MANAGEMENT, ACCESS & USE OF BIG DATA

PROJECT A

Working with National Weather Service Data

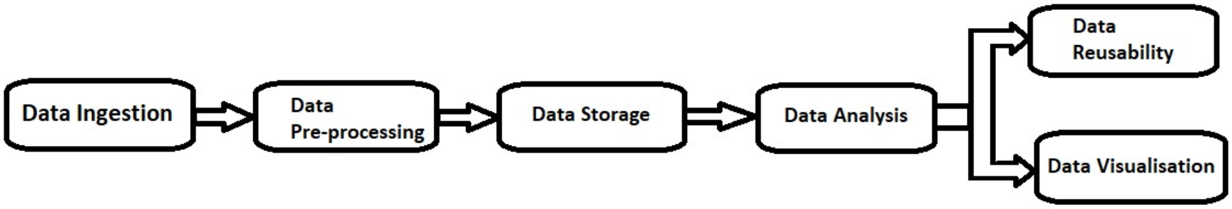
Ojaas Hampiholi

11 March 2020

Introduction:

The project deals with data from the National Weather Service. This project aims to find the trends for various types of natural calamities that occur within the US and to find a pattern between them, if any exists. The data size is very large and has many records. In this project we use a docker allocated to us on Jetstream Indiana University Website. The docker is based on Linux OS and needs basic understanding of Linux for us to be able to do the relevant work. The project aims to design and execute a data pipeline in accordance with the data lifecycle steps that are followed in any big data architecture. We must follow a pipeline execution of various Big Data Architecture steps in order to get results from the data. The pipeline used in this project and the queries as well as the visualization have been explained in detail in the following sections.

Pipeline Architecture and Steps:



Data Ingestion:

This is the first and the most basic stage of any pipeline. In this step we must download the relevant data from the NOAA website (References: Point number 3). This data is open source and freely available to anyone who wishes to pursue applications or projects related to Big Data Analysis and Visualization. These files are downloaded to a landing directory that is created on the Virtual Machine. The data from the website is downloaded in a GZ compressed format. The GZ compression algorithm is used to compress very large files in order to facilitate ease of transfer and recovery of large data files. In order to use the data however, we need the data to be in a csv format, which is easily interpreted by the computer system. Hence, we need to convert the GZ files to csv files. In order to extract the csv from the gz files we use the python utilities script which extracts the files from the landing directory and stores them to the extraction directory.

Data Pre-processing:

The next main step in any pipeline is the pre-processing of the data to get into a format which can be used by the applications in order to analyze the data and get results. In this project we use MongoDB to analyze the data. MongoDB uses JSON format to store and analyze any object while the data that we extracted is in csv format. Hence, we need to convert the data to a JSON object before it can be stored or used in the MongoDB. This conversion is done using the Pandas library available in python. Pandas data frame helps us to select the data columns that need to be stored into the JSON object and helps in converting the data.

Data Storage:

After we have converted the csv files to the JSON format, we can store the files into MongoDB. But, in this project we want to use python alongside MongoDB. Hence, we decided to use PyMongo. PyMongo is a python distribution which contains specialized tools to work on MongoDB in conjugation with python. We launch PyMongo in order to load the data to the database and make it ready for further use.

Data Analysis:

In this stage we have loaded the data in relevant format and made sure that the data is ready for use. Hence, in this stage we apply various queries on the data in order to get results. We performed some basic operations like adding a record to the database, modifying an existing record and deleting a record as well as deleting a group of records. The next thing we did was running queries on the dataset.

Data Reusability:

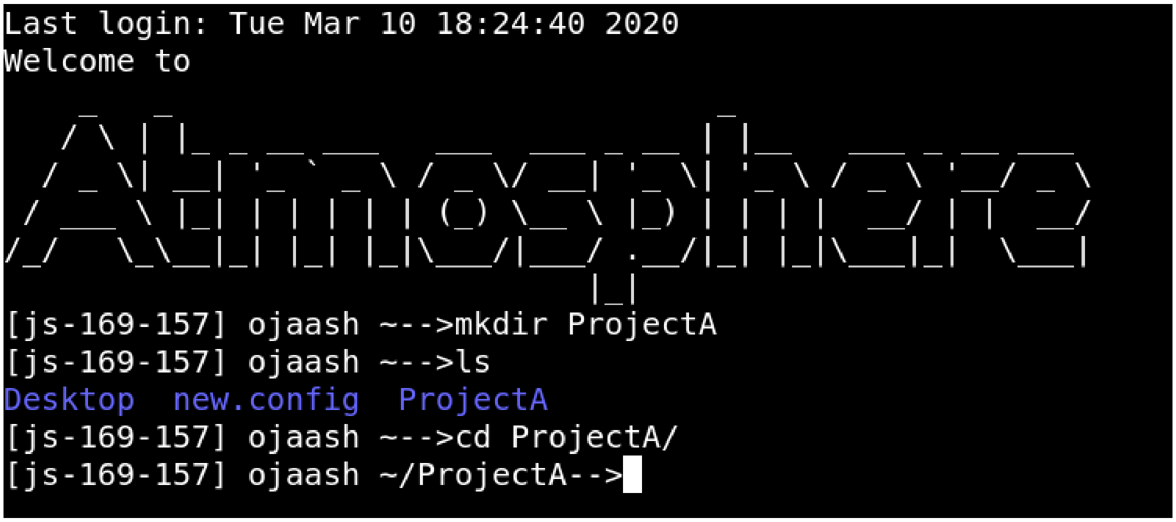
The output of the query was stored into a csv file. This ensures that the relevant data can be retrieved later without having to run the query all over again. Also, converting the output of a query into csv file ensures that the file can be sent to any other system for further analysis and visualization.

Data Visualization:

In this step, the csv file generated in the data reusability step is shifted to our local machine for visualization. I have used Pandas library from python as well as RStudio for the visualization part of this project.

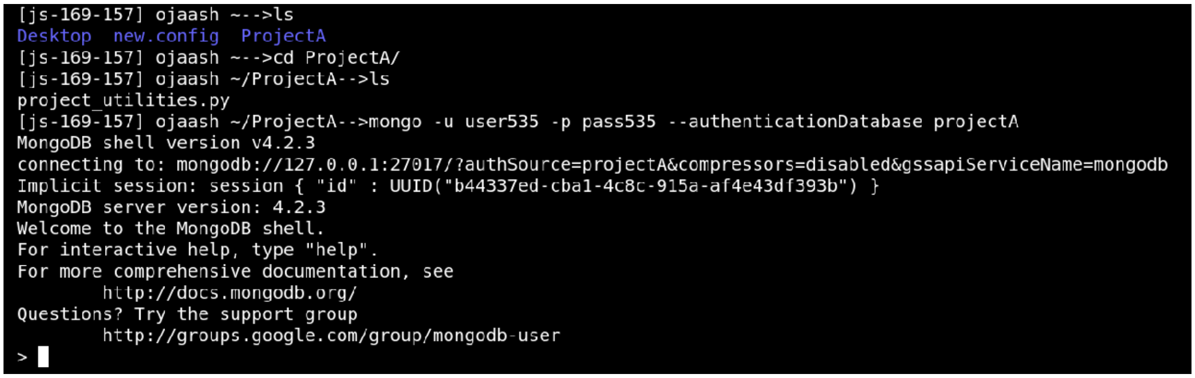
Pipeline Execution (By Steps):

Creating a directory for the Project:

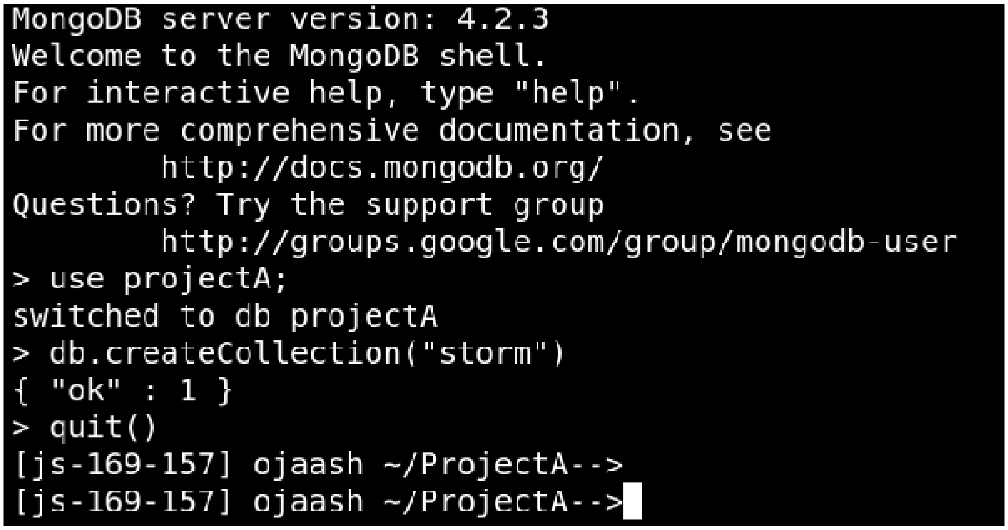


```
Last login: Tue Mar 10 18:24:40 2020
Welcome to
    _   _                             _
   / \ | |_ _ __ ___   ___  ___ _ __ | |__   ___ _ __ ___
  / _ \| __| '_ ` _ \ / _ \/ __| '_ \| '_ \ / _ \ '__/ _ \
 / ___ \ |_| | | | | | (_) \__ \ |_) | | | |  __/ | |  __/
/_/   \_\__|_| |_| |_|\___/|___/ .__/|_| |_|\___|_|  \___|
                               |_|
[js-169-157] ojaash --->mkdir ProjectA
[js-169-157] ojaash --->ls
Desktop  new.config  ProjectA
[js-169-157] ojaash --->cd ProjectA/
[js-169-157] ojaash ~/ProjectA-->
```

