**HOSPITAL CLAIM**

**MANAGEMENT SYSTEM ANALYSIS**

Created by Batch 75

Cohort Code: INTCDB22DW075



Contents

1. Business Challenge / Requirement ……………………………………...3
2. Goal of the project …………………………………………………………3
3. Project Architecture.………………………………………………………..3
4. Dataset Explanation and Schema ………………………………………..5
5. Problem Statement / Tasks………………………………………………..10
6. Code Templates……………………………………………………………..10

* Data Processing……………………………………………………..10
* Conversion of Raw data into Processed Data…………………..12
* Importing processed data to MYSQL…………………………….14
* Data Transformation ………………………………………………22
* Sqoop and Hive and HDFS…….…………………………………26
* Final processing from data………………………………………..28

1. Output Screen……………………………………………………………...30
2. Business Benefits/ Conclusion…………………………………………...35
3. Further Enhancements/Recommendations …………………………….35
4. References / Bibliography………………….. …………………………….35

* **Business Challenge / Requirement**

This is a Hospital Claim Management System which has many databases stored like Patient information, Claim information, Insurance Company Names, Hospital Details etc. So, to correctly manipulate and analyse this huge sets of data, the authority has requested for our help in Big Data Analytics, for handling and analysing of this datasets.

* **Goal of the Project**

To meet our client requirements, we have to use many databases which store patient information, claim information, insurance company names etc. We need to do data cleansing to fix errors , duplicates and irrelevant data from the raw dataset. We have processed, filtered, transformed and analysed these data. Finally, we will plot a graph of 6months claim of patient and claims based on disease.

* **Project Architecture**

1. A linux file server receives 14 files in total in Comma Seperated Value (CSV) format from the Hospital, Insurance Company and other sources. These are the source files that we will be using for our project.
2. Then we use data filtering in python to filter out NULL and invalid values from our data sources.
3. Now transformational logic is applied on the cleansed files and the data are imported to MYSQL for storage.
4. The file data and tables are validated, enriched, analysed, processed before loading into HDFS and HIVE.
5. Finally, after analysing of the tables and datasets, we are able to visualize the graph we need to get the relation between insurance claims and age group.

**HOSPITAL CLAIM MANAGEMENT**

**SYSTEM ANALYSIS ARCHITECTURE**

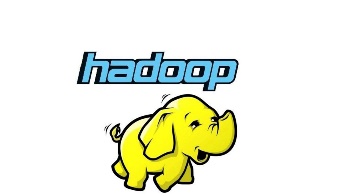
**Data Sources**

**Store, Retrieve and Manage Data**

**Data Preprocessing**

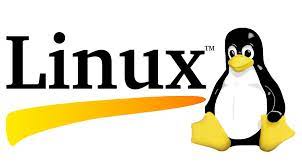
Patient.csv

Claim.csv



Insurer.csv

Data Cleaning



Data Transformation

Hospitals.csv

Group.csv

SubGroup.csv

**Data Ingestion Services**

Matplotlib

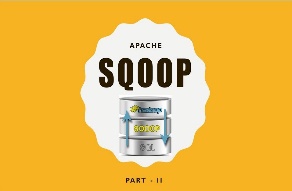
Pandas

Seaborn

**Final Output**

Data Preprocessing

Raw Data



Data Storage





Data Processing

Analysed Data



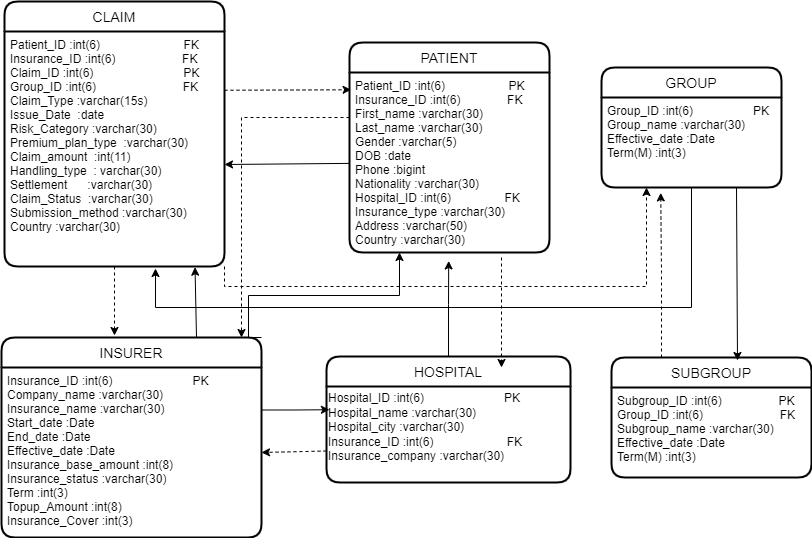
1. Patient claims based on past 6 months

2. Patient claims based on disease

* **Datasets and Schemas**

1. Data coming from third party sources reside in our local directory of Unix and in CSV format.
2. A final master table is created from existing tables which is then inserted in HDFS using SQOOP.
3. Lastly, that table is analysed and visualized using python data analytical package, and matplotlib to map the datasets.

* **DATA MODEL**

****

* **Description of the data source files**
* **Claim CSV file fields**
* Claim\_ID : Unique, not null Primary Key
* Patient\_ID : Unique, not null Primary key for Patient table
* Insurance\_ID : Unique, not null Primary Key for Insurer table
* Group\_ID : Unique, not null Primary Key for Group table
* Claim\_type : Shows the type of claim
* Issue\_date : Issue date for the claim made by patient
* Risk\_Category : Level of risk
* Premium\_plan\_type : Shows the frequency of subscription
* Claim\_Amount : Amount of claim made
* Handling\_type : Type of handling
* Settlement : Status of settlement of claim
* Claim\_status : Status of the claim made
* Submission\_method : Method of claim submission
* Country : Country
* **Patient CSV file fields**
* Patient\_ID : Unique, not null Primary Key
* Insurance\_ID : Unique, not null Primary Key for Insurer table
* Hospital\_ID : Unique, not null Primary Key for Hospital table
* First\_name : First name of the patient
* Last\_name : Last name of the patient
* Gender : Gender
* DOB : Date of birth of patient
* Phone : Contact number of patient
* Nationality : Nationality of patient
* Insurance\_type : Term of the insurance covered
* Address : Full address of patient
* Country : Country in which the patient belong to

* **Insurer CSV file fields**
* Insurance\_ID : Unique, not null Primary Key for Insurer table
* Company\_name : Name of Company providing insurance
* Insurance\_name : Name of Insurance Plan
* Start\_Date : Date of start of insurance plan in dd/mm/yyyy
* End\_Date : End Date of the plan in dd/mm/yyyy
* Effective\_Date : Effective Date of the plan in dd/mm/yyyy
* Insurance\_Base\_Amount : Basic Coverage amount of the plan
* Term : Validity time period of plan in Years
* Topup\_Amount : Amount of topup that can be added
* Insurance\_Cover : Head coverage under the plan
* **Hospital CSV file fields**
* Hospital\_ID : unique, not null Primary key
* Hospital\_Name : Name of the Hospital
* Hospital\_City : City where the hospital is located
* Insurance\_ID : Unique ID of Insurance as foreign key
* Insurance\_Company : Name of Insurance Company
* **Group CSV file fields**
* Group\_ID : Unique, not null Primary key of group table
* Group\_Name : Name of the group of Disease
* Effective\_Date : Effective\_date
* Term : Term of validity in Months
* **SubGroup CSV file fields**
* SubGroup\_ID : Unique, not null primary key
* Group\_ID : Unique , foreign key of table Group
* SubGroup\_Name : Name of the disease under the parent disease
* Effective\_Date : Effective date
* Term : Term of validity in months
* **Description of the final file**
* **Final CSV file fields**
* **Patient\_name : Name of the Patients**
* **Patient\_address : Address of the Patients**
* **Claim\_id : Unique, not null primary key**
* **Insurance\_details : Name of the insurance and insurance company**
* **Claim\_status\_details : Status of the claim made**
* **Claim\_issue\_date : Issue date for the claim made by patient**
* **Insurance\_term : Validity time period of plan in Years**
* **Patient\_age : Age of Patient**
* **Group\_details : Name of the disease and group of disease**
* **Insurance\_amount : Base amount and top up amount**
* **Problem Statement**

