



SDM - Final Project

Name: Disease Data Analysis and Visualization

Team:

- Prarthana Bahuriya
- Jetendra Mulinti
- Goutham Vemula
- Prajeeth Nakka



Agenda

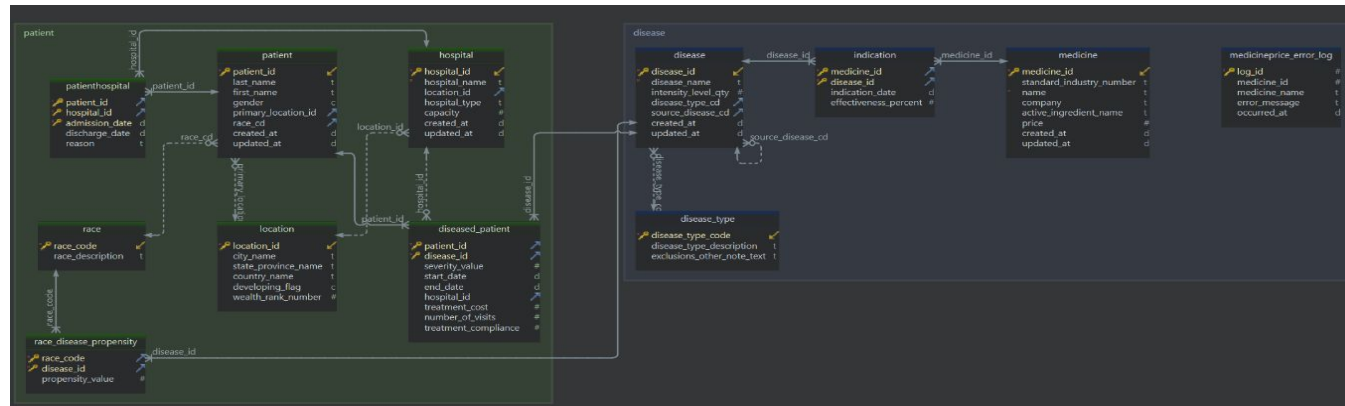
- Introduction to the Project
- Business Problem Overview
- ER Diagram and Data Dictionary
- Operational Database
- Dimensional Database and ETL Process
- Dashboard and Business Analytics
- NO SQL vs. Relational Database Structure
- AWS Architecture and Implementation
- Snowflake in Our Project
- Conclusion



Business Problem :

- **Problem Statement:** Identifying patterns and insights in disease data to understand healthcare business.
- **Project Objective:** Utilize disease data to create a robust data model, perform analytical queries, and provide actionable insights through visualizations.

ER Diagram and Data Dictionary



- [Data Dictionary](#)

Operational Database

Create Statements for Base tables

```
1 DROP TABLE IF EXISTS PatientHospital;
2 DROP TABLE IF EXISTS Diseased_Patient;
3 DROP TABLE IF EXISTS Indication;
4 DROP TABLE IF EXISTS Race_Disease_Propensity;
5 DROP TABLE IF EXISTS Medicine;
6 DROP TABLE IF EXISTS Patient;
7 DROP TABLE IF EXISTS Disease;
8 DROP TABLE IF EXISTS Hospital;
9 DROP TABLE IF EXISTS Disease_Type;
10 DROP TABLE IF EXISTS Race;
11 DROP TABLE IF EXISTS Location;
12
13
14 -- Creating Location Table
15 CREATE TABLE Location (
16     Location_ID SERIAL PRIMARY KEY,
17     City_Name VARCHAR(100),
18     State_Province_Name VARCHAR(100),
19     Country_Name VARCHAR(100),
20     Developing_Flag CHAR(1),
21     Wealth_Rank_Number INT
22 );
23
24 -- Creating Disease Type Table
25 CREATE TABLE Disease_Type (
26     Disease_Type_Code CHAR(5) PRIMARY KEY,
27     Disease_Type_Description VARCHAR(255),
28     Exclusions_Other_Note_Text VARCHAR(255)
29 );
30
31 -- Creating Race Table
32 CREATE TABLE Race (
```