Logging into Mongo Shell:

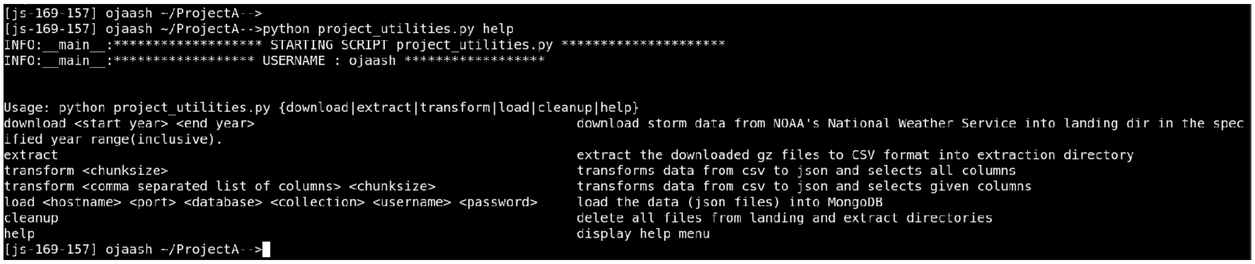


```
[js-169-157] ojaash --->ls
Desktop  new.config  ProjectA
[js-169-157] ojaash --->cd ProjectA/
[js-169-157] ojaash ~/ProjectA-->ls
project_utilities.py
[js-169-157] ojaash ~/ProjectA-->mongo -u user535 -p pass535 --authenticationDatabase projectA
MongoDB shell version v4.2.3
connecting to: mongodb://127.0.0.1:27017/?authSource=projectA&compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("b44337ed-cba1-4c8c-915a-af4e43df393b") }
MongoDB server version: 4.2.3
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
> 
```



```
MongoDB server version: 4.2.3
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
> use projectA;
switched to db projectA
> db.createCollection("storm")
{ "ok" : 1 }
> quit()
[js-169-157] ojaash ~/ProjectA-->
[js-169-157] ojaash ~/ProjectA-->
```

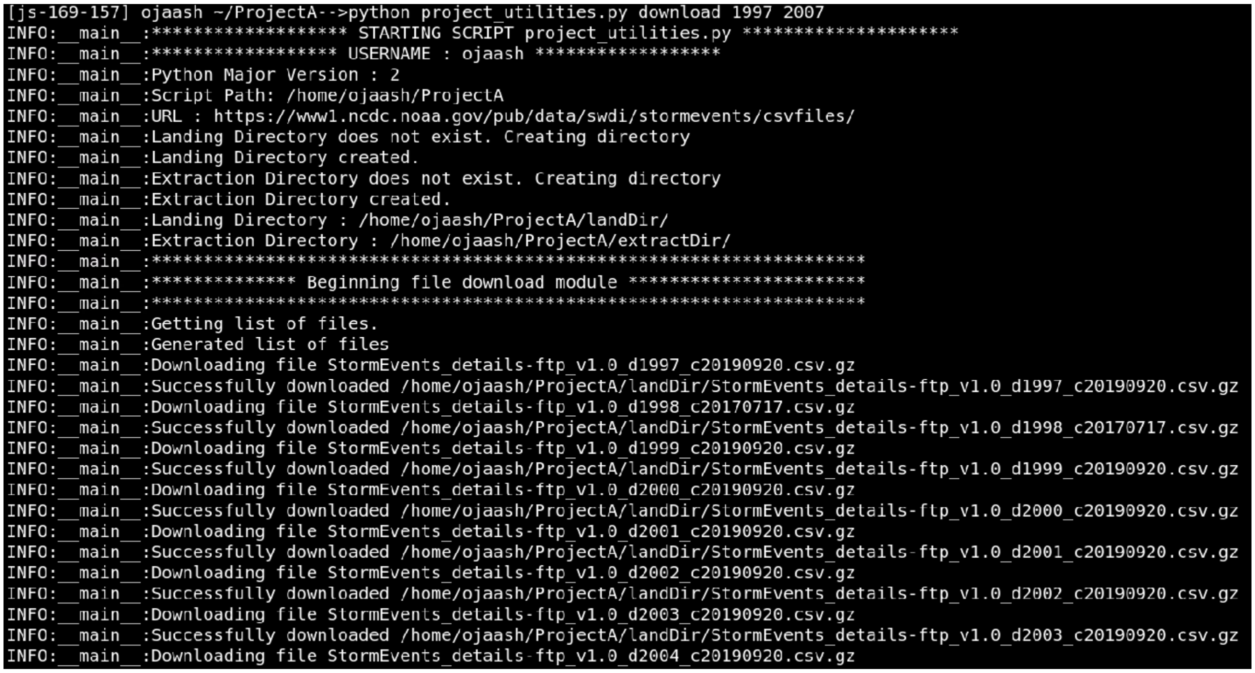
Instructions on how to use project_utilities.py file:



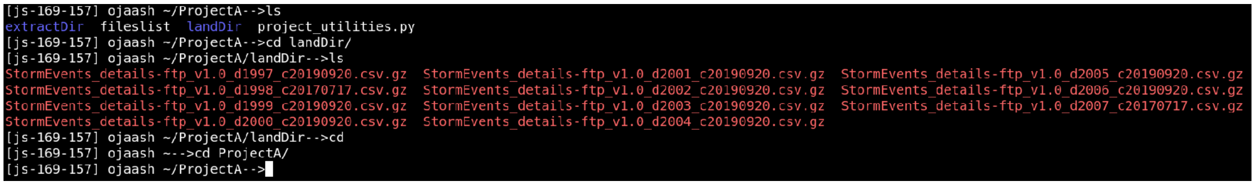
```
[js-169-157] ojaash ~/ProjectA-->
[js-169-157] ojaash ~/ProjectA-->python project_utilities.py help
INFO:__main__:******************* STARTING SCRIPT project_utilities.py *********************
INFO:__main__:****************** USERNAME : ojaash ******************


Usage: python project_utilities.py {download|extract|transform|load|cleanup|help}
download <start year> <end year>                                         download storm data from NOAA's National Weather Service into landing dir in the spec
ified year range(inclusive).
extract                                                                  extract the downloaded gz files to CSV format into extraction directory
transform <chunksize>                                                    transforms data from csv to json and selects all columns
transform <comma separated list of columns> <chunksize>                  transforms data from csv to json and selects given columns
load <hostname> <port> <database> <collection> <username> <password>     load the data (json files) into MongoDB
cleanup                                                                  delete all files from landing and extract directories
help                                                                     display help menu
[js-169-157] ojaash ~/ProjectA-->
```

Downloading the files from the year range 1997-2007:

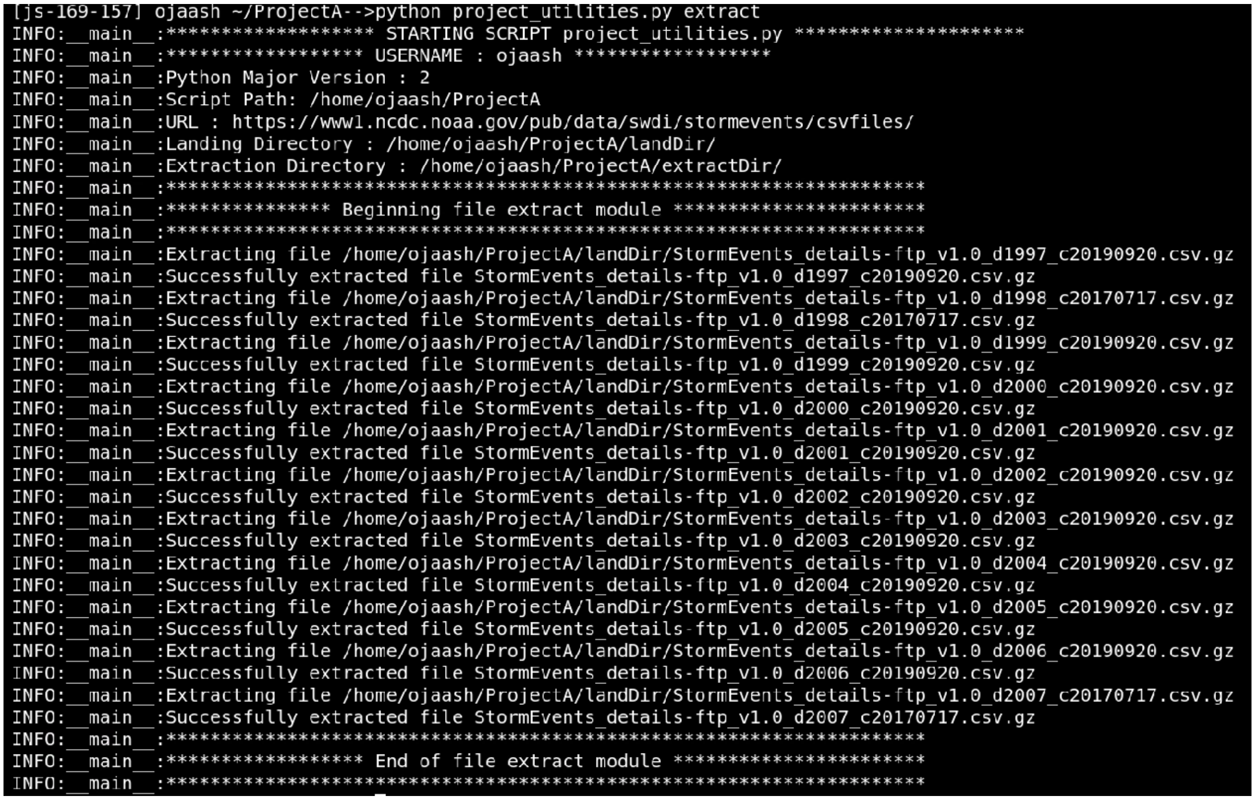


```
[js-169-157] ojaash ~/ProjectA-->python project_utilities.py download 1997 2007
INFO:__main__:******************* STARTING SCRIPT project_utilities.py *********************
INFO:__main__:****************** USERNAME : ojaash ******************
INFO:__main__:Python Major Version : 2
INFO:__main__:Script Path: /home/ojaash/ProjectA
INFO:__main__:URL : https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/
INFO:__main__:Landing Directory does not exist. Creating directory
INFO:__main__:Landing Directory created.
INFO:__main__:Extraction Directory does not exist. Creating directory
INFO:__main__:Extraction Directory created.
INFO:__main__:Landing Directory : /home/ojaash/ProjectA/landDir/
INFO:__main__:Extraction Directory : /home/ojaash/ProjectA/extractDir/
INFO:__main__:**********************************************************************
INFO:__main__:************** Beginning file download module ************************
INFO:__main__:**********************************************************************
INFO:__main__:Getting list of files.
INFO:__main__:Generated list of files
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d1997_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1997_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d1998_c20170717.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1998_c20170717.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d1999_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1999_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d2000_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2000_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d2001_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2001_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d2002_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2002_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d2003_c20190920.csv.gz
INFO:__main__:Successfully downloaded /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2003_c20190920.csv.gz
INFO:__main__:Downloading file StormEvents_details-ftp_v1.0_d2004_c20190920.csv.gz
```



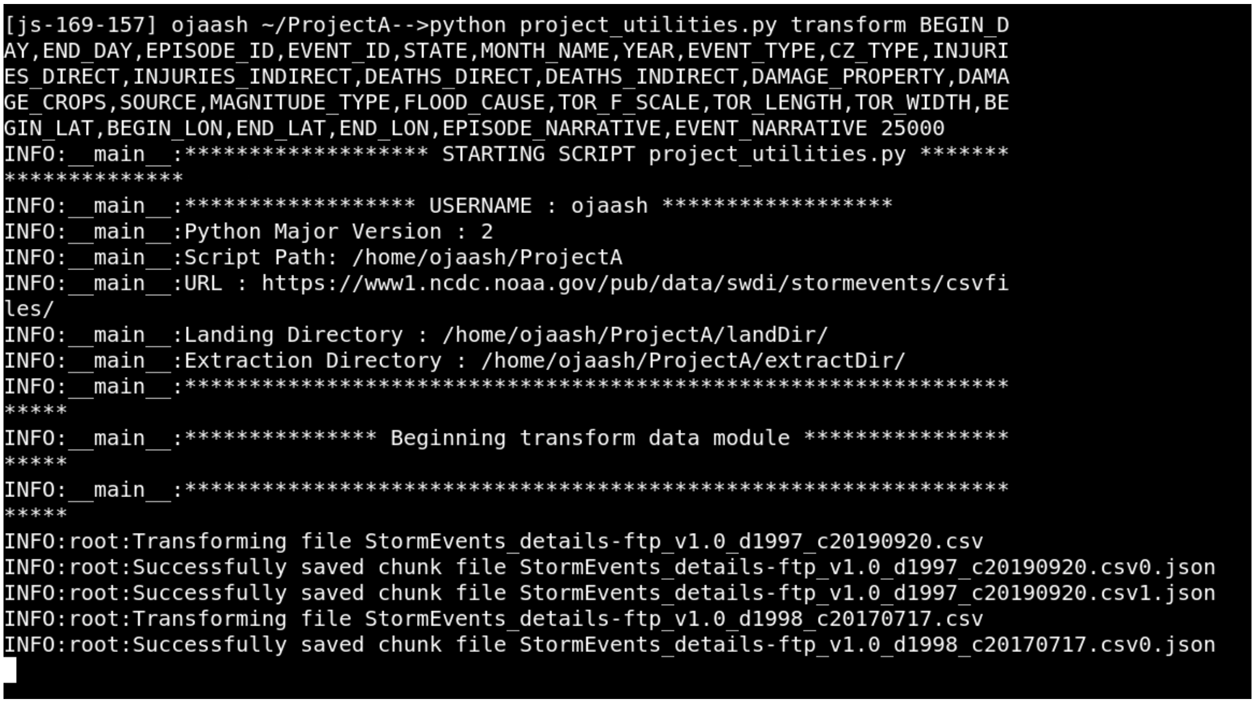
```
[js-169-157] ojaash ~/ProjectA-->ls
extractDir  fileslist  landDir  project_utilities.py
[js-169-157] ojaash ~/ProjectA-->cd landDir/
[js-169-157] ojaash ~/ProjectA/landDir-->ls
StormEvents_details-ftp_v1.0_d1997_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2001_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2005_c20190920.csv.gz
StormEvents_details-ftp_v1.0_d1998_c20170717.csv.gz  StormEvents_details-ftp_v1.0_d2002_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2006_c20190920.csv.gz
StormEvents_details-ftp_v1.0_d1999_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2003_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2007_c20170717.csv.gz
StormEvents_details-ftp_v1.0_d2000_c20190920.csv.gz  StormEvents_details-ftp_v1.0_d2004_c20190920.csv.gz
[js-169-157] ojaash ~/ProjectA/landDir-->cd
[js-169-157] ojaash ~-->cd ProjectA/
[js-169-157] ojaash ~/ProjectA-->
```

Extracting CSV files from GZ files:



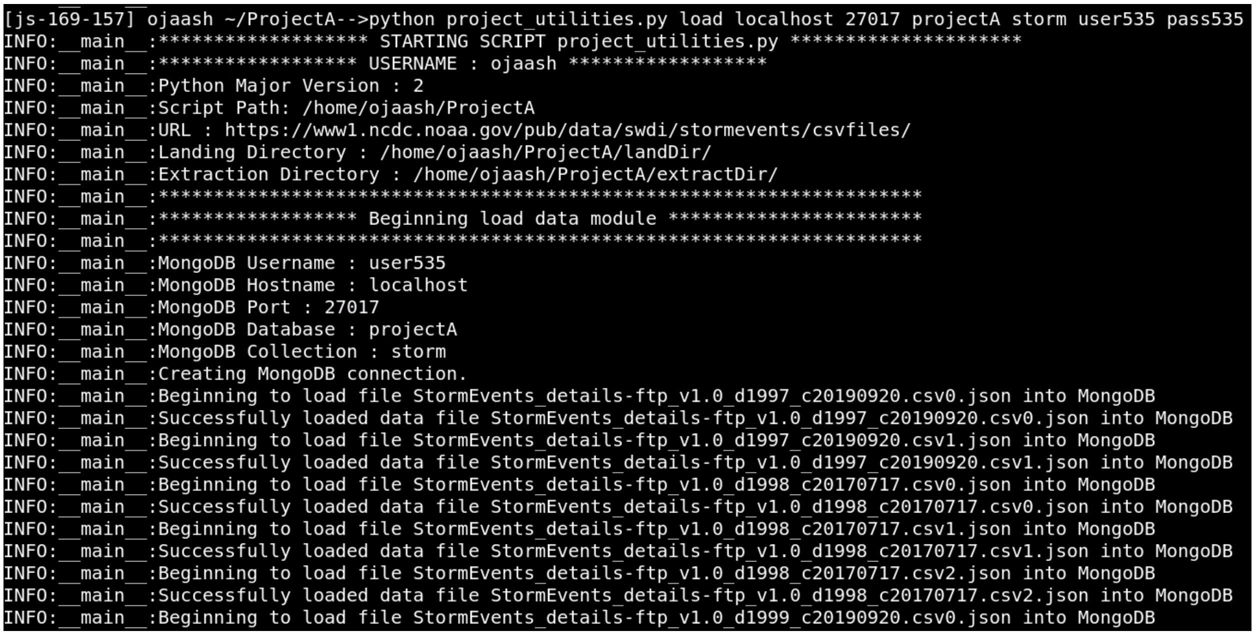
```
[js-169-157] ojaash ~/ProjectA-->python project_utilities.py extract
INFO:__main__:******************* STARTING SCRIPT project_utilities.py *********************
INFO:__main__:****************** USERNAME : ojaash ******************
INFO:__main__:Python Major Version : 2
INFO:__main__:Script Path: /home/ojaash/ProjectA
INFO:__main__:URL : https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/
INFO:__main__:Landing Directory : /home/ojaash/ProjectA/landDir/
INFO:__main__:Extraction Directory : /home/ojaash/ProjectA/extractDir/
INFO:__main__:*********************************************************************
INFO:__main__:*************** Beginning file extract module ***********************
INFO:__main__:*********************************************************************
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1997_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d1997_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1998_c20170717.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d1998_c20170717.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d1999_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d1999_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2000_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2000_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2001_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2001_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2002_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2002_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2003_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2003_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2004_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2004_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2005_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2005_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2006_c20190920.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2006_c20190920.csv.gz
INFO:__main__:Extracting file /home/ojaash/ProjectA/landDir/StormEvents_details-ftp_v1.0_d2007_c20170717.csv.gz
INFO:__main__:Successfully extracted file StormEvents_details-ftp_v1.0_d2007_c20170717.csv.gz
INFO:__main__:*********************************************************************
INFO:__main__:****************** End of file extract module ***********************
INFO:__main__:*********************************************************************
```

Transforming the csv files to corresponding JSON Objects:



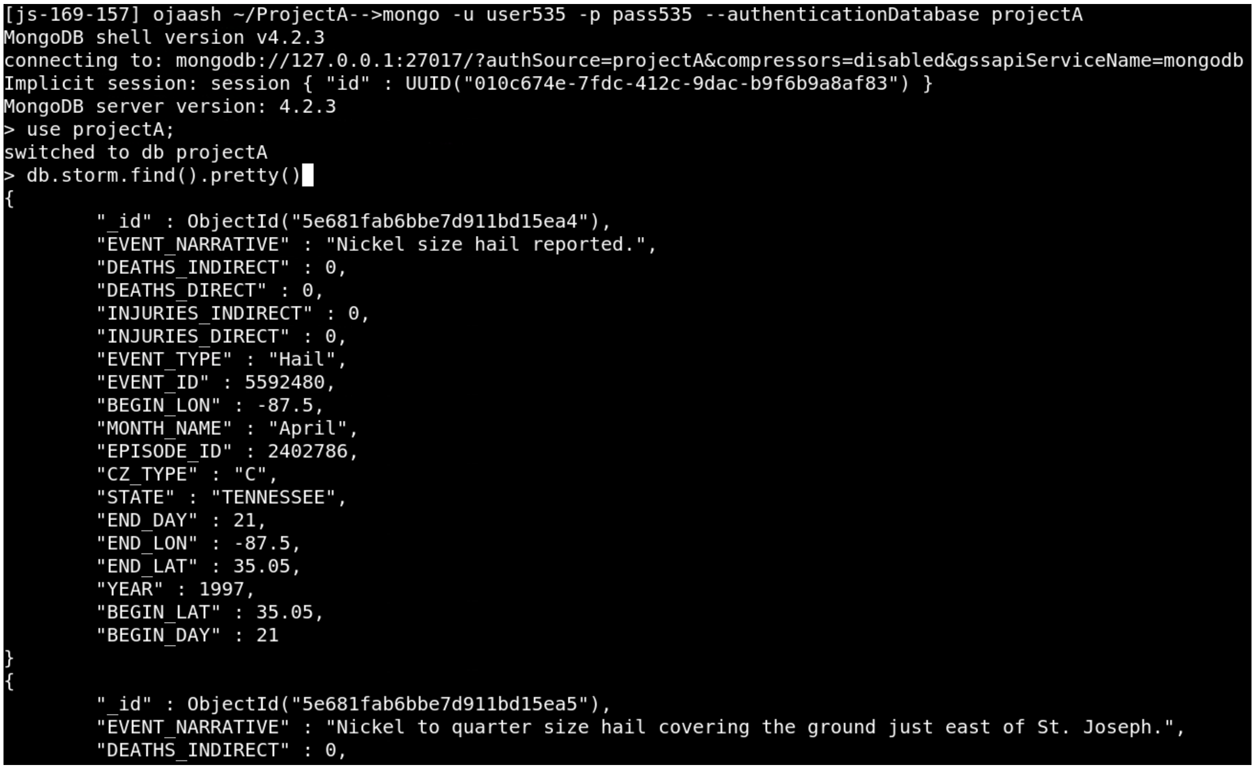
```
[js-169-157] ojaash ~/ProjectA-->python project_utilities.py transform BEGIN_D
AY,END_DAY,EPISODE_ID,EVENT_ID,STATE,MONTH_NAME,YEAR,EVENT_TYPE,CZ_TYPE,INJURI
ES_DIRECT,INJURIES_INDIRECT,DEATHS_DIRECT,DEATHS_INDIRECT,DAMAGE_PROPERTY,DAMA
GE_CROPS,SOURCE,MAGNITUDE_TYPE,FLOOD_CAUSE,TOR_F_SCALE,TOR_LENGTH,TOR_WIDTH,BE
GIN_LAT,BEGIN_LON,END_LAT,END_LON,EPISODE_NARRATIVE,EVENT_NARRATIVE 25000
INFO:__main__:******************* STARTING SCRIPT project_utilities.py *******
**************
INFO:__main__:****************** USERNAME : ojaash ******************
INFO:__main__:Python Major Version : 2
INFO:__main__:Script Path: /home/ojaash/ProjectA
INFO:__main__:URL : https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfi
les/
INFO:__main__:Landing Directory : /home/ojaash/ProjectA/landDir/
INFO:__main__:Extraction Directory : /home/ojaash/ProjectA/extractDir/
INFO:__main__:****************************************************************
*****
INFO:__main__:*************** Beginning transform data module ****************
*****
INFO:__main__:****************************************************************
*****
INFO:root:Transforming file StormEvents_details-ftp_v1.0_d1997_c20190920.csv
INFO:root:Successfully saved chunk file StormEvents_details-ftp_v1.0_d1997_c20190920.csv0.json
INFO:root:Successfully saved chunk file StormEvents_details-ftp_v1.0_d1997_c20190920.csv1.json
INFO:root:Transforming file StormEvents_details-ftp_v1.0_d1998_c20170717.csv
INFO:root:Successfully saved chunk file StormEvents_details-ftp_v1.0_d1998_c20170717.csv0.json
```

Loading the data into MongoDB:



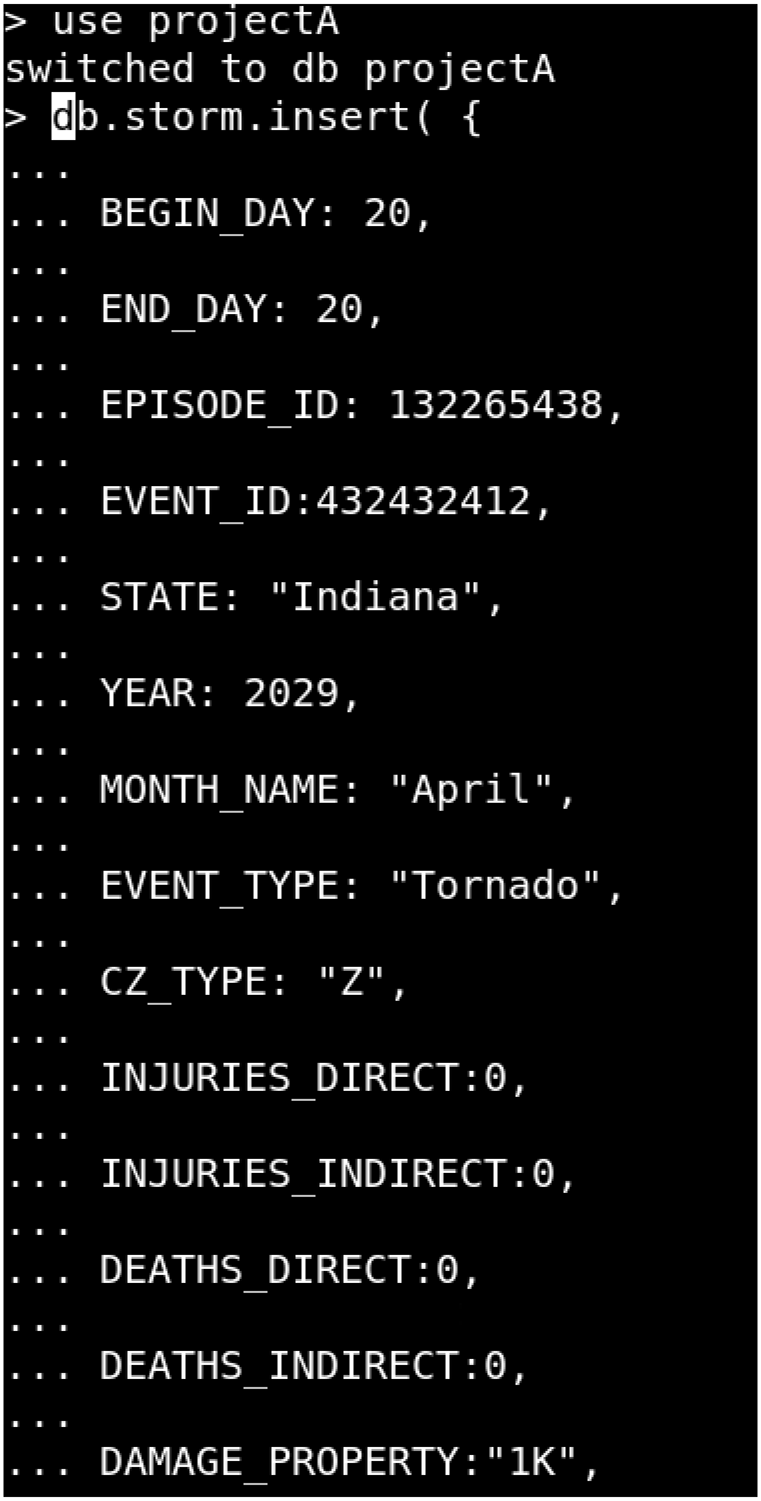
```
[js-169-157] ojaash ~/ProjectA-->python project_utilities.py load localhost 27017 projectA storm user535 pass535
INFO:__main__:******************* STARTING SCRIPT project_utilities.py *********************
INFO:__main__:****************** USERNAME : ojaash ******************
INFO:__main__:Python Major Version : 2
INFO:__main__:Script Path: /home/ojaash/ProjectA
INFO:__main__:URL : https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/
INFO:__main__:Landing Directory : /home/ojaash/ProjectA/landDir/
INFO:__main__:Extraction Directory : /home/ojaash/ProjectA/extractDir/
INFO:__main__:*********************************************************************
INFO:__main__:****************** Beginning load data module ***********************
INFO:__main__:*********************************************************************
INFO:__main__:MongoDB Username : user535
INFO:__main__:MongoDB Hostname : localhost
INFO:__main__:MongoDB Port : 27017
INFO:__main__:MongoDB Database : projectA
INFO:__main__:MongoDB Collection : storm
INFO:__main__:Creating MongoDB connection.
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1997_c20190920.csv0.json into MongoDB
INFO:__main__:Successfully loaded data file StormEvents_details-ftp_v1.0_d1997_c20190920.csv0.json into MongoDB
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1997_c20190920.csv1.json into MongoDB
INFO:__main__:Successfully loaded data file StormEvents_details-ftp_v1.0_d1997_c20190920.csv1.json into MongoDB
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1998_c20170717.csv0.json into MongoDB
INFO:__main__:Successfully loaded data file StormEvents_details-ftp_v1.0_d1998_c20170717.csv0.json into MongoDB
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1998_c20170717.csv1.json into MongoDB
INFO:__main__:Successfully loaded data file StormEvents_details-ftp_v1.0_d1998_c20170717.csv1.json into MongoDB
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1998_c20170717.csv2.json into MongoDB
INFO:__main__:Successfully loaded data file StormEvents_details-ftp_v1.0_d1998_c20170717.csv2.json into MongoDB
INFO:__main__:Beginning to load file StormEvents_details-ftp_v1.0_d1999_c20190920.csv0.json into MongoDB
```

Checking status of files in the Database:



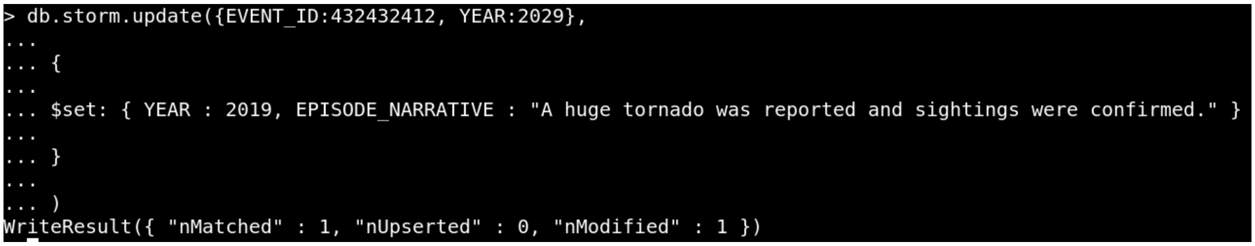
```
[js-169-157] ojaash ~/ProjectA-->mongo -u user535 -p pass535 --authenticationDatabase projectA
MongoDB shell version v4.2.3
connecting to: mongodb://127.0.0.1:27017/?authSource=projectA&compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("010c674e-7fdc-412c-9dac-b9f6b9a8af83") }
MongoDB server version: 4.2.3
> use projectA;
switched to db projectA
> db.storm.find().pretty()
{
        "_id" : ObjectId("5e681fab6bbe7d911bd15ea4"),
        "EVENT_NARRATIVE" : "Nickel size hail reported.",
        "DEATHS_INDIRECT" : 0,
        "DEATHS_DIRECT" : 0,
        "INJURIES_INDIRECT" : 0,
        "INJURIES_DIRECT" : 0,
        "EVENT_TYPE" : "Hail",
        "EVENT_ID" : 5592480,
        "BEGIN_LON" : -87.5,
        "MONTH_NAME" : "April",
        "EPISODE_ID" : 2402786,
        "CZ_TYPE" : "C",
        "STATE" : "TENNESSEE",
        "END_DAY" : 21,
        "END_LON" : -87.5,
        "END_LAT" : 35.05,
        "YEAR" : 1997,
        "BEGIN_LAT" : 35.05,
        "BEGIN_DAY" : 21
}
{
        "_id" : ObjectId("5e681fab6bbe7d911bd15ea5"),
        "EVENT_NARRATIVE" : "Nickel to quarter size hail covering the ground just east of St. Joseph.",
        "DEATHS_INDIRECT" : 0,
```

Adding a new record to the collection:



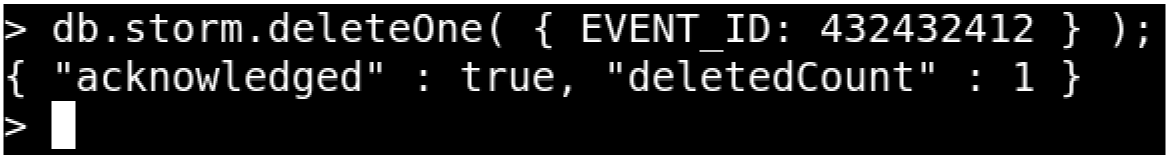
```
> use projectA
switched to db projectA
> db.storm.insert( {
...
... BEGIN_DAY: 20,
...
... END_DAY: 20,
...
... EPISODE_ID: 132265438,
...
... EVENT_ID:432432412,
...
... STATE: "Indiana",
...
... YEAR: 2029,
...
... MONTH_NAME: "April",
...
... EVENT_TYPE: "Tornado",
...
... CZ_TYPE: "Z",
...
... INJURIES_DIRECT:0,
...
... INJURIES_INDIRECT:0,
...
... DEATHS_DIRECT:0,
...
... DEATHS_INDIRECT:0,
...
... DAMAGE_PROPERTY:"1K",
```