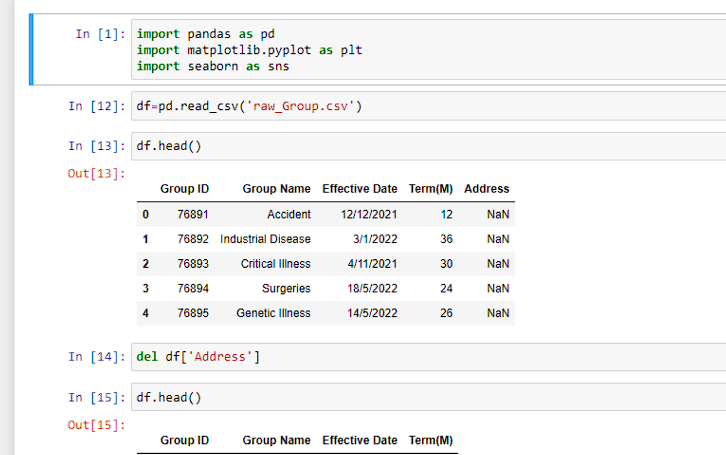
From this huge dataset, we process and analyze the data carefully and find out the following pattern :

1. Insurance claims made by patient who are admitted in the Hospital from the past 6 months by the patient’s age.
2. Claims of patient based on disease.
3. Claim uses and balance amount based on Insurance company.

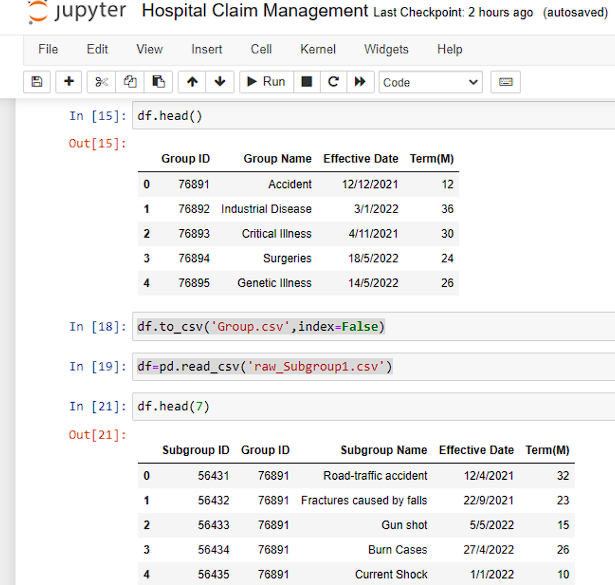
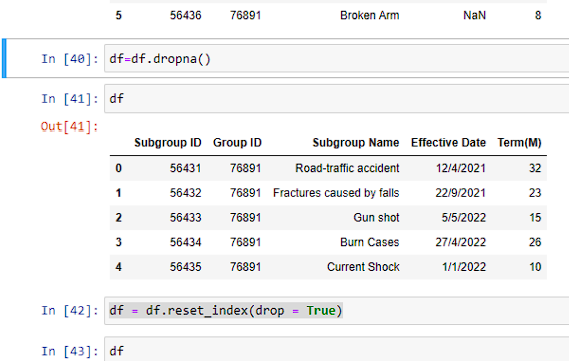
* **Code Templates**
* **Data Processing :**

First we fetch all the raw source files and then cleanse it using python pandas, such as removing duplicate values, null values, then it gets converted to processed data. Here is some code snippets.

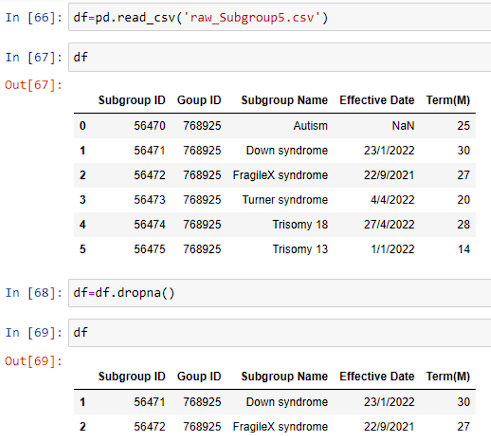
Cleansing of group.csv



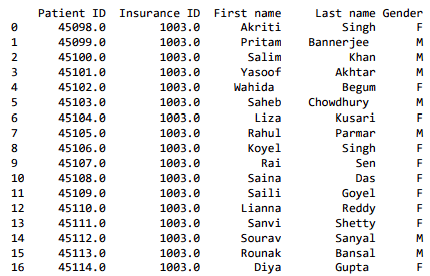
Cleansing of subgroup files by removing duplicates.

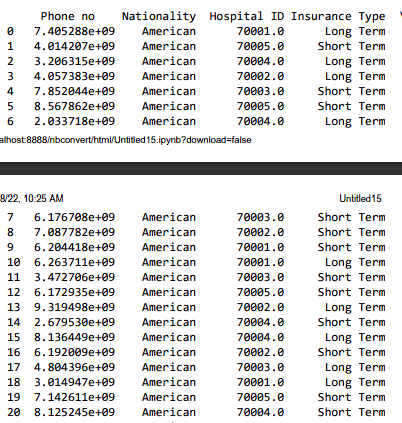
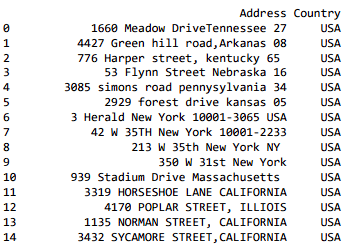


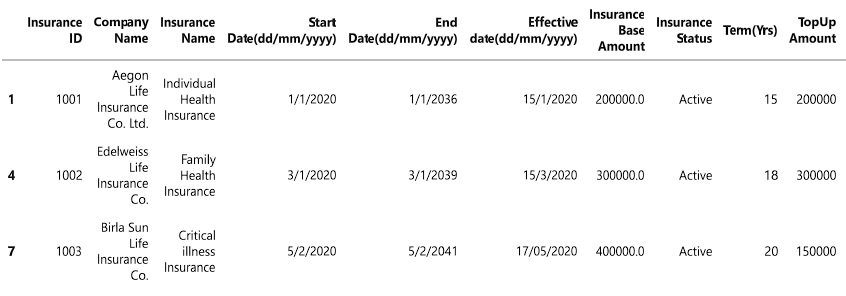
Cleansing of subgroup files by removing null values.