Insert Statements for Base tables

```
---- Location
INSERT INTO Location (Location_ID, City_Name, State_Province_Name, Country_Name, Developing_Flag, Wealth_Rank_Number) VALUES
(1, 'New York', 'New York', 'USA', 'N', 1),
(2, 'Los Angeles', 'California', 'USA', 'N', 2),
(3, 'Chicago', 'Illinois', 'USA', 'N', 3),
(4, 'Houston', 'Texas', 'USA', 'N', 4),
(5, 'Phoenix', 'Arizona', 'USA', 'N', 5),
(6, 'Philadelphia', 'Pennsylvania', 'USA', 'N', 6),
(7, 'San Antonio', 'Texas', 'USA', 'N', 7),
(8, 'San Diego', 'California', 'USA', 'N', 8),
(9, 'Dallas', 'Texas', 'USA', 'N', 9),
(10, 'San Jose', 'California', 'USA', 'N', 10),
(11, 'Austin', 'Texas', 'USA', 'N', 11),
(12, 'Jacksonville', 'Florida', 'USA', 'N', 12),
(13, 'Fort Worth', 'Texas', 'USA', 'N', 13),
(14, 'Columbus', 'Ohio', 'USA', 'N', 14),
(15, 'Charlotte', 'North Carolina', 'USA', 'N', 15),
(16, 'San Francisco', 'California', 'USA', 'N', 16),
(17, 'Indianapolis', 'Indiana', 'USA', 'N', 17),
(18, 'Seattle', 'Washington', 'USA', 'N', 18),
(19, 'Denver', 'Colorado', 'USA', 'N', 19),
(20, 'Washington', 'D.C.', 'USA', 'N', 20);

select * from Location;

--- Disease_Type
INSERT INTO Disease_Type (Disease_Type_Code, Disease_Type_Description, Exclusions_Other_Note_Text) VALUES
('DTC01', 'Infectious', 'Excludes chronic infections'),
('DTC02', 'Metabolic', 'Excludes inherited metabolic disorders');
```

Checks, Triggers & Views

Referential Integrity Check

```
150 ----- Patient
151
152 INSERT INTO Patient(Patient_id, last_name, first_name, gender, primary_location_id, race_cd, created_at, updated_at) VALUES
153 ('1', 'Chevez', 'Ryan', 'M', 8, 'WHI', '2022-09-12', '2023-01-08'),
154 ('2', 'Banks', 'Jason', 'M', 7, 'BLA', '2021-07-05', '2023-01-04'),
155 ('3', 'Ramirez', 'Devon', 'M', 15, 'OTH', '2022-07-01', '2023-01-11'),
156 ('4', 'Pacheco', 'Kathleen', 'W', 5, 'HIS', '2022-11-18', '2023-01-07'),
157 ('5', 'Jacobs', 'Heather', 'W', 9, 'ASI', '2021-03-02', '2023-01-06'),
158 ('6', 'Olson', 'Gina', 'M', 8, 'OTH', '2022-02-22', '2023-01-02'),
159 ('7', 'Sanders', 'Brandon', 'M', 3, 'BLA', '2021-06-03', '2023-01-01'),
160 ('8', 'Alexander', 'Karen', 'F', 10, 'HIS', '2021-01-17', '2023-01-02'),
161 ('9', 'Cole', 'Ronald', 'F', 16, 'ASI', '2021-01-27', '2023-01-11'),
162 ('10', 'Hughes', 'Earl', 'M', 13, 'HIS', '2022-07-17', '2023-01-11'),
163 ('11', 'Anderson', 'Dana', 'F', 5, 'BLA', '2021-04-09', '2023-01-07'),
164 ('12', 'Weaver', 'Amanda', 'M', 3, 'WHI', '2022-09-14', '2023-01-01'),
165
166 Data Output Messages Notifications
167
168 ERROR: Key (race_cd)=(WHI) is not present in table "race". Insert or update on table "patient" violates foreign key constraint "patient_race_cd_fkey"
169
170 SQL state: 23503
171
172 Detail: Key (race_cd)=(WHI) is not present in table "race".
```

