**



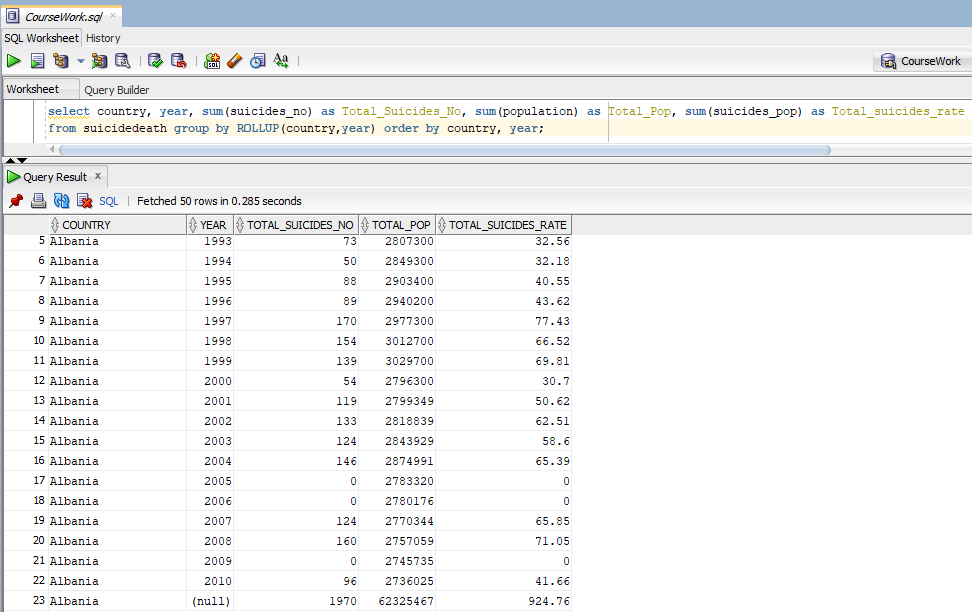
**Summing total number of suicides happening in country Albania in Year 1987.**