* **Processed data**

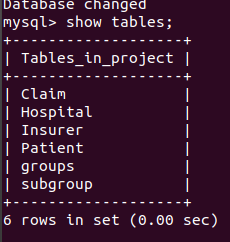
****



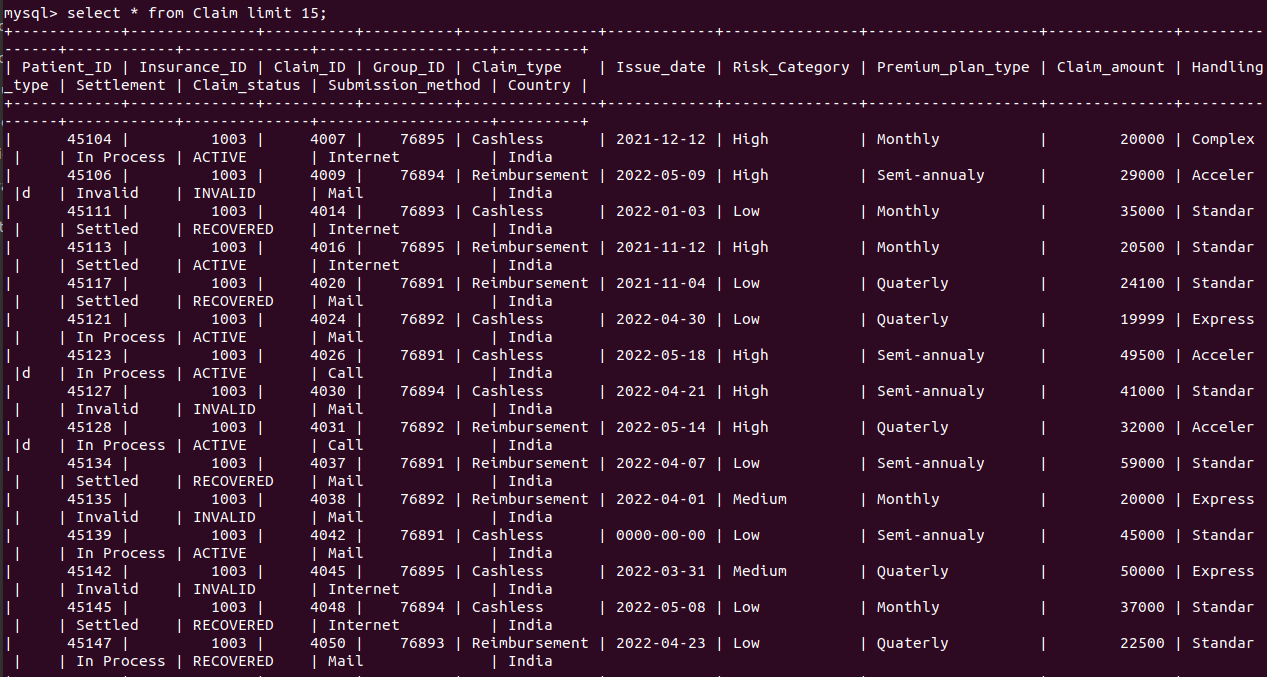
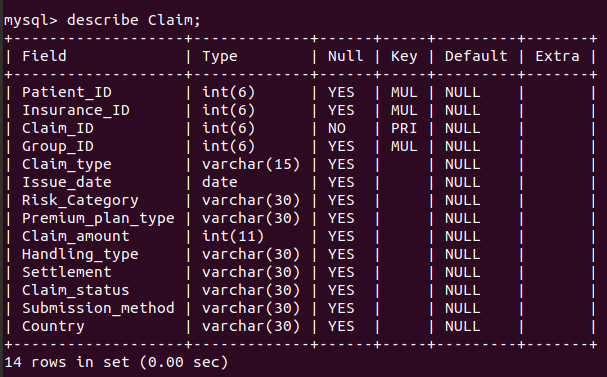
Insurer.csv after processing and cleansing

* **Importing Processed data to MYSQL**

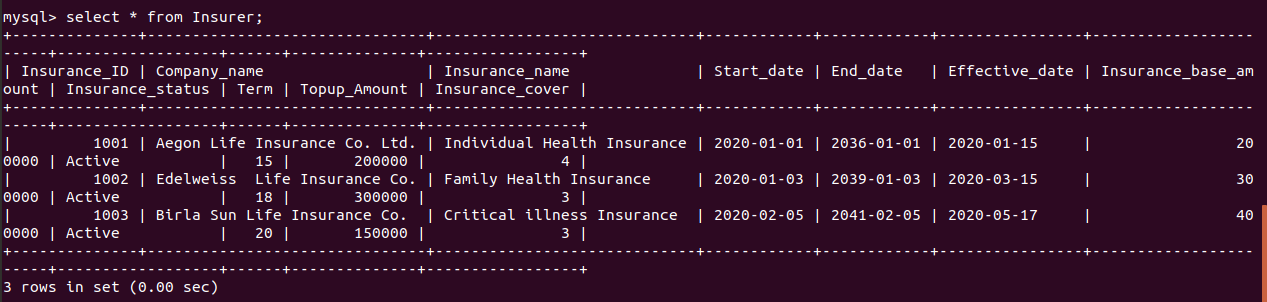
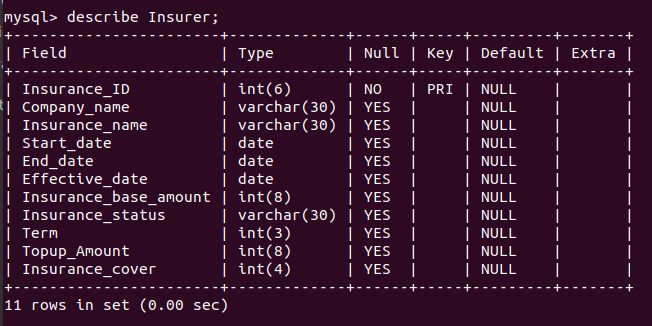
List of all tables, that are imported to MySQL from the processed files.



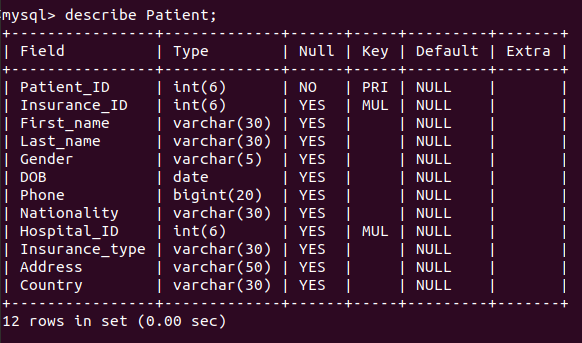
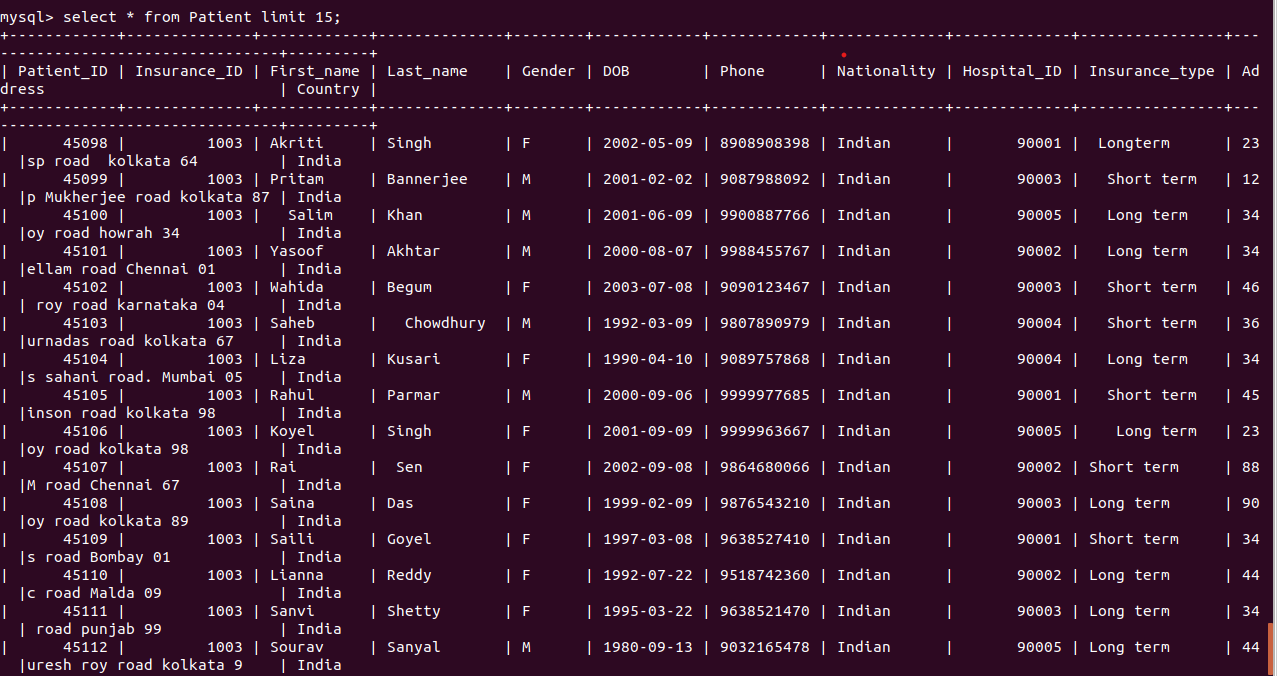
* **CLAIM TABLE**