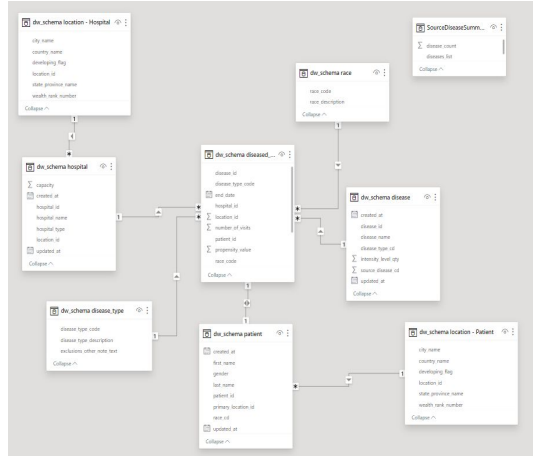
Gender Trigger Check

```
26 -----
27
28 INSERT INTO Patient(Patient_id, last_name, first_name, gender, primary_location_id, race_cd, created_at, updated_at) VALUES
29 ('100', 'Chevez', 'Ryan', 'Q', 8, 'WHI', '2022-09-12', '2023-01-08');
30
31
32
33 Data Output Messages Notifications
34
35 ERROR: Invalid gender value. Allowed values are 'M', 'F', 'Other'.
36
37 CONTEXT: PL/pgSQL function validate_patient_gender() line 4 at RAISE
38
39 SQL state: P0001
```

PriceTrigger Check

```
68
69
70 INSERT INTO Medicine
71 (Medicine_ID, Standard_Industry_Number, Name, Company, Price, Active_Ingredient_Name, Updated_At) VALUES
72 ('100', 'ME074187', 'Prednis', 'Med', -1, 'Prednisone', '2023-08-21');
73
74
75 Data Output Messages Notifications
76
77 ERROR: Price cannot be negative
78
79 CONTEXT: PL/pgSQL function check_medicine_price() line 9 at RAISE
```

Dimensional Database and ETL Process



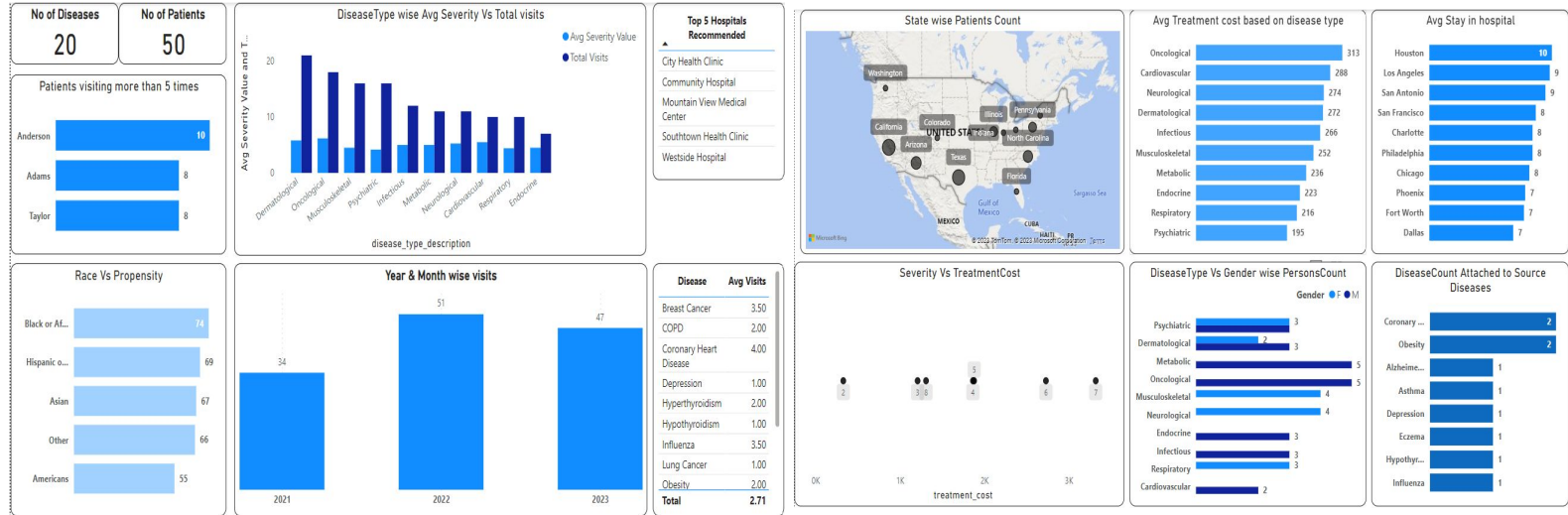
Create Statements for OLAP

```
1 --- CREATE SCHEMA IF NOT EXISTS dw_schema;
2
3 DROP TABLE IF EXISTS dw_schema.location;
4 DROP TABLE IF EXISTS dw_schema.race;
5 DROP TABLE IF EXISTS dw_schema.disease_type;
6 DROP TABLE IF EXISTS dw_schema.hospital;
7 DROP TABLE IF EXISTS dw_schema.patient;
8 DROP TABLE IF EXISTS dw_schema.diseased_patient;
9
10
11
12 -- DW Table: Location
13 CREATE TABLE dw_schema.location (
14     location_id INT PRIMARY KEY,
15     city_name VARCHAR(100),
16     state_province_name VARCHAR(100),
17     country_name VARCHAR(100),
18     developing_flag CHAR(1),
19     wealth_rank_number INT
20 );
21
22 -- DW Table: Race
23 CREATE TABLE dw_schema.race (
24     race_code CHAR(3) PRIMARY KEY,
25     race_description VARCHAR(255)
26 );
27
28 -- DW Table: Disease Type
29 CREATE TABLE dw_schema.disease_type (
30     disease_type_code CHAR(5) PRIMARY KEY,
31     disease_type_description VARCHAR(255),
32     exclusions_other_note_text VARCHAR(255)
```