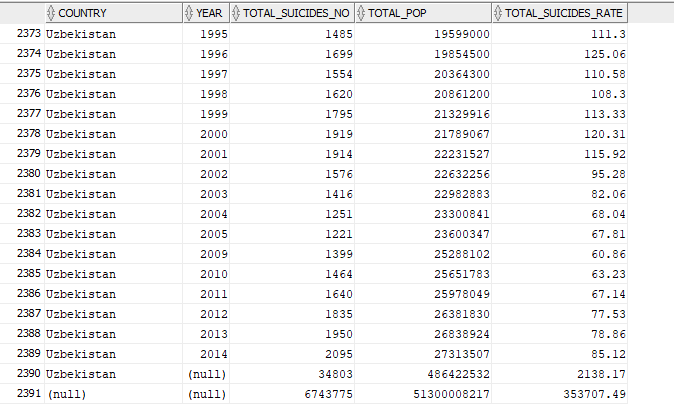


**Grouping  by country and year and adding number of suicide happening in each year.**

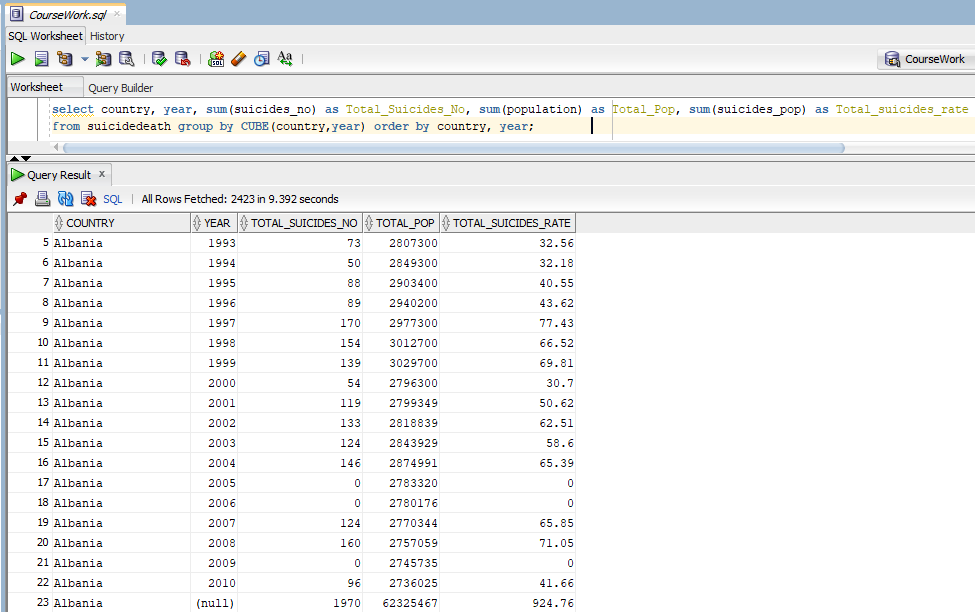


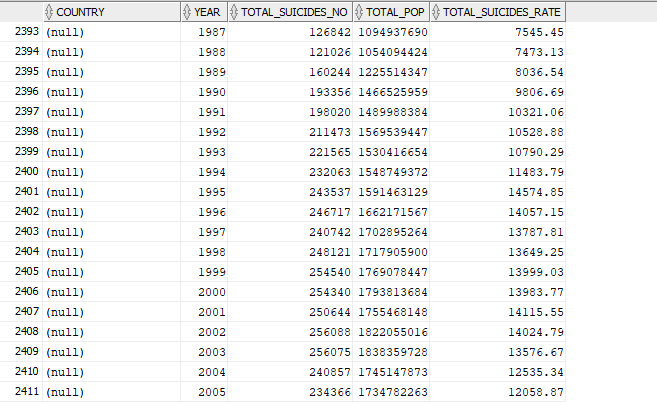
**Group by using ROLLUP.**





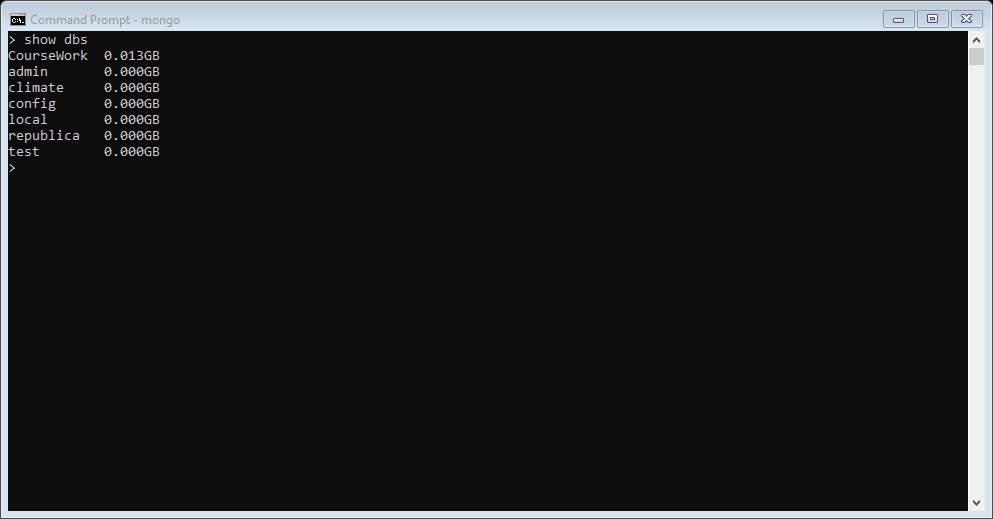
**Group by using CUBE.**



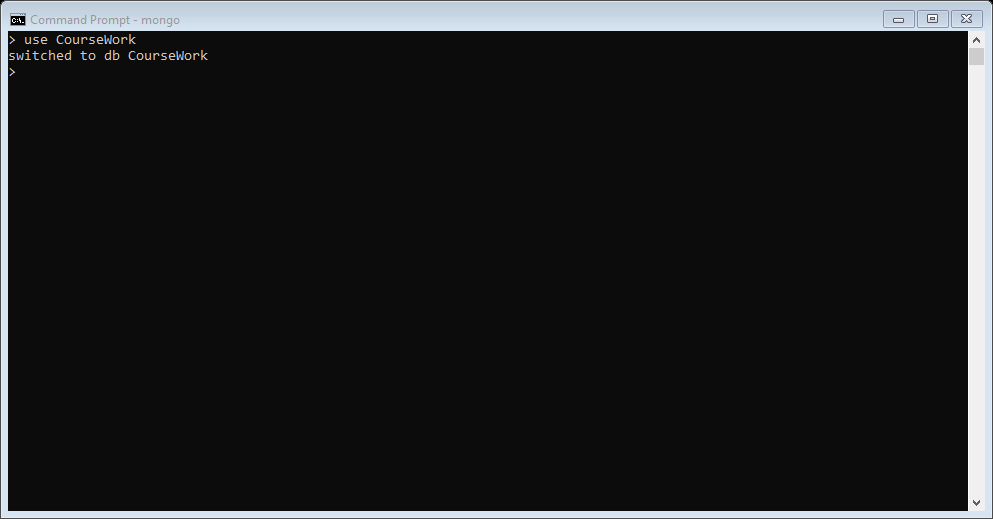


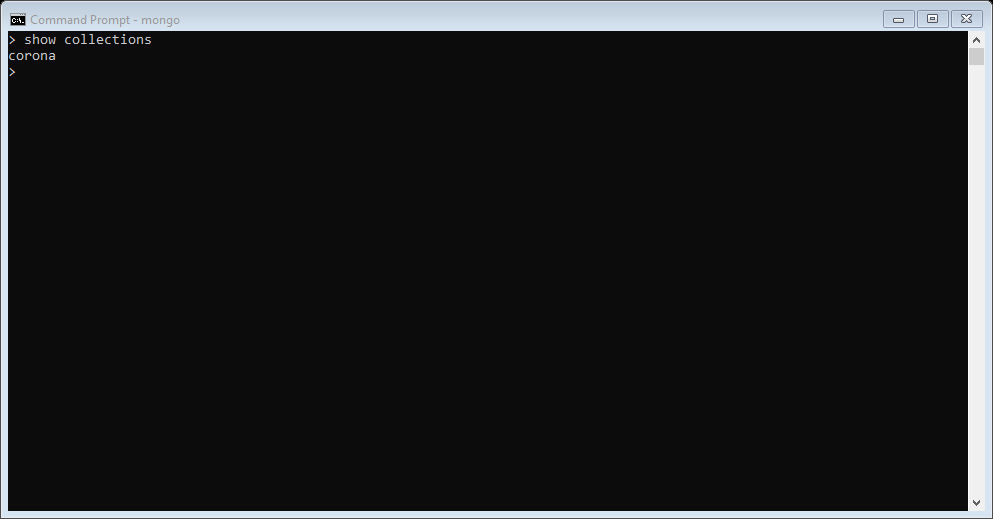
## Query for MongoDB

**Display store database**



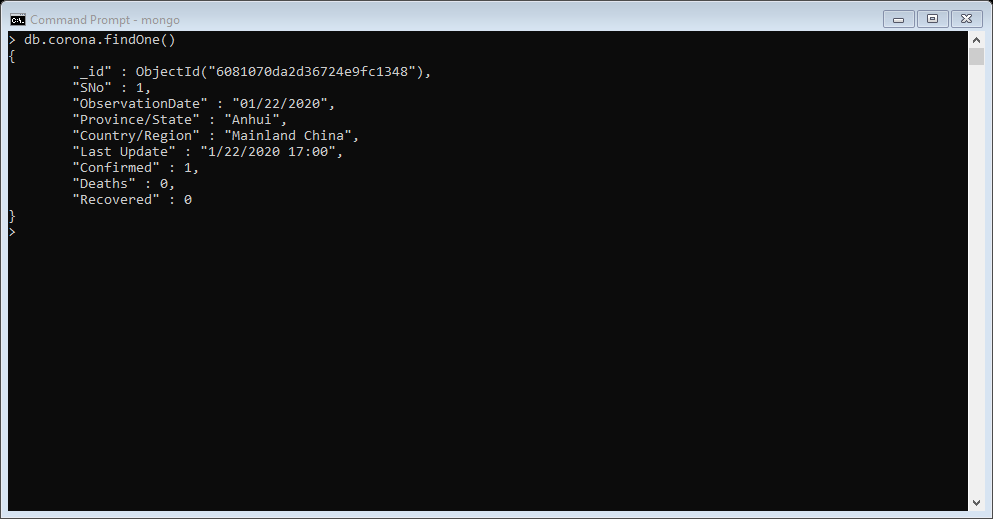
**Using CouseWork database.**



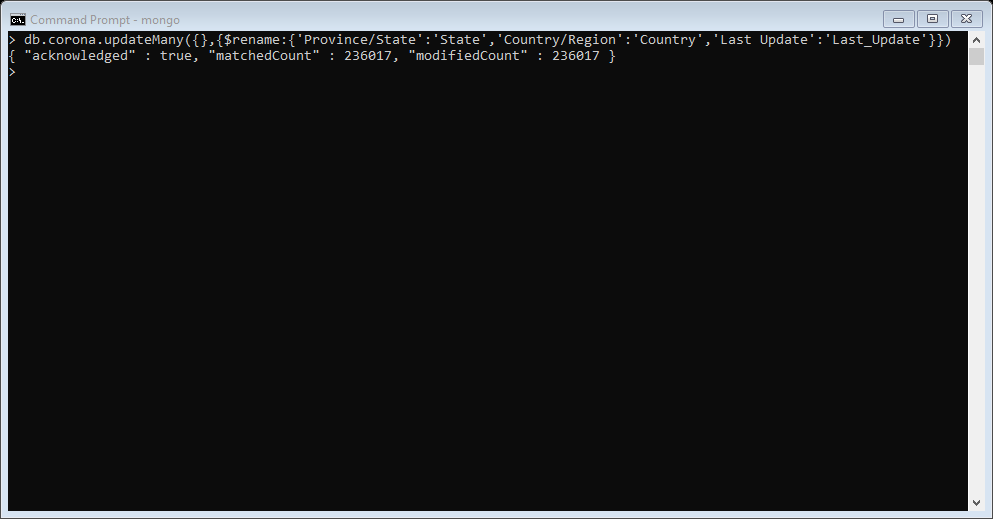
**Displaying collections of CourseWork database.**