****

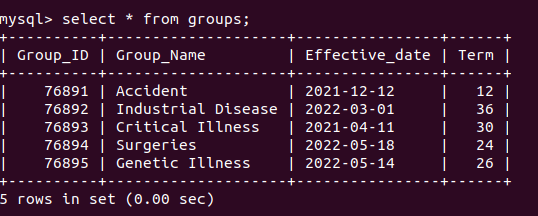
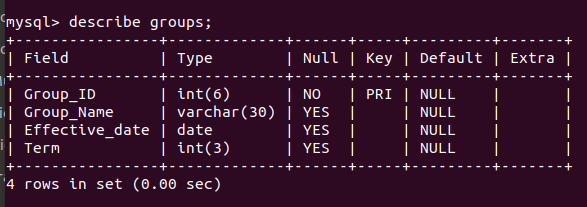
* **INSURER TABLE**

****

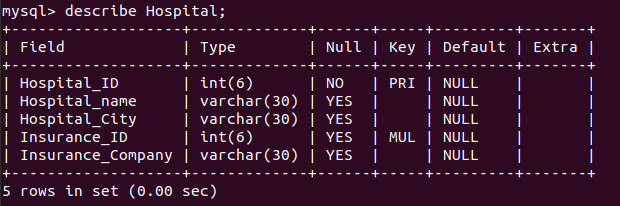
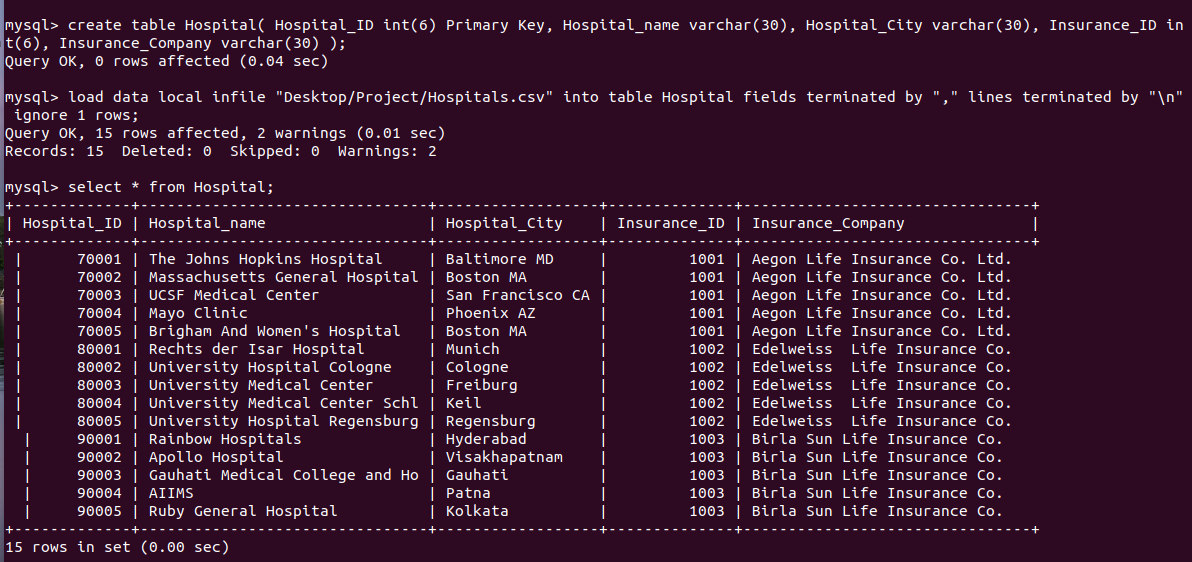
* **PATIENT TABLE**

****

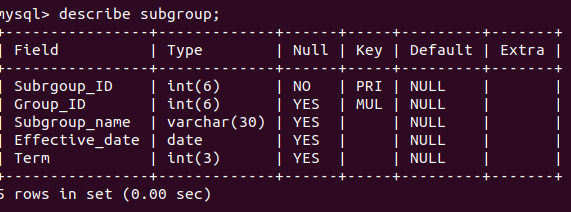
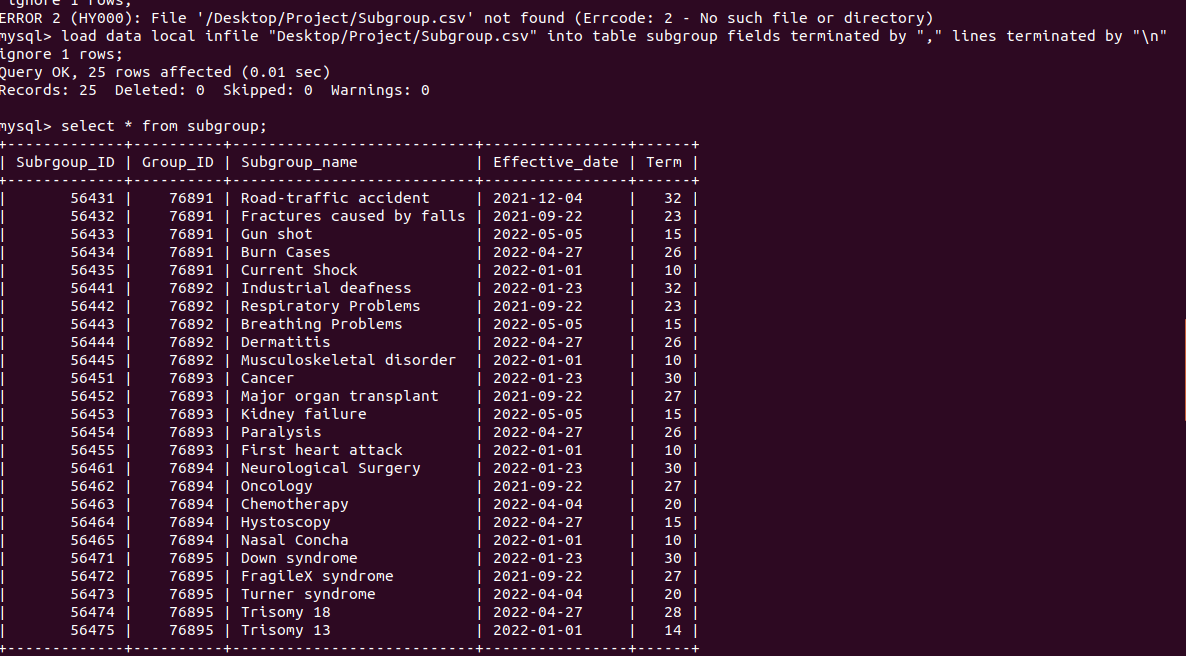
* **GROUPS TABLE**