Modifying an existing record:

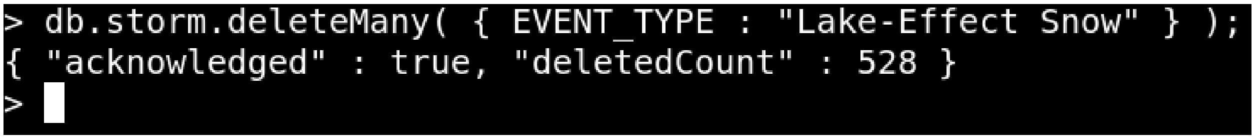


```
> db.storm.update({EVENT_ID:432432412, YEAR:2029},
...
... {
...
... $set: { YEAR : 2019, EPISODE_NARRATIVE : "A huge tornado was reported and sightings were confirmed." }
...
... }
...
... )
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

Deleting records from the collection:

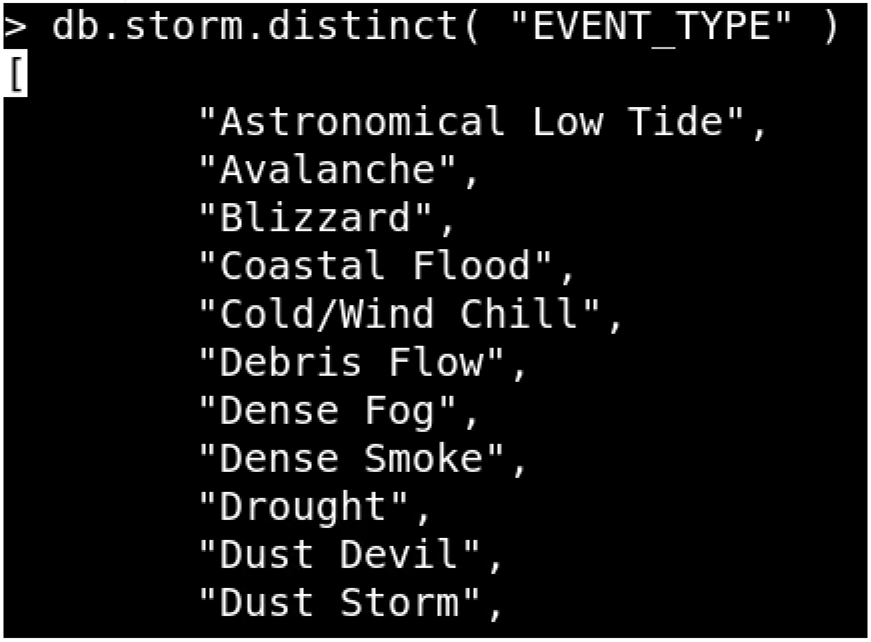


```
> db.storm.deleteOne( { EVENT_ID: 432432412 } );
{ "acknowledged" : true, "deletedCount" : 1 }
> 
```

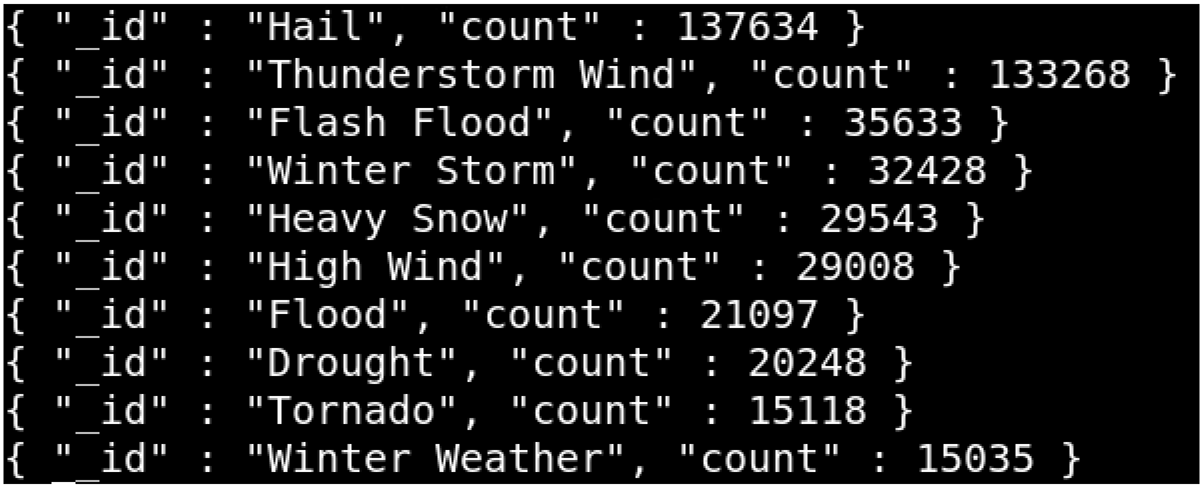


```
> db.storm.deleteMany( { EVENT_TYPE : "Lake-Effect Snow" } );
{ "acknowledged" : true, "deletedCount" : 528 }
> 
```

Querying records from the collections:

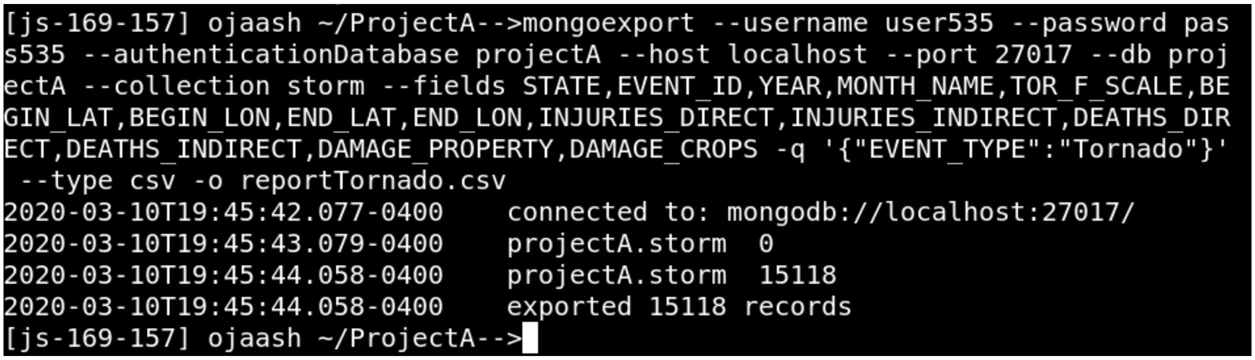


```
> db.storm.distinct( "EVENT_TYPE" )
[
        "Astronomical Low Tide",
        "Avalanche",
        "Blizzard",
        "Coastal Flood",
        "Cold/Wind Chill",
        "Debris Flow",
        "Dense Fog",
        "Dense Smoke",
        "Drought",
        "Dust Devil",
        "Dust Storm",
```

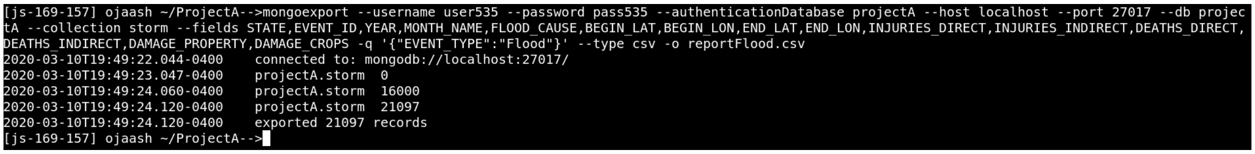


```
{ "_id" : "Hail", "count" : 137634 }
{ "_id" : "Thunderstorm Wind", "count" : 133268 }
{ "_id" : "Flash Flood", "count" : 35633 }
{ "_id" : "Winter Storm", "count" : 32428 }
{ "_id" : "Heavy Snow", "count" : 29543 }
{ "_id" : "High Wind", "count" : 29008 }
{ "_id" : "Flood", "count" : 21097 }
{ "_id" : "Drought", "count" : 20248 }
{ "_id" : "Tornado", "count" : 15118 }
{ "_id" : "Winter Weather", "count" : 15035 }
```

Exporting the queried records to a csv file:



```
[js-169-157] ojaash ~/ProjectA-->mongoexport --username user535 --password pas
s535 --authenticationDatabase projectA --host localhost --port 27017 --db proj
ectA --collection storm --fields STATE,EVENT_ID,YEAR,MONTH_NAME,TOR_F_SCALE,BE
GIN_LAT,BEGIN_LON,END_LAT,END_LON,INJURIES_DIRECT,INJURIES_INDIRECT,DEATHS_DIR
ECT,DEATHS_INDIRECT,DAMAGE_PROPERTY,DAMAGE_CROPS -q '{"EVENT_TYPE":"Tornado"}'
 --type csv -o reportTornado.csv
2020-03-10T19:45:42.077-0400    connected to: mongodb://localhost:27017/
2020-03-10T19:45:43.079-0400    projectA.storm  0
2020-03-10T19:45:44.058-0400    projectA.storm  15118
2020-03-10T19:45:44.058-0400    exported 15118 records
[js-169-157] ojaash ~/ProjectA-->
```



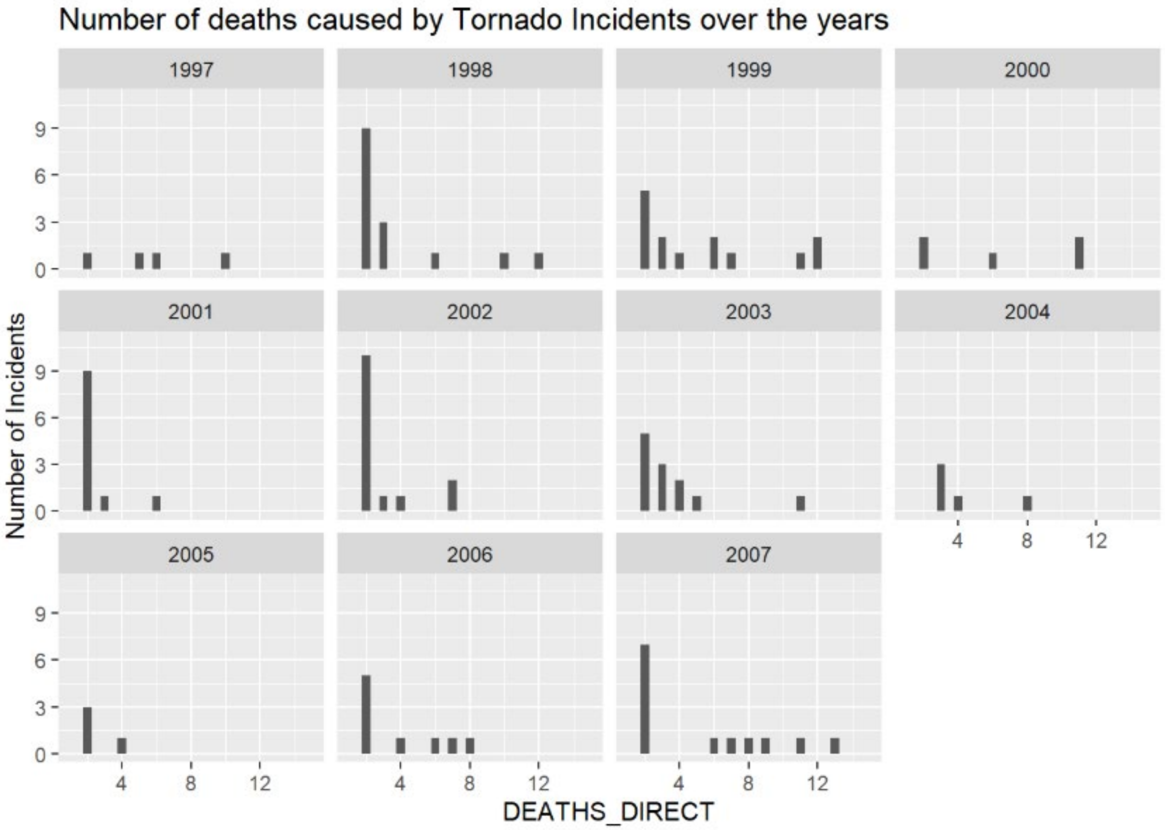
```
[js-169-157] ojaash ~/ProjectA-->mongoexport --username user535 --password pass535 --authenticationDatabase projectA --host localhost --port 27017 --db projec
tA --collection storm --fields STATE,EVENT_ID,YEAR,MONTH_NAME,FLOOD_CAUSE,BEGIN_LAT,BEGIN_LON,END_LAT,END_LON,INJURIES_DIRECT,INJURIES_INDIRECT,DEATHS_DIRECT,
DEATHS_INDIRECT,DAMAGE_PROPERTY,DAMAGE_CROPS -q '{"EVENT_TYPE":"Flood"}' --type csv -o reportFlood.csv
2020-03-10T19:49:22.044-0400    connected to: mongodb://localhost:27017/
2020-03-10T19:49:23.047-0400    projectA.storm  0
2020-03-10T19:49:24.060-0400    projectA.storm  16000
2020-03-10T19:49:24.120-0400    projectA.storm  21097
2020-03-10T19:49:24.120-0400    exported 21097 records
[js-169-157] ojaash ~/ProjectA-->
```

The pipeline steps implemented above follow the major steps required in any data lifecycle. The essential steps that were followed were:

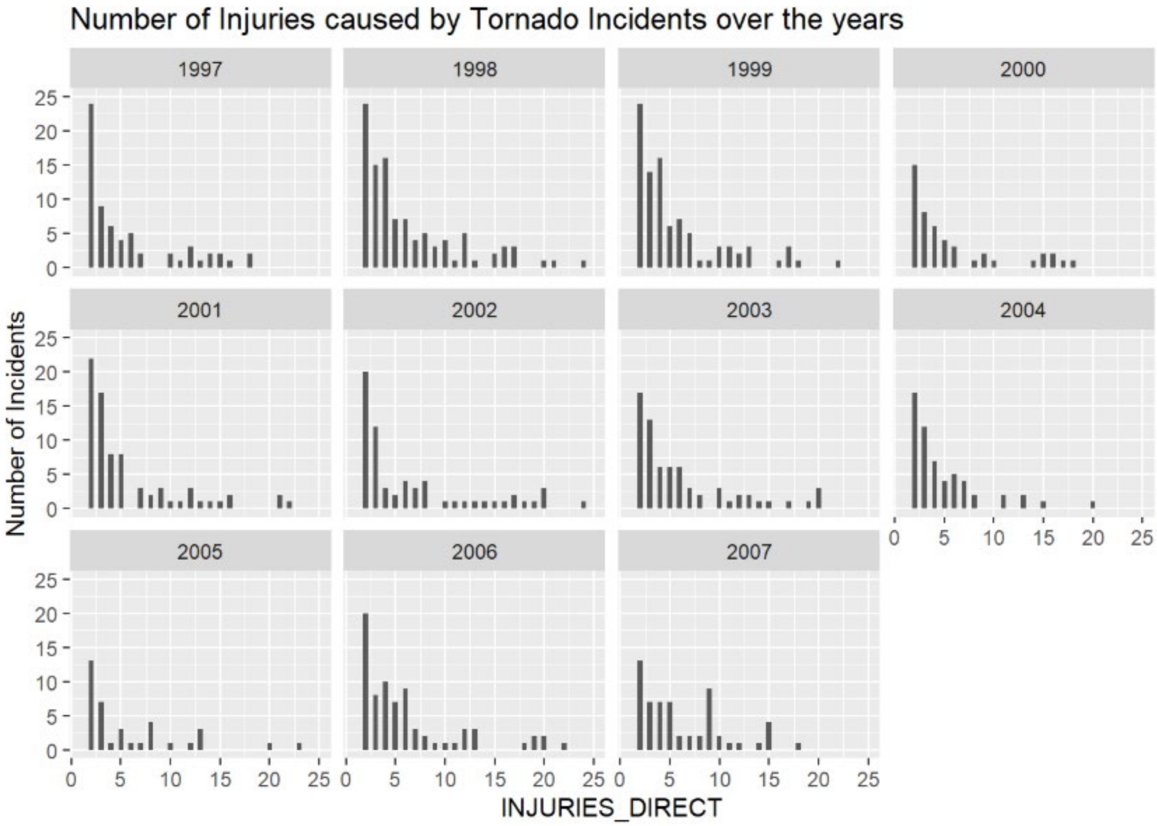
1. Creating a VM instance for the project.
2. Creating a folder for the project.
3. Creating a Mongo Database in the folder created above.
4. Loading the python utilities file in the folder and seeing instructions on how to run the file.
5. Downloading the files by giving the year range and checking if the files appear in the landing directory.
6. Extracting csv files from the GZ files.
7. Converting the csv files to JSON objects so that they can be accessed using MongoDB.
8. Loading the data into MongoDB and checking the status of data in the database,
9. Adding, modifying and deleting some records from the database.
10. Exporting the queried records to a csv file (which is then transferred to local machine using WinSCP).
11. Applying Visualization techniques to the data (Using Python Pandas and R ggplot2 libraries).

Visualization:

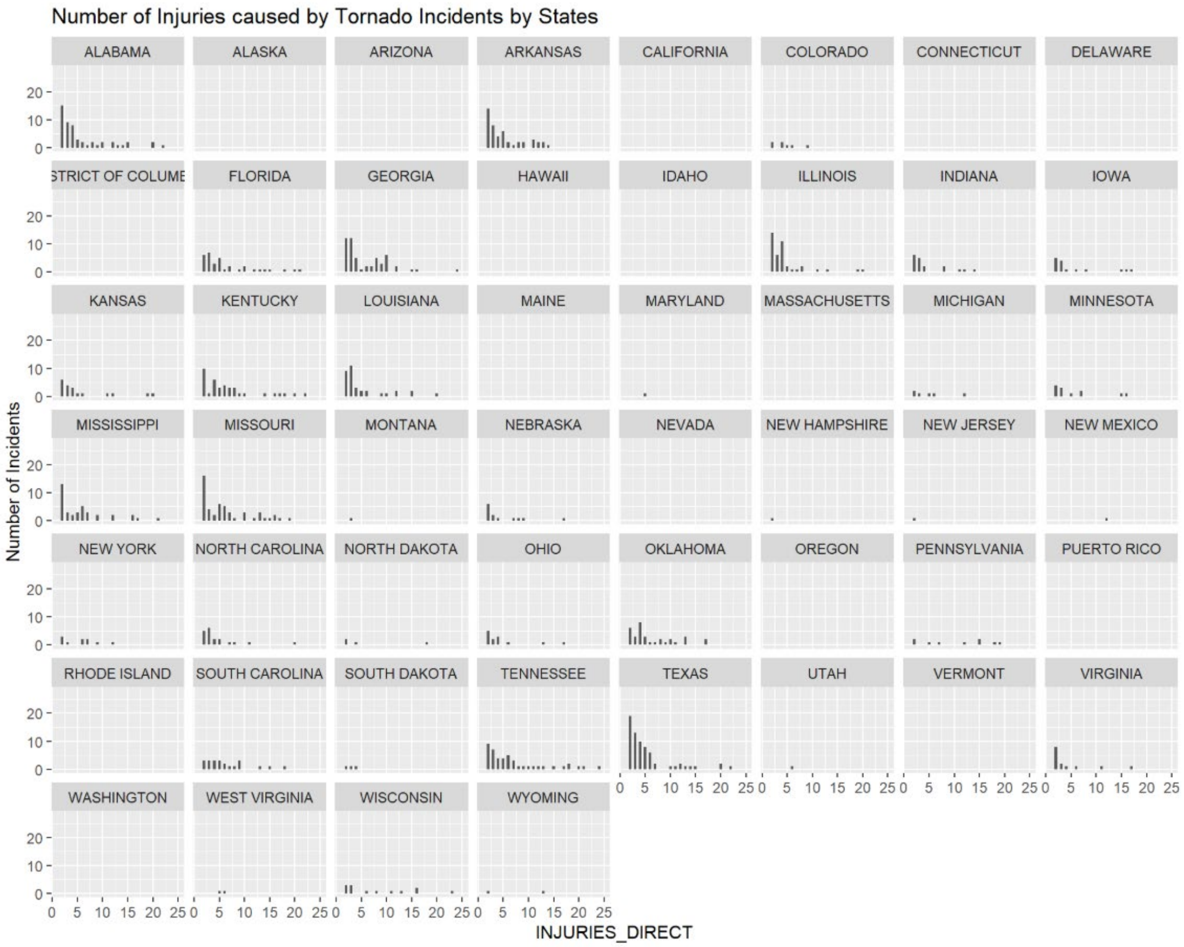
For the file reportTornado.csv:



From the graph we can see that the most storms led to very low mortality rates in almost all years. But the number of deaths for some of the storms is very high for the year 1999. This implies that the storms in 1999 were more severe as compared to other years.

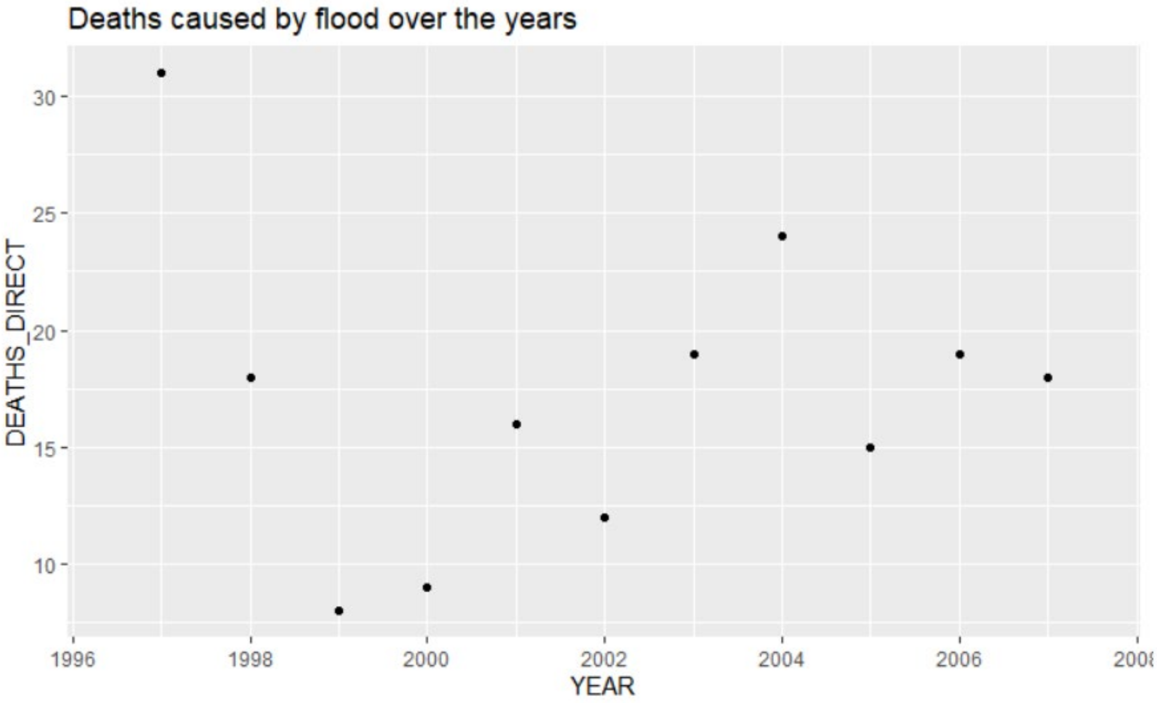