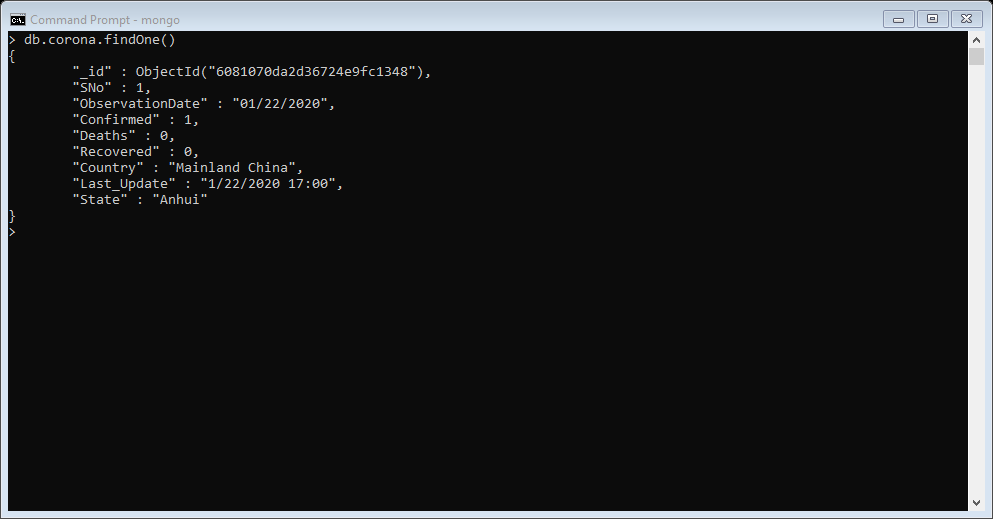
**Renaming Column**

**Before:**

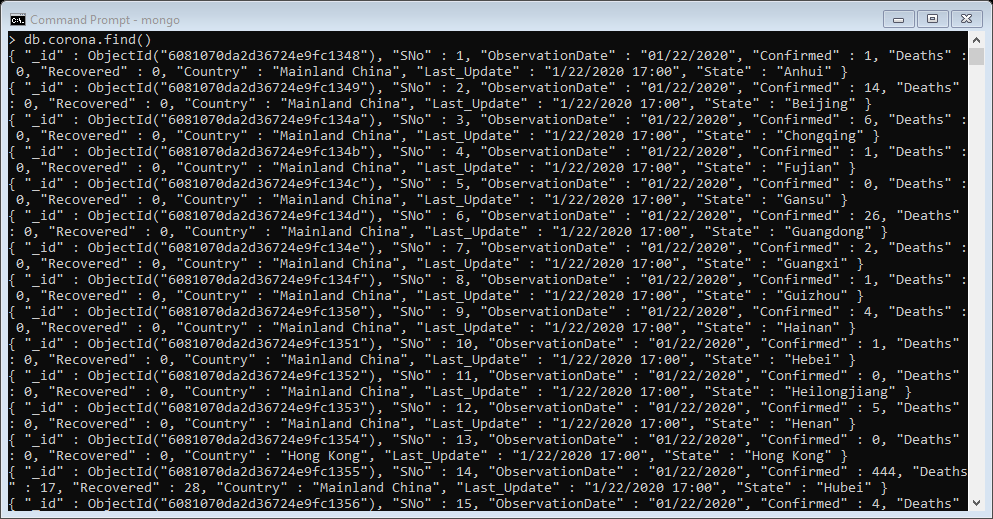


**After:**

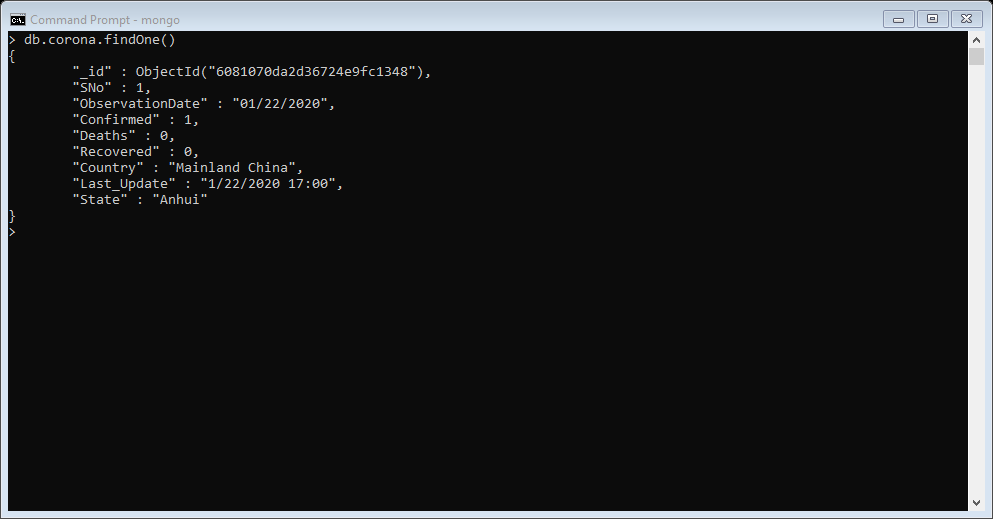




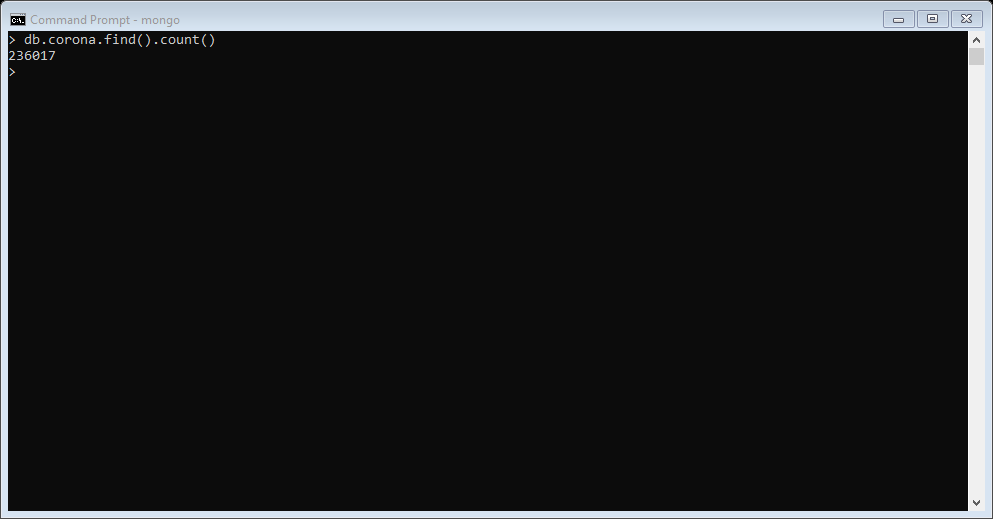
**Displaying all the data present in corona collection.**



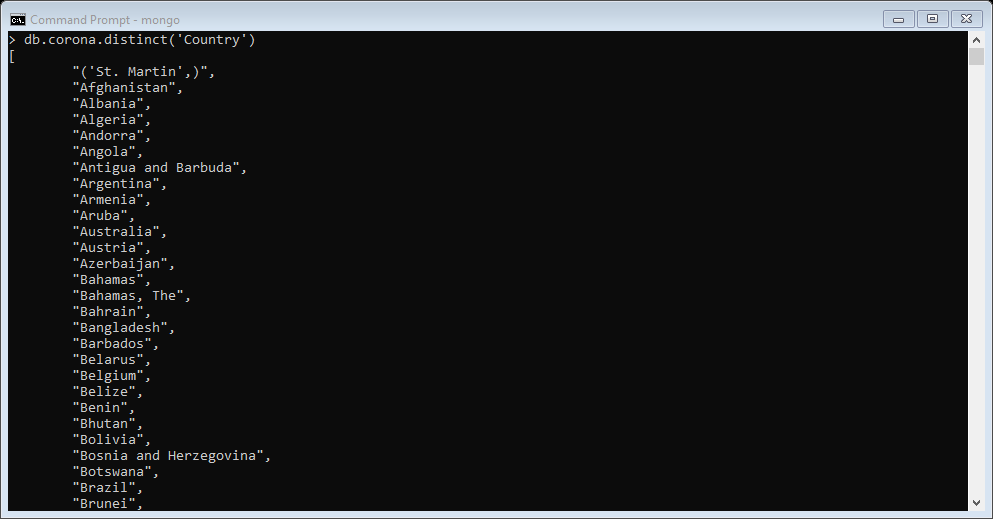
**Displaying only one data stored in a corona collection.**



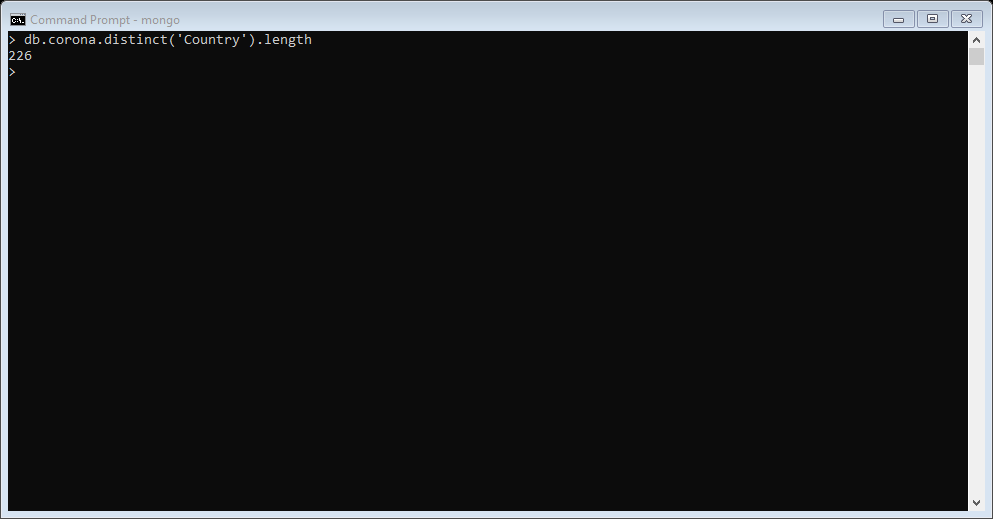
**Counting how many data contains in corona collection.**



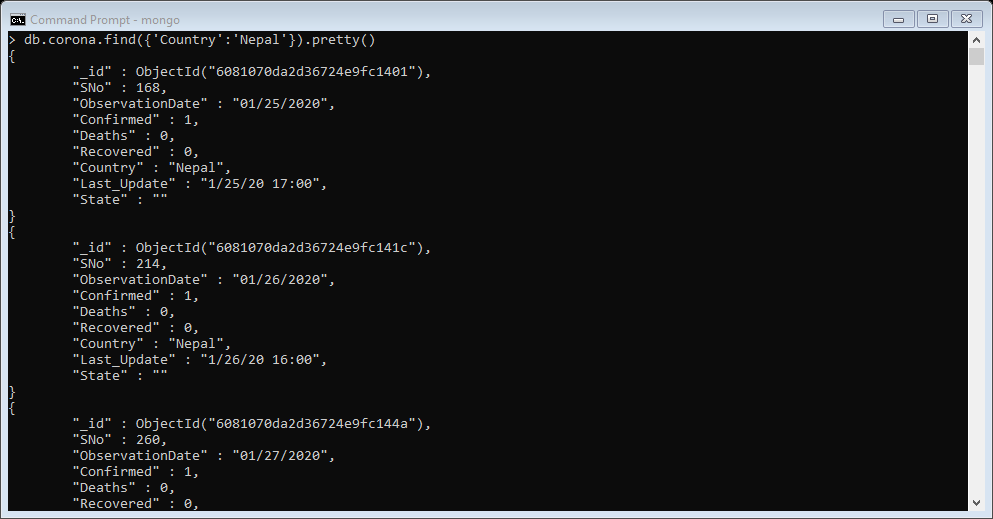
**Displaying unique countries names.**



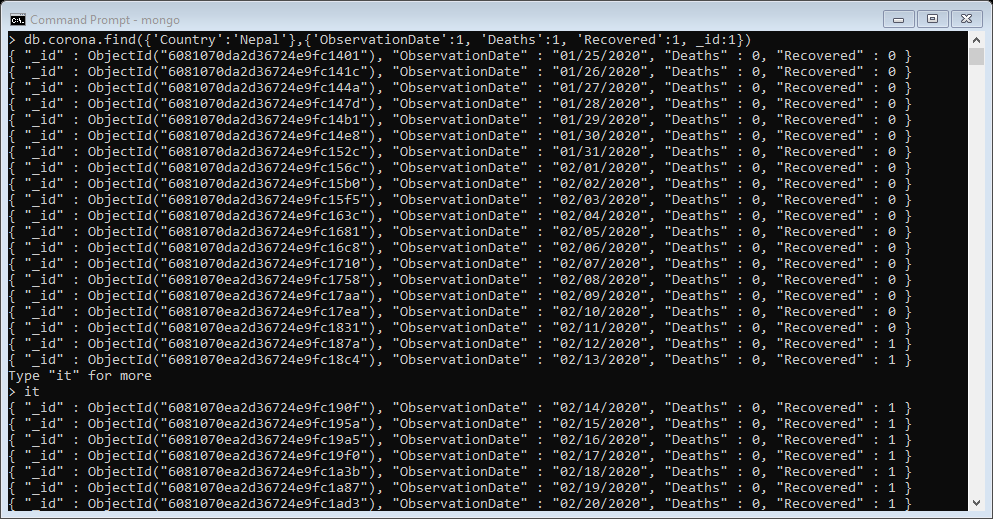
**Displaying unique countries count.**



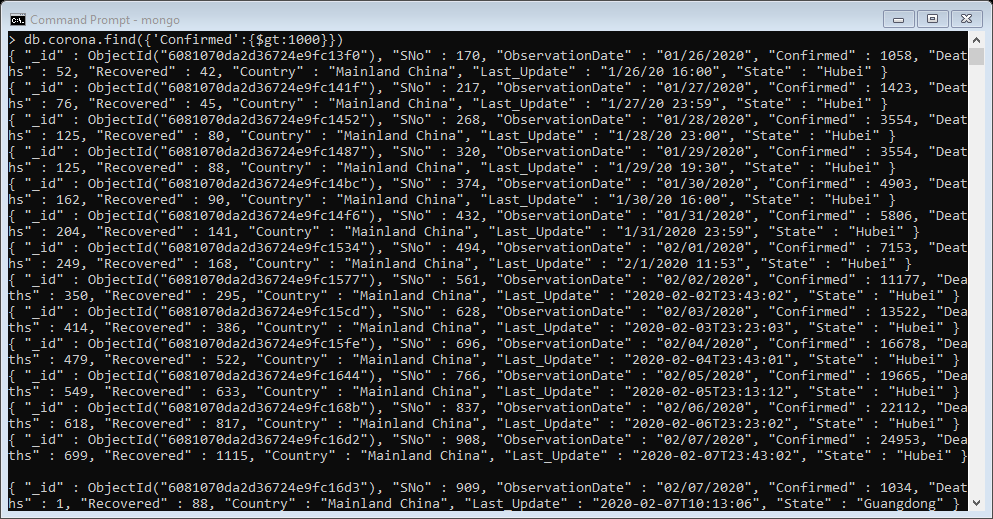
**Displaying those data which has a country name Nepal.**