****

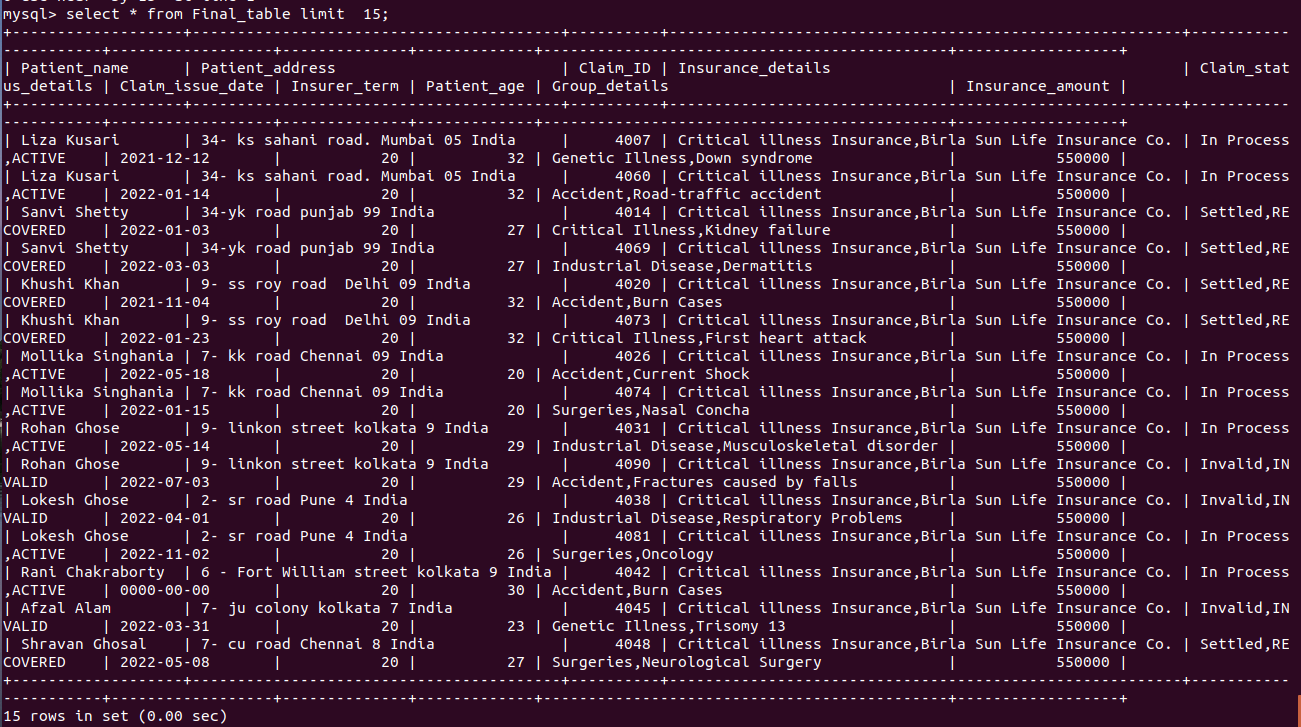
* **HOSPITAL TABLE**

****

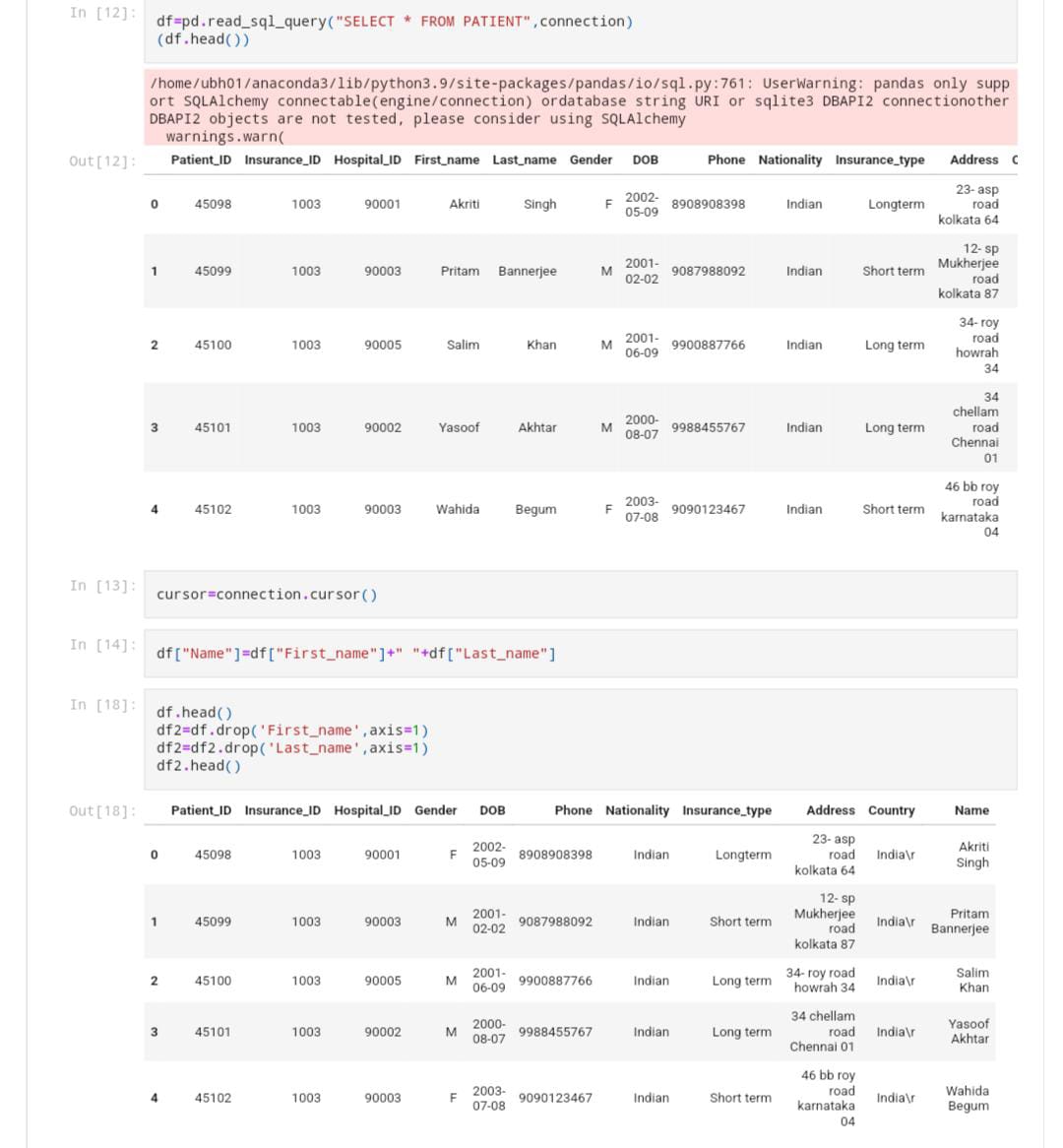
* **SUBGROUP TABLE**

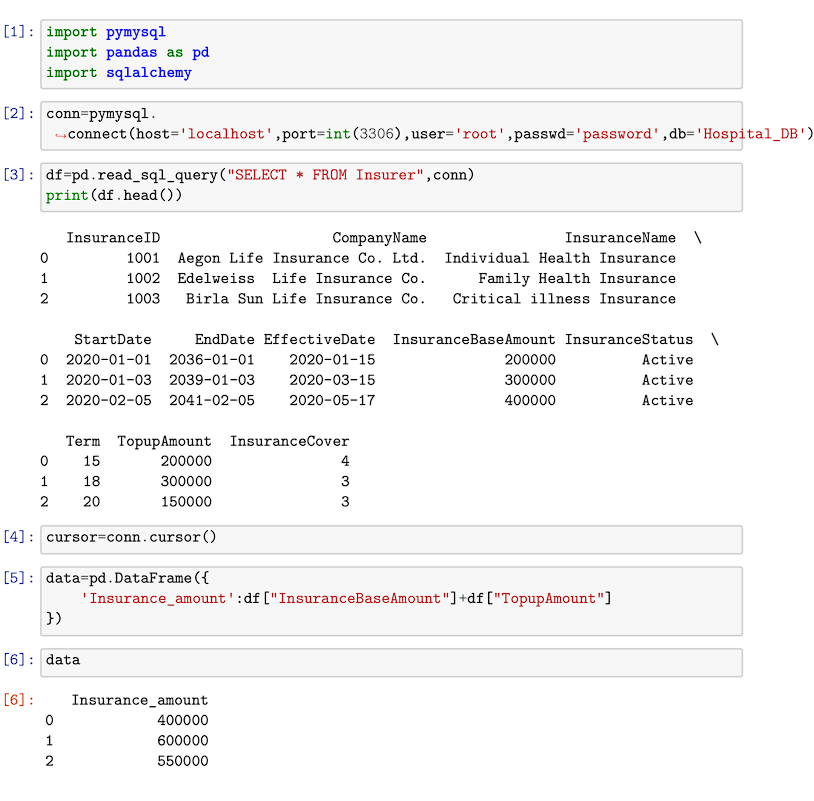


Final Table in MYSQL

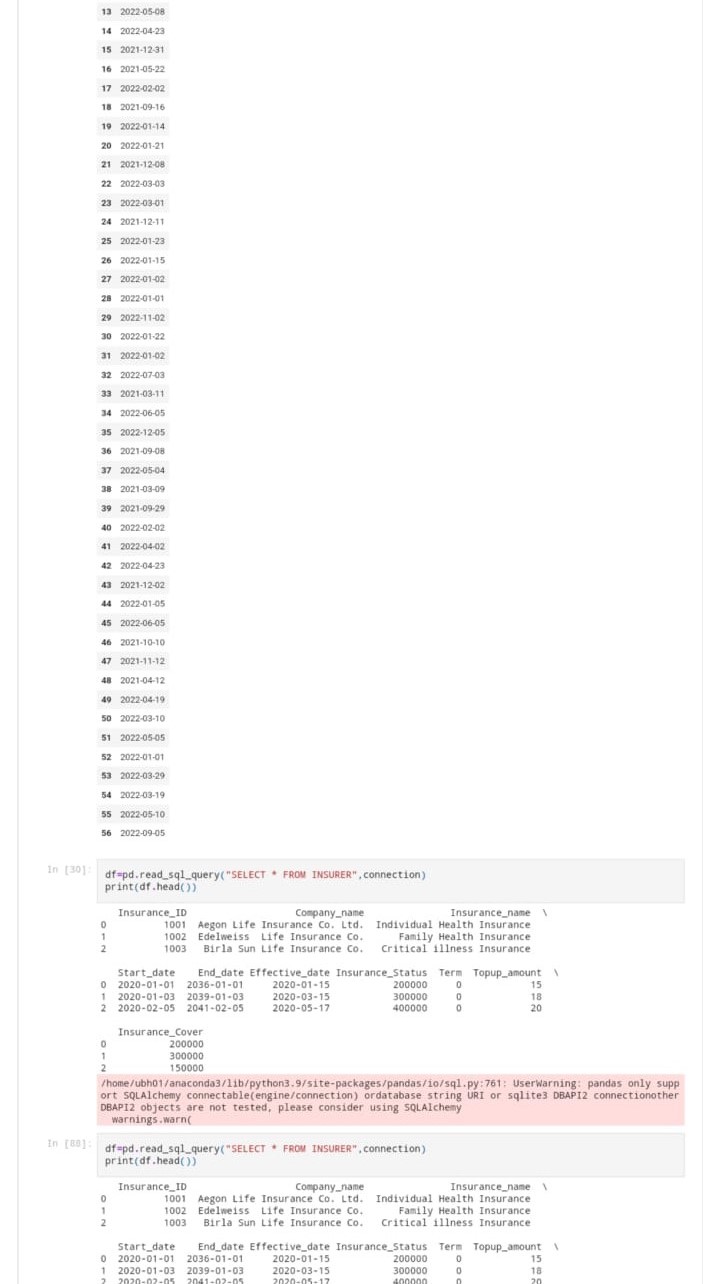


* **DATA TRANSFORMATION**

****



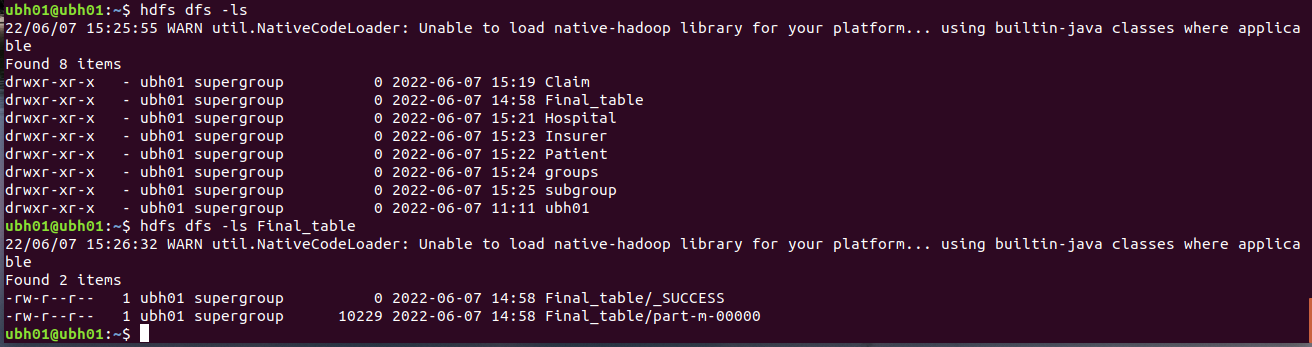