Insert Statements for OLAP

```
1 -- Insert data into Location
2 INSERT INTO dw_schema.location (location_id,city_name, state_province_name, country_name, developing_flag, wealth_rank_number)
3 SELECT location_id,city_name, state_province_name, country_name, developing_flag, wealth_rank_number
4 FROM public.location;
5
6 -- Insert data into Race
7 INSERT INTO dw_schema.race (race_code, race_description)
8 SELECT race_code, race_description
9 FROM public.race;
10
11
12 -- Insert data into Disease Type
13 INSERT INTO dw_schema.disease_type (disease_type_code, disease_type_description, exclusions_other_note_text)
14 SELECT disease_type_code, disease_type_description, exclusions_other_note_text
15 FROM public.disease_type;
16
17 -- Insert data into Hospital
18 INSERT INTO dw_schema.hospital (hospital_id,hospital_name, location_id, hospital_type, capacity, created_at, updated_at)
19 SELECT hospital_id,hospital_name, location_id, hospital_type, capacity, created_at, updated_at
20 FROM public.hospital;
21
22
23 -- Insert data into Patient
24 INSERT INTO dw_schema.patient (Patient_id,last_name, first_name, gender, primary_location_id, race_cd, created_at, updated_at)
25 SELECT Patient_id,last_name, first_name, gender, primary_location_id, race_cd, created_at, updated_at
26 FROM public.Patient;
27
28
29 -- Insert data into Disease
30 INSERT INTO dw_schema.disease (disease_id, disease_name, intensity_level_qty, disease_type_cd, source_disease_cd, created_at,
31 SELECT disease_id, disease_name, intensity_level_qty, disease_type_cd, source_disease_cd, created_at,
32 FROM public.disease;
```

Dashboard and Business Analytics



State wise Patients Count

Avg Treatment cost based on disease type

Oncological	313
Cardiovascular	288
Neurological	274
Dermatological	272
Infectious	266
Musculoskeletal	252
Metabolic	236
Endocrine	223
Respiratory	216
Psychiatric	195

Avg Stay in hospital

Houston	10
Los Angeles	9
San Antonio	9
San Francisco	8
Charlotte	8
Philadelphia	8
Chicago	8
Phoenix	7
Fort Worth	7
Dallas	7