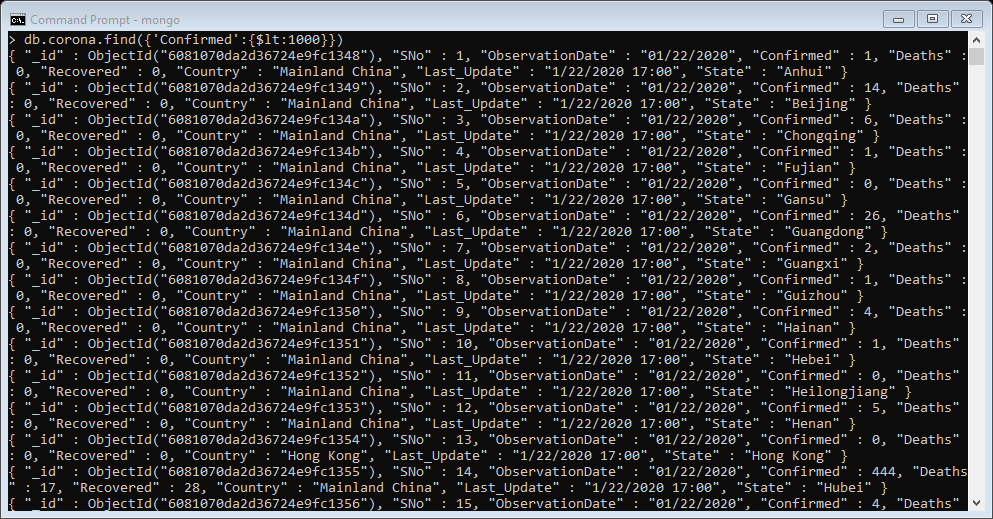
**Displaying column data ObservationDeath, Deaths, Recovered, \_id which has a country name Nepal.**



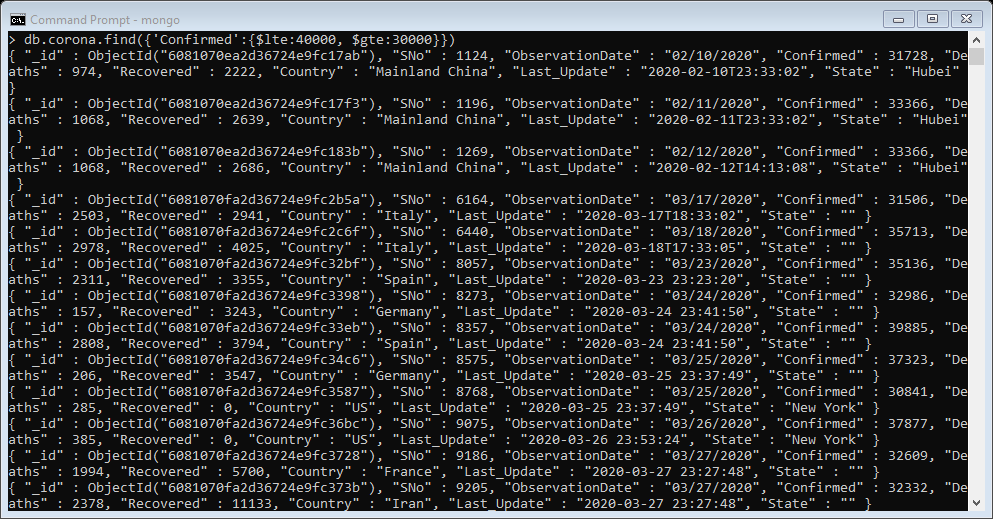
**Displaying data’s which has confirmed corona cases greater than 1000.**



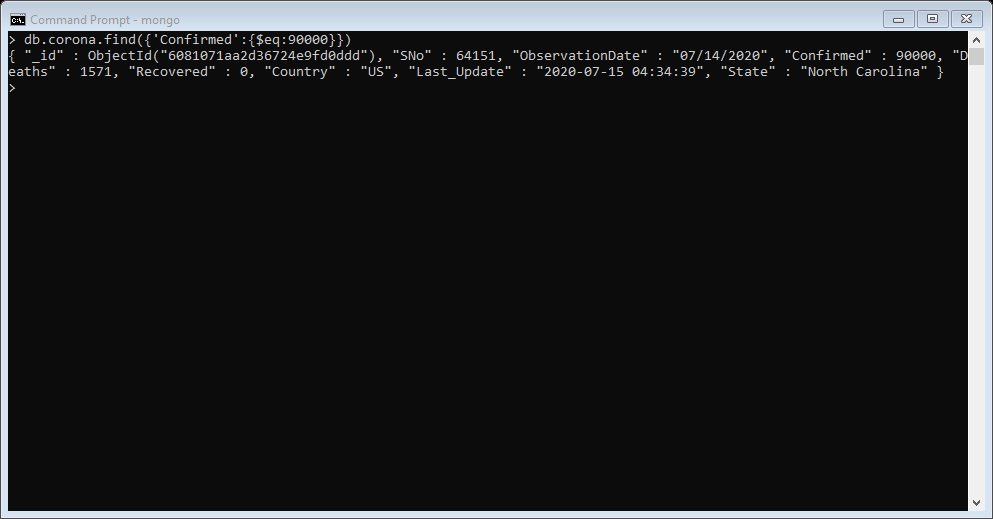
**Displaying data’s which has confirmed corona cases less than 1000.**



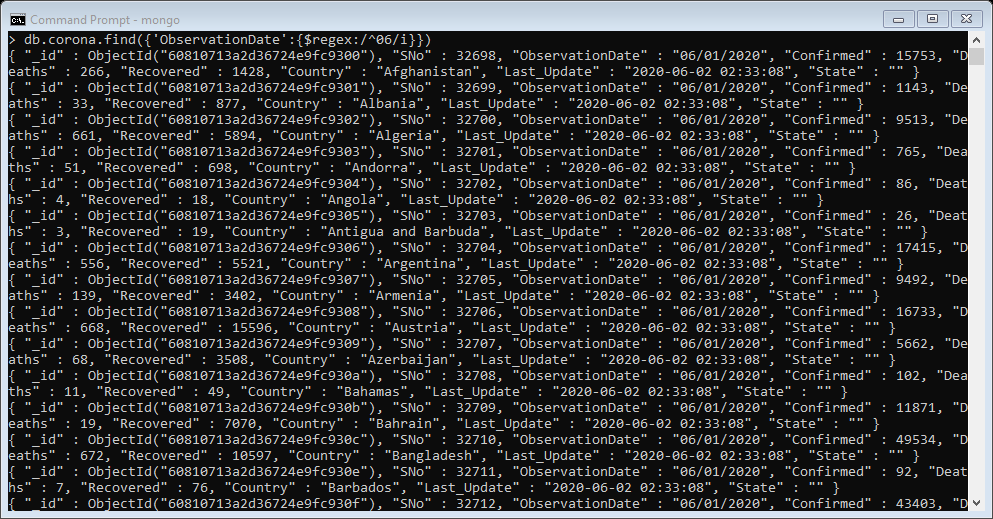
**Displaying data’s which has confirmed corona cases greater or equal to 30000 and less than or equal to 40000.**



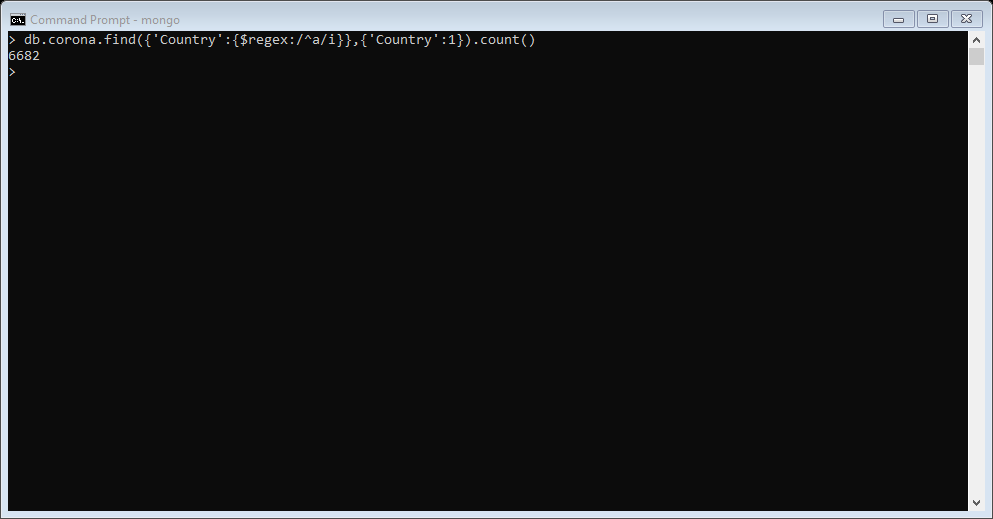
**Displaying data’s which has confirmed corona cases equal to 90000.**