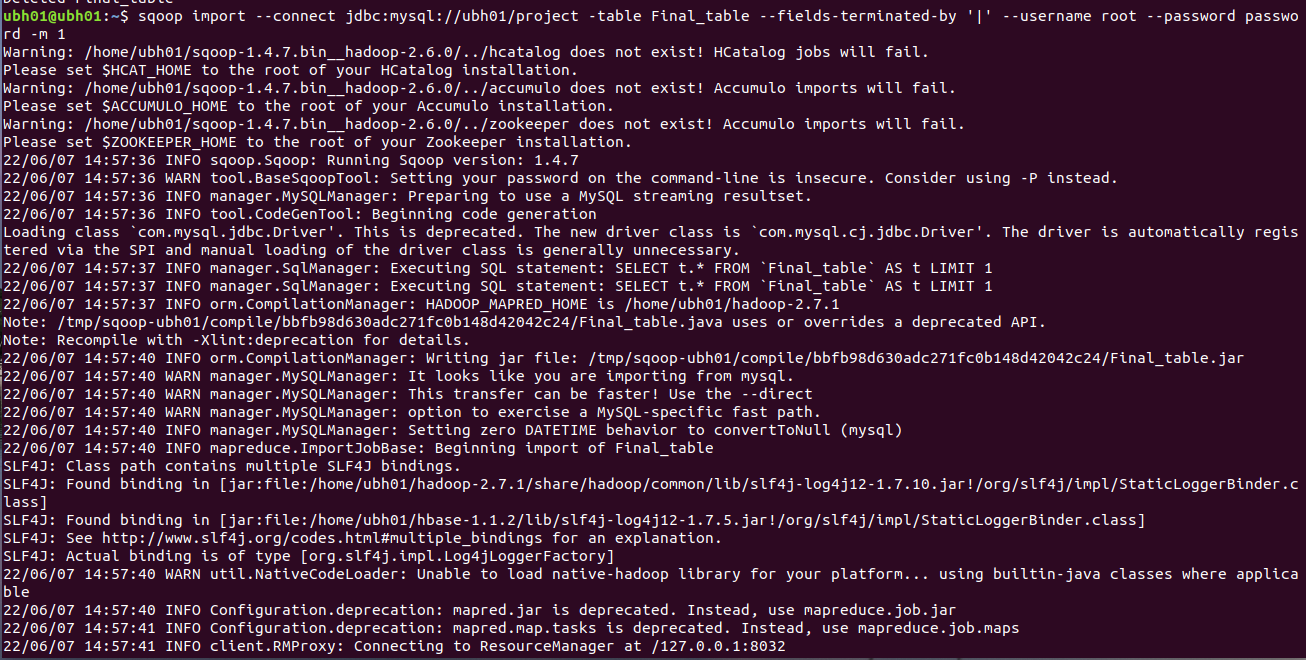
s

* **SQOOP and HIVE and HDFS**

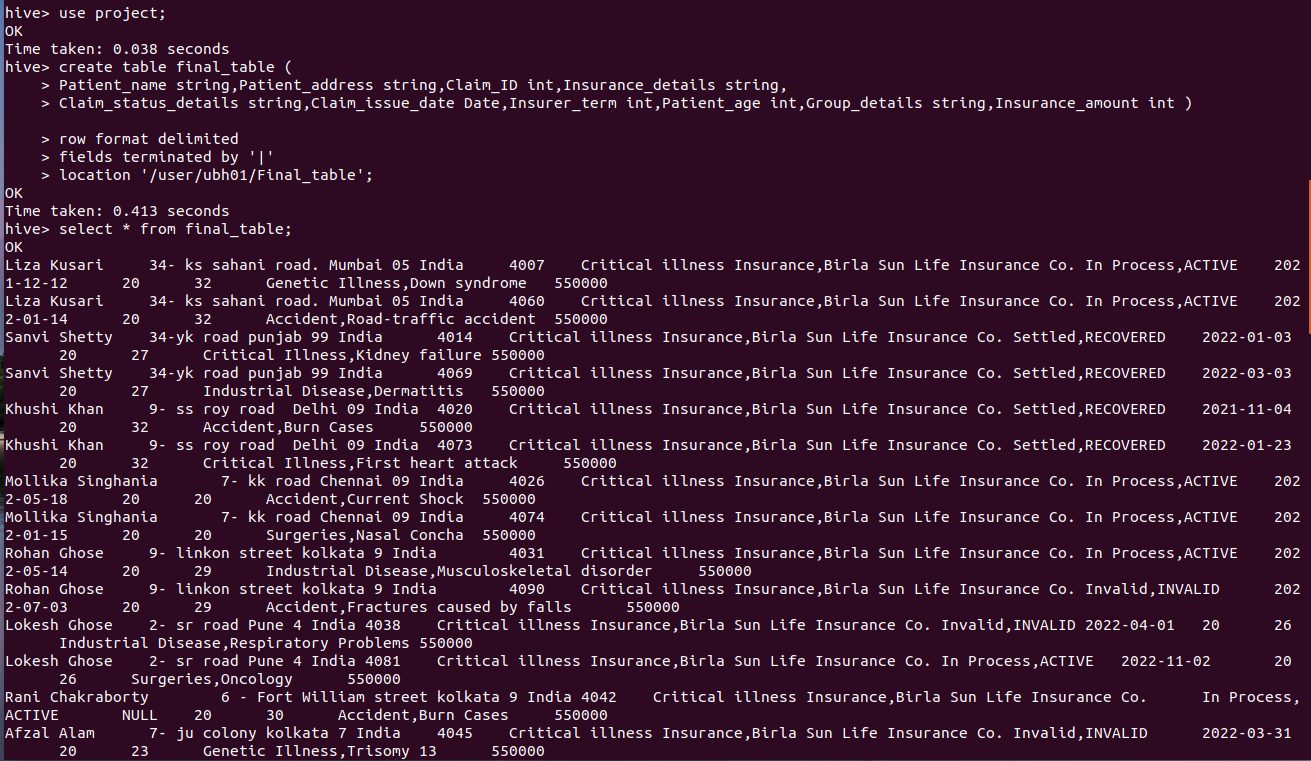
All tables are imported to HDFS from MYSQL



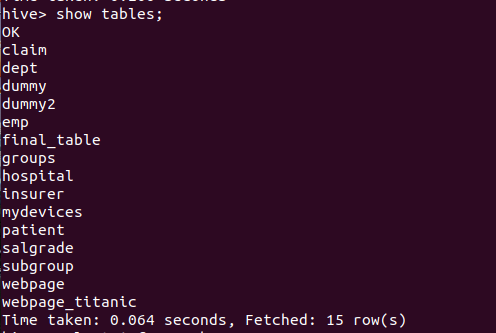
We have used Sqoop for the import from HDFS to HIVE



Like this, we have created the schema for every table in HIVE