The number of injuries caused by most of the tornados is usually below 5 in all the years, but in the year 2007 a lot of cases were observed where the number of injuries was high. This supports the previous conclusion that the severity of storm was higher in 2007.



This graph helps us to visualize that the storms are concentrated in a few states and not actually spread across the entire country. The major affected states in the country are Alabama, Arkansas, Florida, Georgia, Illinois, Kentucky, Louisiana, Tennessee and Texas.

For the file reportFlood.csv:



The graph plotted above shows that the number of death incidents caused by flood incidents were very high in the year 1997 as compared to any other years.

Challenges Faced:

Some of the major challenges that I faced during the project were:

1. Transferring the python utilities file from local machine to the VM – I read the instruction on FTP/SFTP from the Jetstream website in order to transfer the file to VM. Clicking Ctrl + Alt + Shift opens the option to search for the directories of the VM and to upload any file from the local machine to VM.
2. Chunk size during the Transform step: Keeping a smaller chunk size leads to a lot of csv files being created and hence a lot of corresponding JSON objects are created.

Conclusion:

The project was done using a wide array of tools that are used at various stages of the pipeline. The tools that I have used in this project vary from python for conversion of data, MongoDB for the storage, analysis and querying of the data and finally RStudio for the exploratory analysis of the csv file. All those tools along with the data pipeline form the data lifecycle in general. After doing the project, I got a lot more comfortable with using Linux in general as well as had a hands-on experience with using MongoDB to store and assess various types of data in a non-relational database environment. The visualizations that are attached in the report and as a part of the subsidiary data gave a very good learning curve to my exploratory analysis skills. The project helped me to identify the trend in the storms according to the years as well as according to the states.

References:

1. Canvas Project Instructions - https://iu.instructure.com/courses/1859908/assignments/10091034?module_item_id=19873345
2. Canvas utilities python file - https://iu.instructure.com/courses/1859908/files/folder/Project%20A
3. NOAA Website - https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/
4. R aggregate function - https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/aggregate
5. https://rstudio.com/products/rstudio/download/
6. https://docs.mongodb.com/manual/tutorial/
7. https://api.mongodb.com/python/current/tutorial.html
8. https://jetstream-cloud.org/

List of Files in supporting files:

1. Python utility file
2. Images of visualization
3. Images of the queried csv files