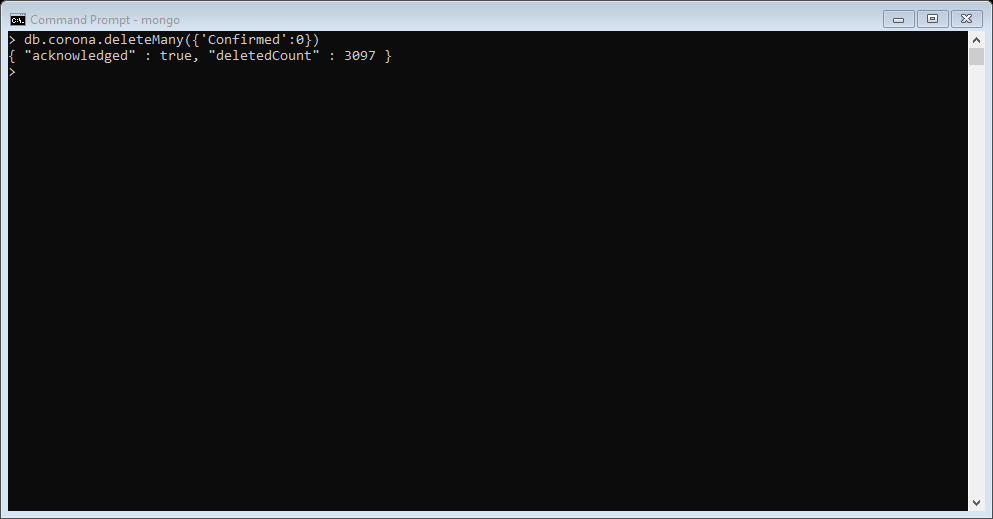
**Displaying data’s which has ObservationDate which has month 06.**



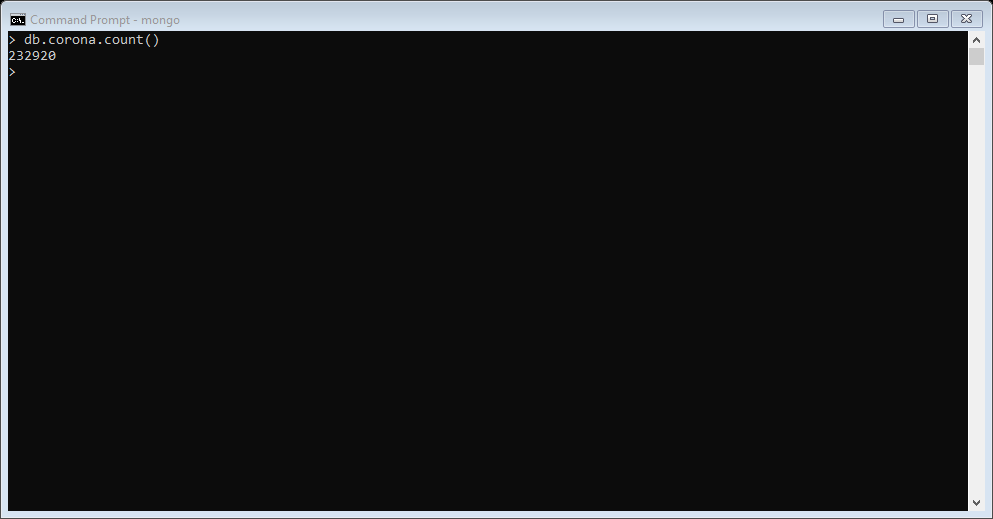
**Counting those data’s which has country name starts with ‘a’ or ‘A’.**



**Deleting data which has confirm corona cases zero.**

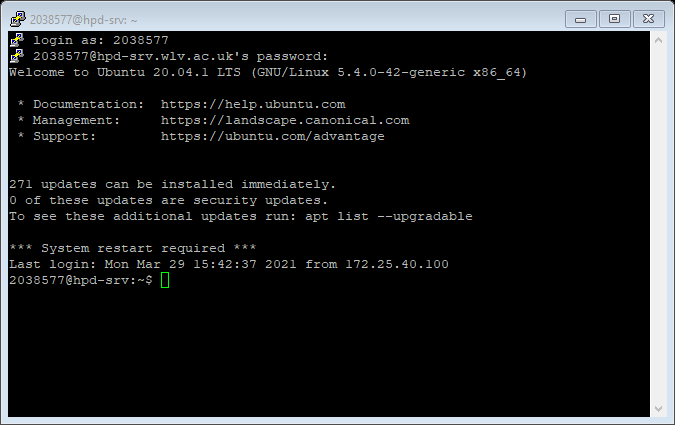


**Counting data contained in corona collection after removing certain data.**

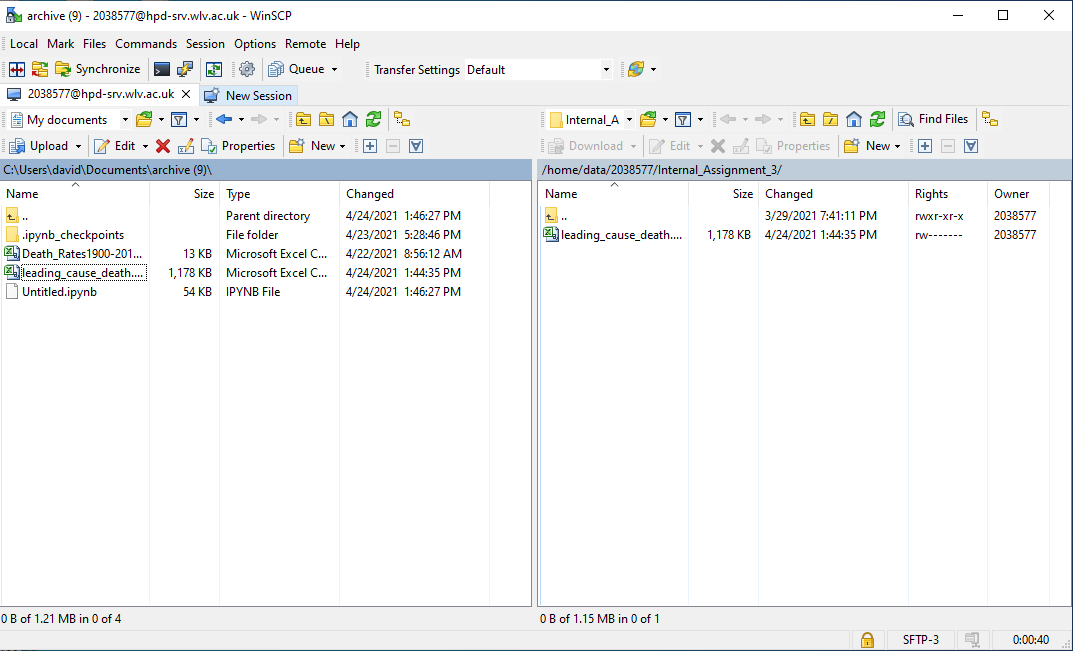


## Query for Hadoop

Login using username and password in putty

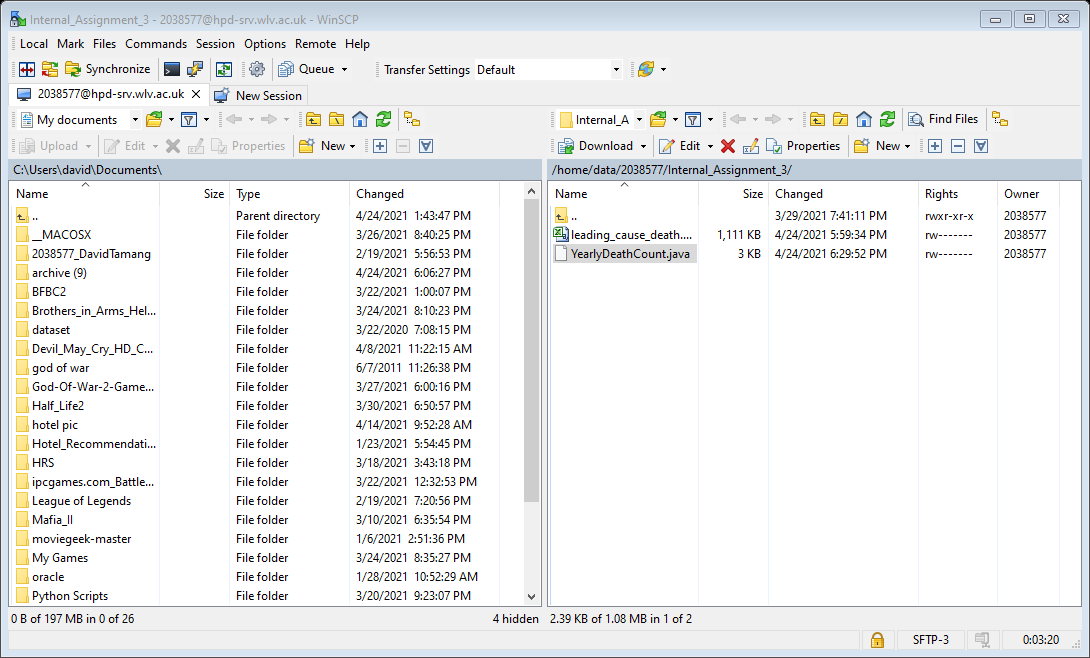
****

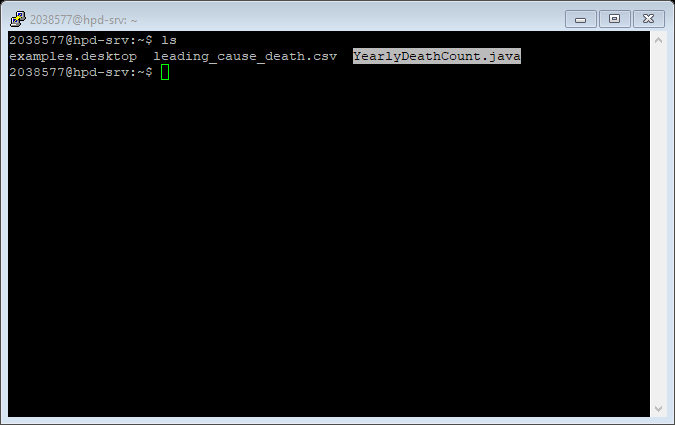
Uploading CSV file in WinSCP:

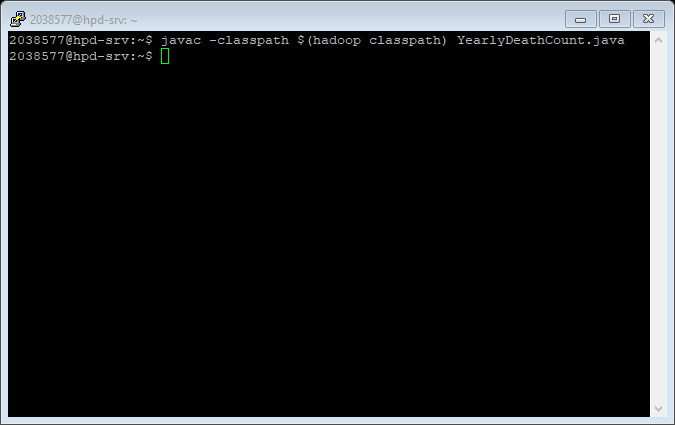


After uploading CSV file in WinSCP:

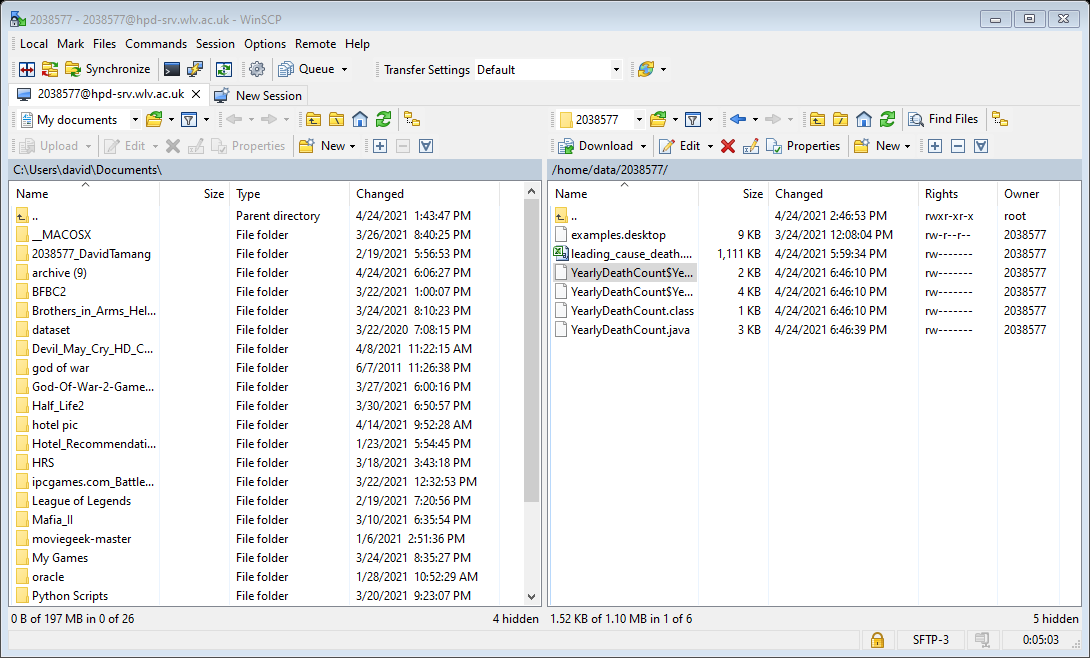


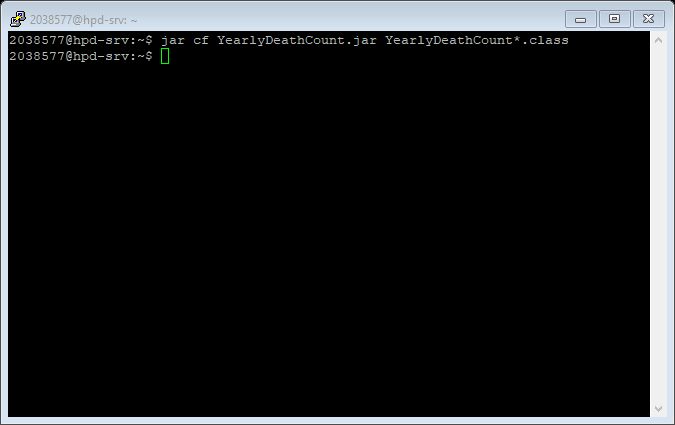
Uploading YearlyDeathCount.java in winSCP:

After uploading YearlyDeathCount.java in winSCP:

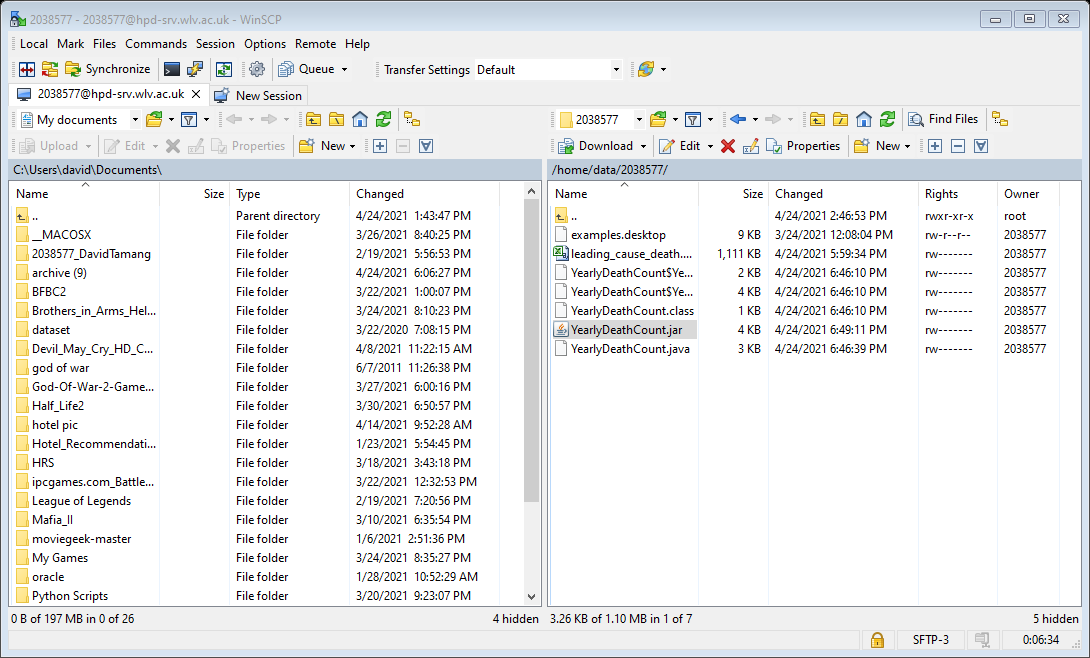
Creating YearlyDeathCount.class file using YearlyDeathCount.java:

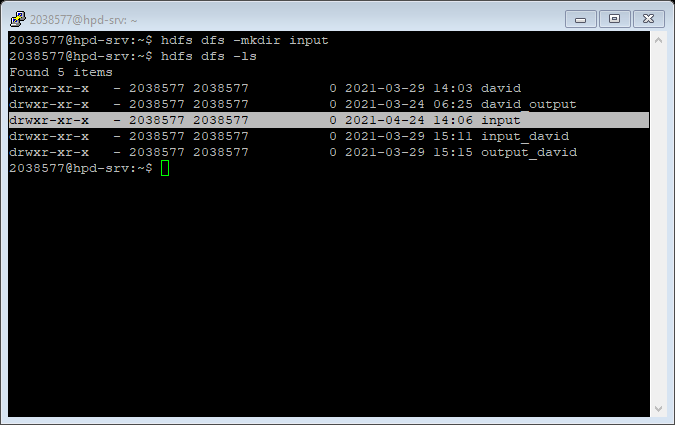
After Creating YearlyDeathCount.class file using YearlyDeathCount.java:

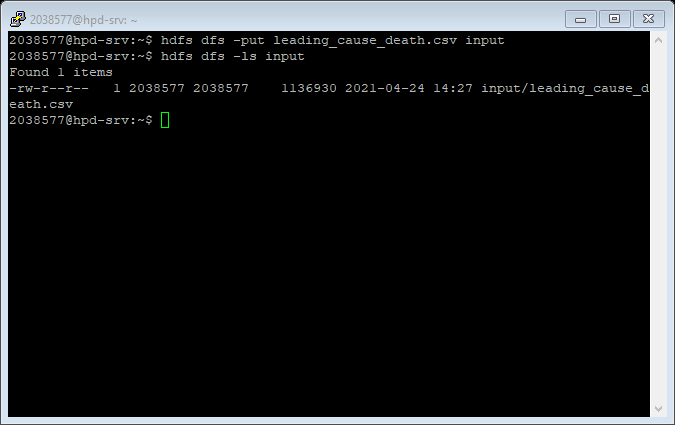


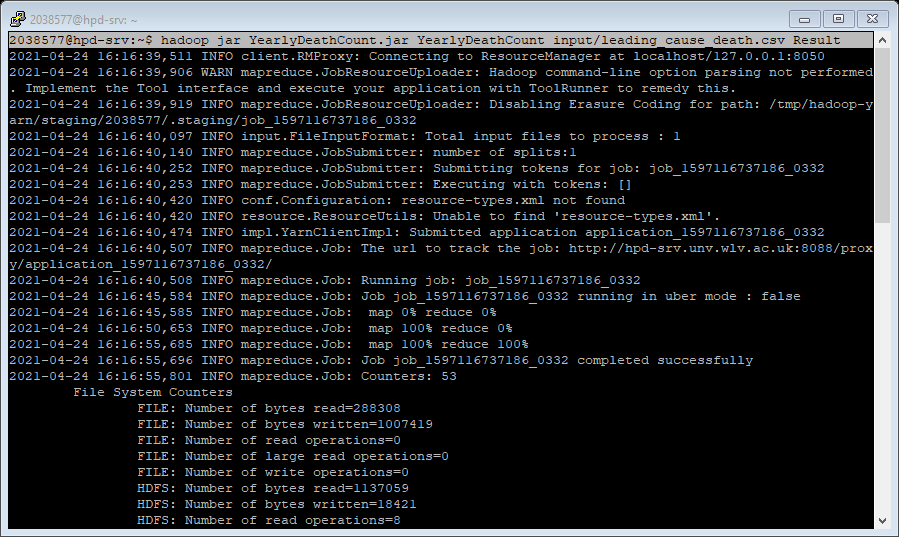
Creating jar file:

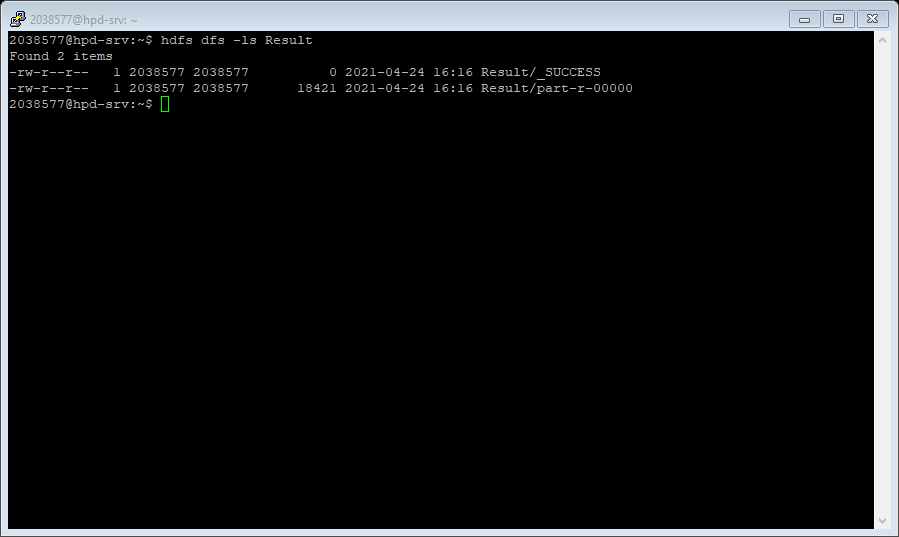
After Creating jar file:



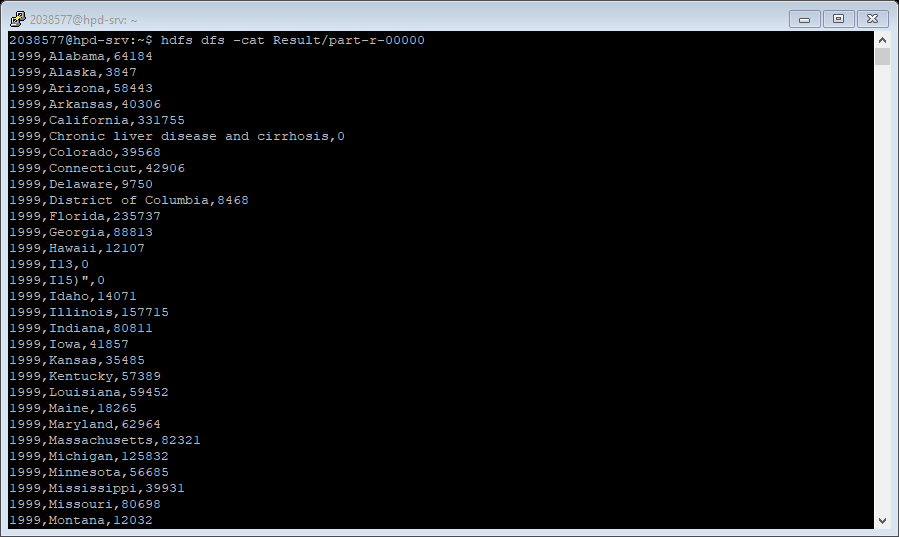
Creating new folder in hadoop:

Putting leading\_cause\_death.csv file inside of input folder:

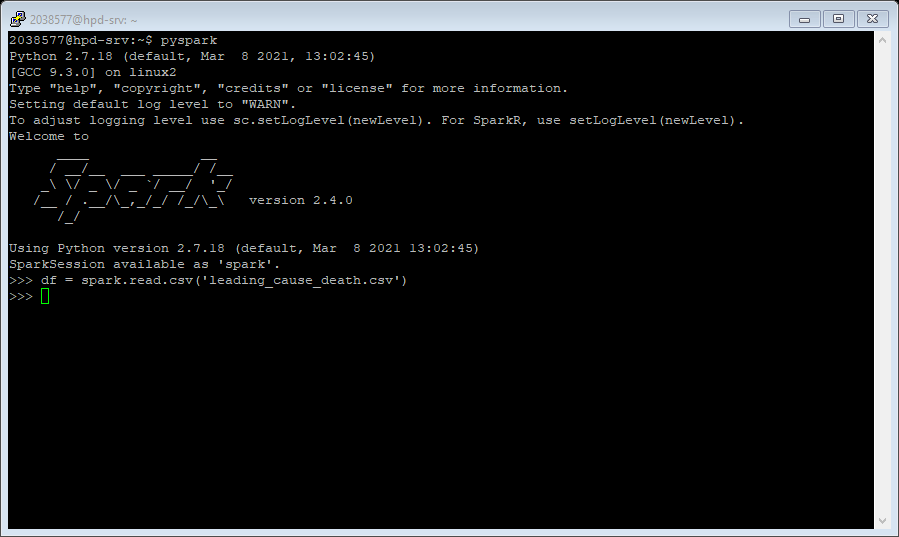
Using jar file to performe map-reduce operation and saving output inside of Result folder:

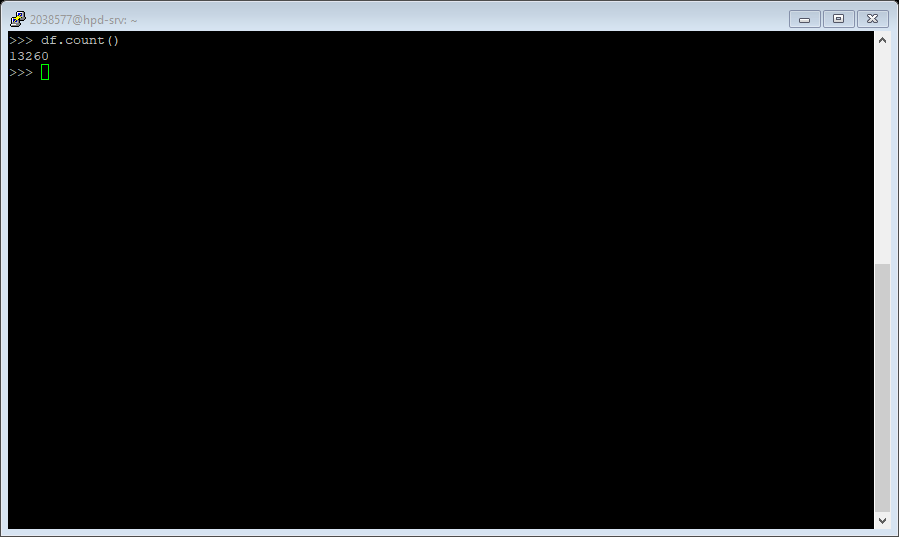
After performing map-reduce operation:

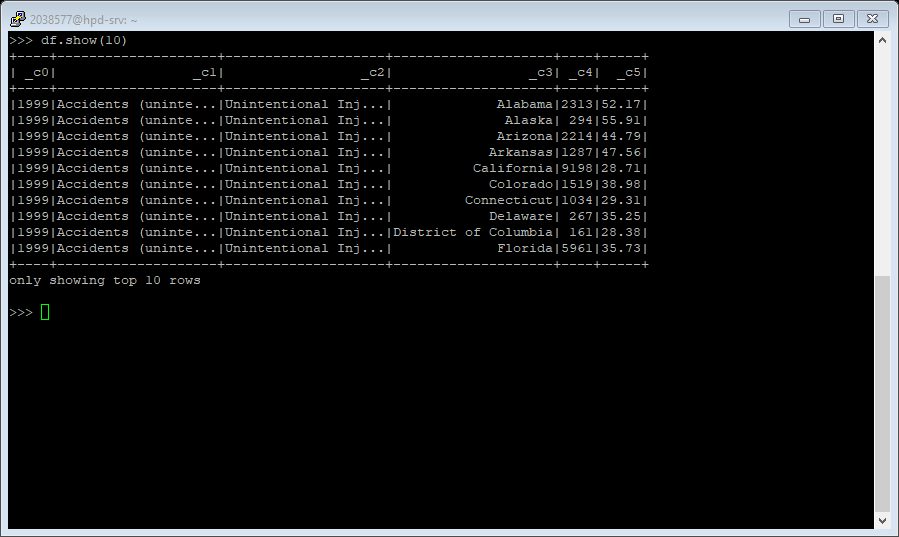
Displaying output of map-reduce operation on csv file:

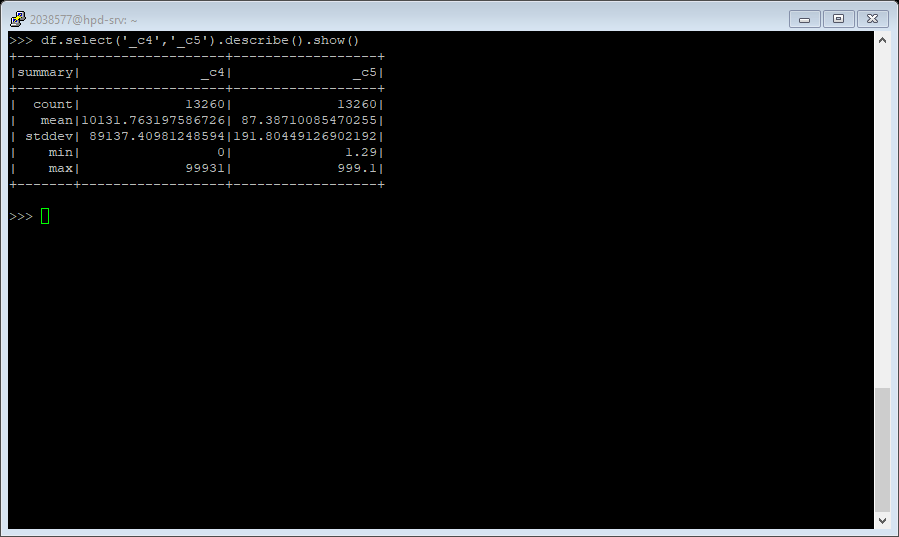


**PySpark**

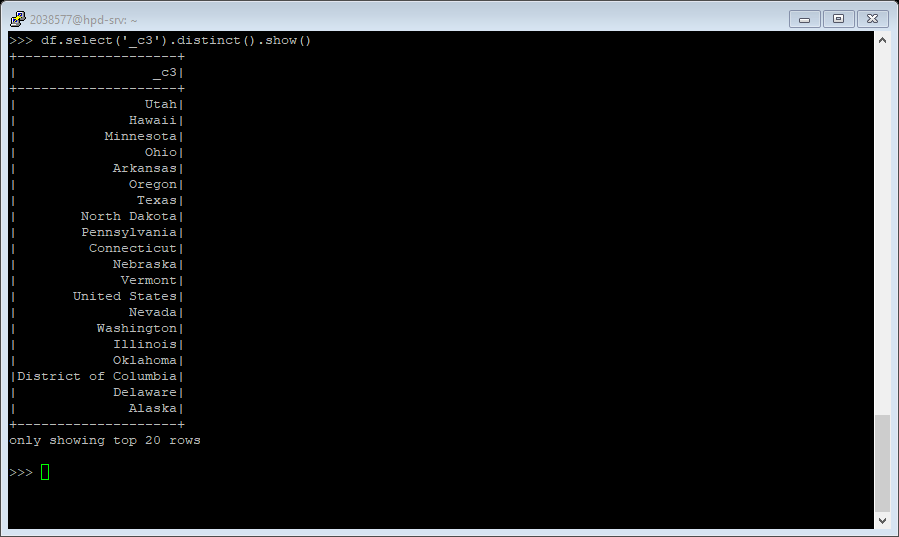
Read CSV

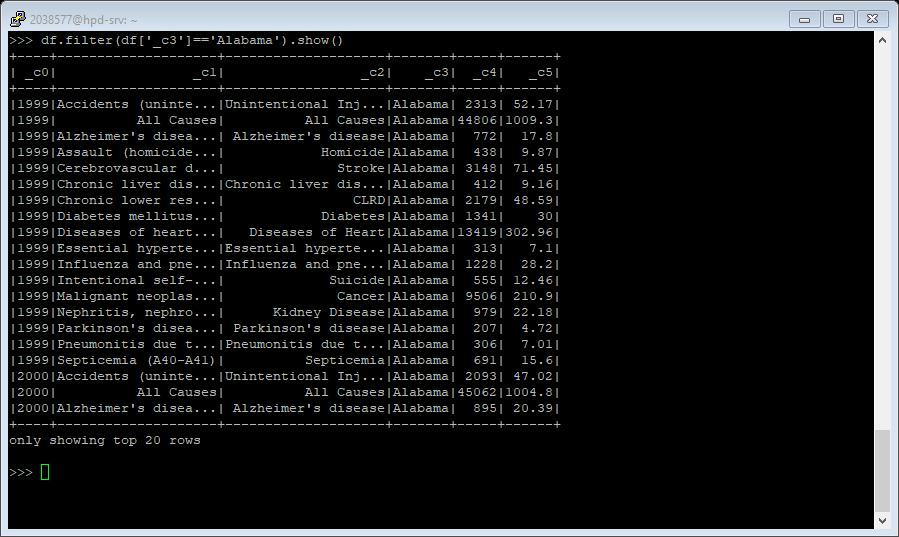
Count how many data present in df:

Showing 10 rows of data:

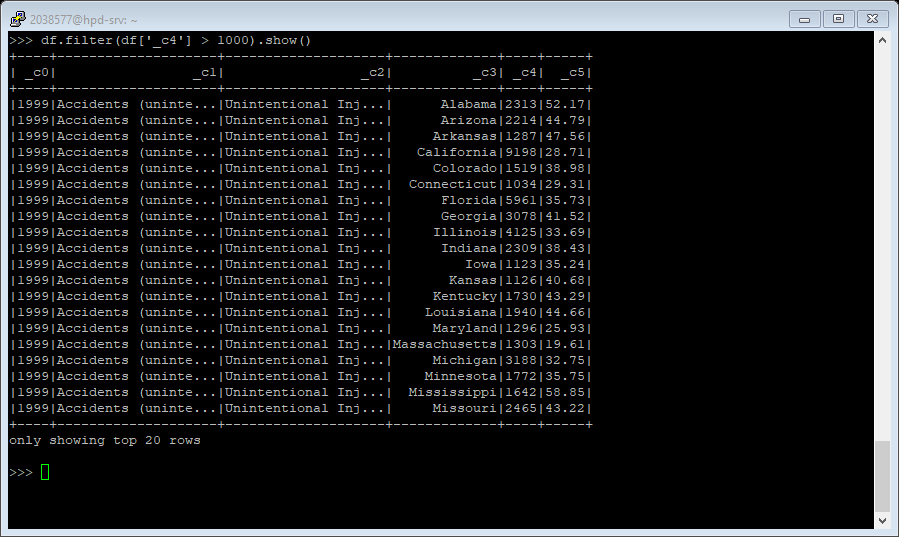
Selecting \_c4 ,\_c5 column and showing its summary:

Showing distinct values present inside of \_c3 column:



Showing only those values which containing state ‘Alabama’:

Showing only those values which has death greater than 1000:



# References

chicco, D. & Jurman, G., 2020. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC Medical Informatics and Decision Making,* 20(1).

DataFlair, 2019. *DataFlair.* [Online]   
Available at: https://data-flair.training/blogs/advantages-of-mongodb/#:~:text=A%20great%20advantage%20of%20MongoDB%20is%20that%20it,data%2C%20you%20can%20distribute%20it%20to%20several%20machines.  
[Accessed 6 4 20221].

DataFlair, 2019. *DataFlair.* [Online]   
Available at: https://data-flair.training/blogs/advantages-of-mongodb/  
[Accessed 6 4 2021].

Educba, 2020. *Educba.* [Online]   
Available at: https://www.educba.com/what-is-oracle/  
[Accessed 22 04 2021].

Guru99, 2021. *Guru99.* [Online]   
Available at: https://www.guru99.com/what-is-mongodb.html  
[Accessed 22 04 2021].

Human Development Reports, 2020. *UNITED NATIONS DEVELOPMENT PROGRAMME.* [Online]   
Available at: http://hdr.undp.org/en/indicators/137506  
[Accessed 21 4 2021].

Juneja, A. & Das, N. N., 2019. Big Data Quality Framework: Pre-Processing Data in Weather Monitoring Application. *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon).*

Malavika, 2019. *Medium.* [Online]   
Available at: https://medium.com/@shireesha.tibacademy/advantages-disadvantages-of-oracle-sql-1cf77e7022d1  
[Accessed 5 4 2021].

Singh, H., 2020. *The NineHertz.* [Online]   
Available at: https://theninehertz.com/blog/advantages-of-using-oracle-database  
[Accessed 6 4 2021].

Stevepaul, 2015. *Mindsmapped.* [Online]   
Available at: https://www.mindsmapped.com/hadoop-advantages-and-disadvantages/  
[Accessed 6 4 2021].

WHO, 2018. *World Health Organization.* [Online]   
Available at: https://www.who.int/health-topics/suicide#tab=tab\_1  
[Accessed 21 04 2021].

World Bank, 2018. *DataBank World Development Indicators.* [Online]   
Available at: http://databank.worldbank.org/data/source/world-development-indicators#  
[Accessed 21 04 2021].