We have imported the tables : claim, final\_table, groups, hospital, insurer, patient and subgroup to HIVE.

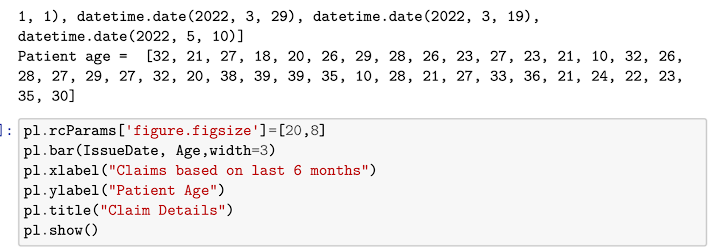


* **Final processing from data**

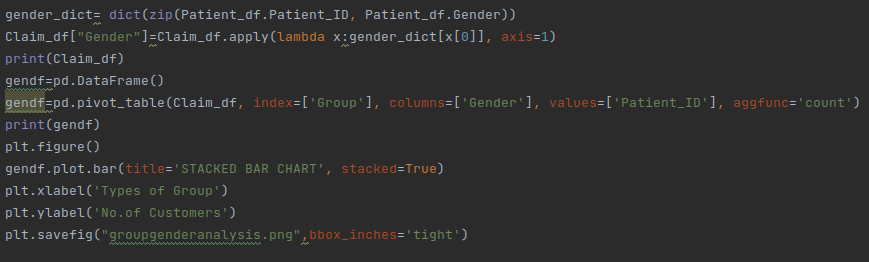
Analyzing data from Final table to plot the final Graphs**.**