Race Vs Propensity

Black or Af...	74
Hispanic o...	69
Asian	67
Other	66
Americans	55

Year & Month wise visits

Year	Visits
2021	34
2022	51
2023	47

Disease Avg Visits

Disease	Avg Visits
Breast Cancer	3.50
COPD	2.00
Coronary Heart Disease	4.00
Depression	1.00
Hypertthyroidism	2.00
Hypothyroidism	1.00
Influenza	3.50
Lung Cancer	1.00
Obesity	2.00
Total	2.71

Severity Vs TreatmentCost

DiseaseType Vs Gender wise PersonsCount

DiseaseType	F	M
Psychiatric	3	0
Dermatological	2	3
Metabolic	0	5
Oncological	0	5
Musculoskeletal	4	0
Neurological	4	0
Endocrine	3	0
Infectious	3	0
Respiratory	3	0
Cardiovascular	2	0

DiseaseCount Attached to Source Diseases

Coronary ...	2
Obesity	2
Alzheim...	1
Asthma	1
Depression	1
Eczema	1
Hypothy...	1
Influenza	1

NO SQL vs. Relational Database Structure

NoSQL Database:

- Schema Flexibility
- Horizontal Scaling
- Data Model Options
- Denormalization

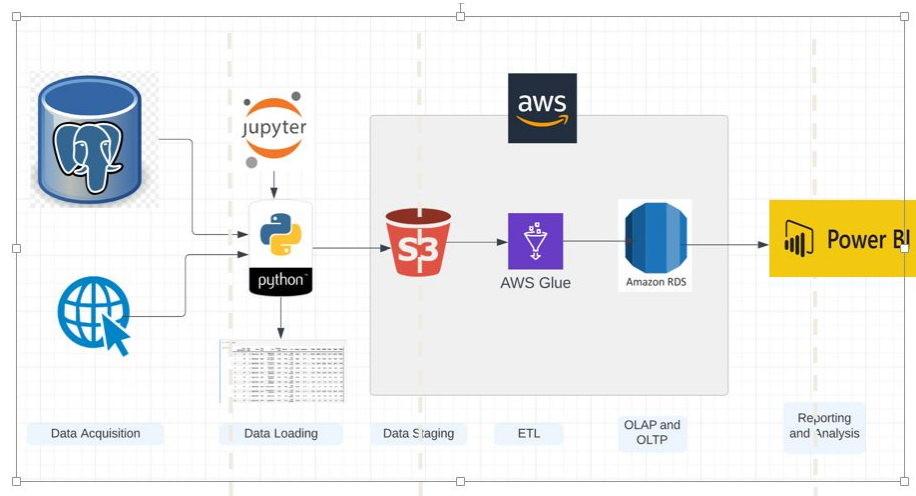
Relational Database:

- Structured Schema
- ACID Transactions
- Complex Query Support
- Normalization

Person_ID	Last_Name	First_Name	Gender	Primary_Location_ID	Race_CD
1	Doe	Jane	F	123	WHT
2	Smith	John	M	456	BLK

```
{
  "Person_ID": 1,
  "Last_Name": "Doe",
  "First_Name": "Jane",
  "Gender": "F",
  "Primary_Location": {
    "Location_ID": 123,
    "City": "Springfield",
    "State": "IL"
  },
  "Race": "White"
},
{
  "Person_ID": 2,
  "Last_Name": "Smith",
  "First_Name": "John",
  "Gender": "M",
  "Primary_Location": {
    "Location_ID": 456,
    "City": "Chicago",
    "State": "IL"
  },
  "Race": "Black"
}
```

AWS Workflow





Snowflake Commentary

- **Scalability:** Effortlessly scales to meet analytical demands with on-the-fly compute resource adjustment.
- **Performance:** Delivers rapid data processing for real-time insights, vital for healthcare decision-making.
- **Time Travel:** Allows users to access historical data, providing the ability to query past states of the database for auditing, recovery, or analytical purposes.
- **Concurrent Access:** Multiple users can analyze data simultaneously without performance lag.
- **Data Sharing:** Facilitates seamless and secure sharing with stakeholders, promoting collaborative research.
- **Security:** Offers robust encryption and compliance features, ensuring patient data is protected.
- **Ecosystem Integration:** Connects with a vast array of tools and platforms for streamlined workflows.



Thank you.....