Solution

I will be doing this project in Extract,Transform and Load manner popularly known as ETL.

The first process will be Extract (E) the data where

- All the given csv and json files will be uploaded into AWS S3 bucket.

The second process will Transform(T) the data where

- Connection will be created in between Amazon Web Service Simple Storage Service(AWS S3)bucket and databricks using access keys and secret keys for authentication.
- Spark dataframe Application Programming Interface(API)s will be reading each csv and json files, creating dataframes from S3 to databricks with the help of spark session.
- Those dataframes will be used for data manipulation, data cleansing, transformation on databricks.
- Transformation like joining the tables, removing duplicates, replacing the null values for specific columns by NA, filtering will be done.

Some of the transformation use cases are explained below.

Use Case -1 Which disease has a maximum number of claims.

Data Cleaning(DropDuplicates,Replace)

- Duplicate rows in table claims and disease will be dropped.
- Null values will be replaced with NA.

Joining tables(join)

- Disease and claims table will be joined based on "disease_name" column which will associate each claim with its corresponding disease.

Grouping and Aggregation (GroupBy,Count,agg)

- Data will be grouped by the "Disease_name" column
- Number of claims will be counted for each disease.
- Data will be aggregated to calculate the total number of claims for each disease.

Sorting(sort)

- Aggregated data will be sorted in descending order based on the count of claims which will bring disease with the maximum number of claims to the top.

Selecting Top Result(select)

- Top row will select the sorted data to identify the disease with the maximum number of claims.

Use Case -2 List all the patients below the age of 18 who are admitted for cancer

- Joining Tables:
  - Patient_records table and disease table will be joined based on the disease_name column to associate each patient's disease with their records.
- Filtering Patients with Cancer:

    - Joined Dataframe will be filtered to include only records where the disease is cancer.
- Calculating Age:
    - Age of each patient will be calculated based on their patient_birth_date and the current date.
- Filtering Patients Below 18:
    - DataFrame will be filtered on the basis of patients below the age of 18.

The third process will Load(L) the data where

- Connection will be established in between AWS S3 and Redshift by specifying the Identity and Access Management(IAM) role and Amazon Resource Name(ARN).
- Final cleaned data after extraction and transformation will be published on redshift tables through Structured Query Language(SQL) credential.
- Separate tables will be created for the output of individual use cases.

Technologies and Platforms to be used in this solution

- Spark/Pyspark
- Draw.io for uml
- AWS (S3,Redshift)
- Databricks Community Edition
- Jira
- Github