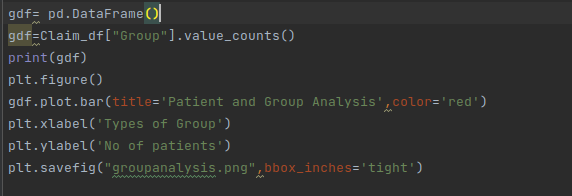
* **Claims based on last 6 months.**

****

****

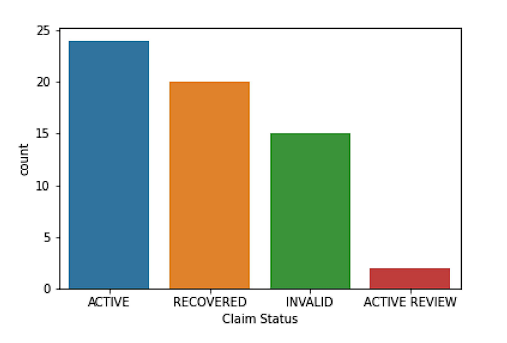
* **Patient claims based on disease.**

****

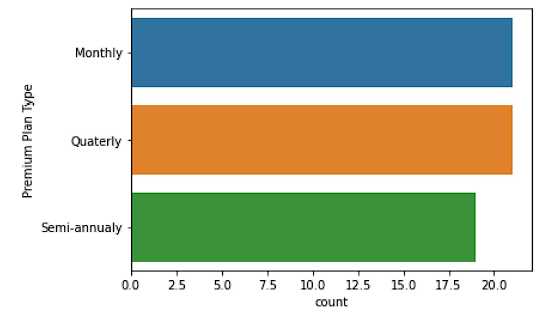
****

* **OUTPUT SCREEN**

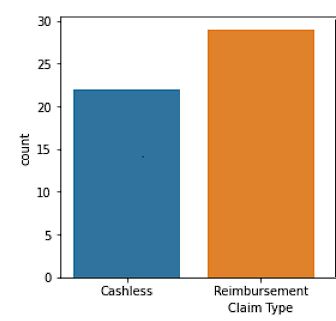
Different claims made and the claim status

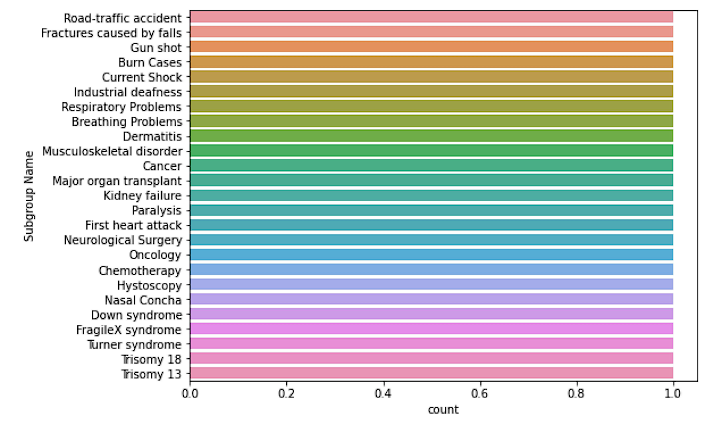
****

Insurance Premium Plan Type

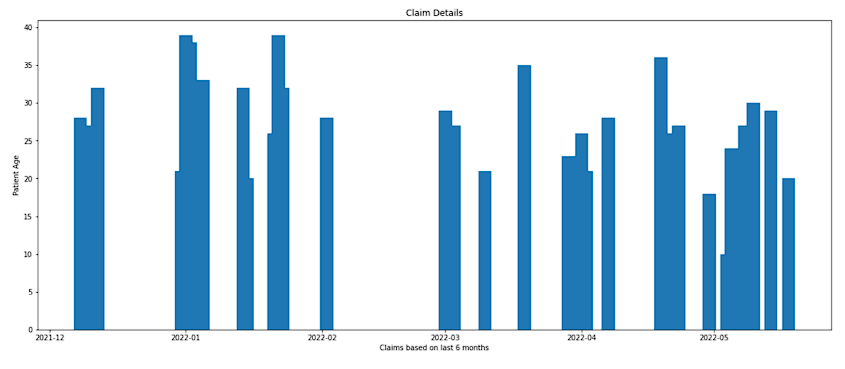


Type of claim request Cashless or Reimbursement

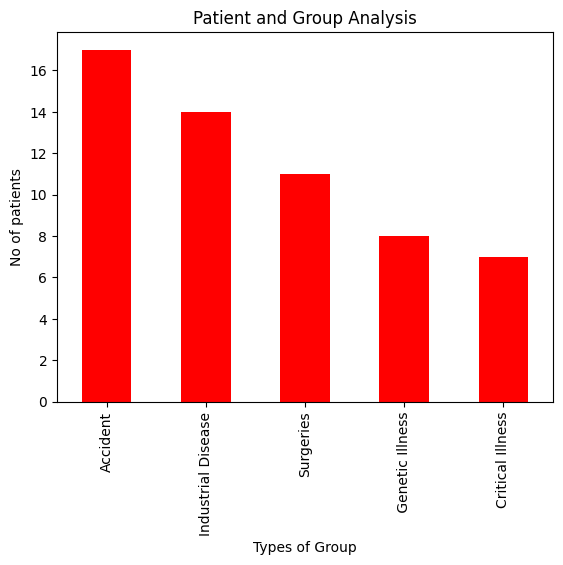


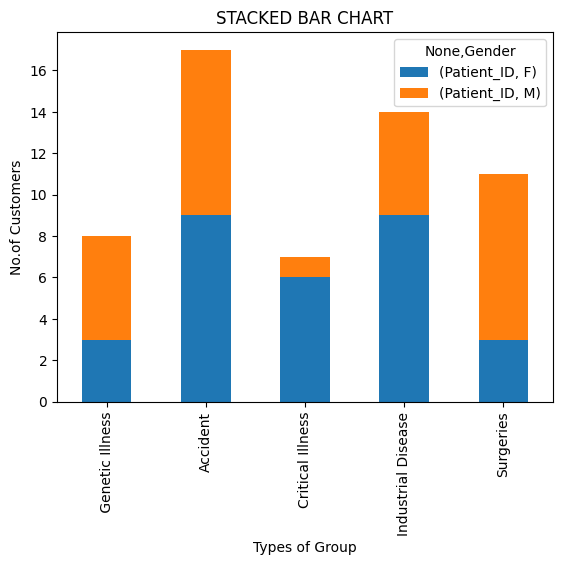


Subgroups of Disease

 Patient claims based on past 6 months

Patient claims based on disease



 Patient claims based on disease, classified by Gender

* **Business Benefits/Conclusion**

The company is facing challenges to get the insights they need to improve their services as the dataset is on a much bigger scale and may contain missing records. So the company wants to take the help of Data Analytics using Big Data Ecosystem in order to evaluate the quality of care provided by health care providers. To do this, we have been assigned to identify patterns of past claims made for insurance in order to understand the customers better, and provide targeted experience and improve customer retention.

* **Further Enhancements/Recommendations**

This project has a vast scope in future for all the Insurance Companies. Companies can get detailed analysis of data from this project and can serve the customers according to their special needs and benefits. They can also change their insurance plans and business needs for maximum profits and also provide with the optimal service to patients.

* **References/Bibliography**

1. Python Data analytics
2. DrawIO
3. GeeksforGeeks
4. Hive
5. Jupyter
6. Hadoop

* **GitHub Link**

[BigData-Project-Hospital-Claim-Management](https://github.com/sourin-das/BigData-Project-Hospital-Claim-Management/)