ABC DENTAL CLINIC DATABASE DESIGN

Introduction

The dental clinic named ABC Dental Clinic has a world class type of service in dentistry. Accompanied with a small group of employees. However, ABC had a problem for accommodating their clients like appointing Schedule, handling patients once they arrive at clinic, payment method and lastly the most important every business had been the revenue of their clinic. So, they need a data engineer professional to make and design their database as one improvement of the business systematically.

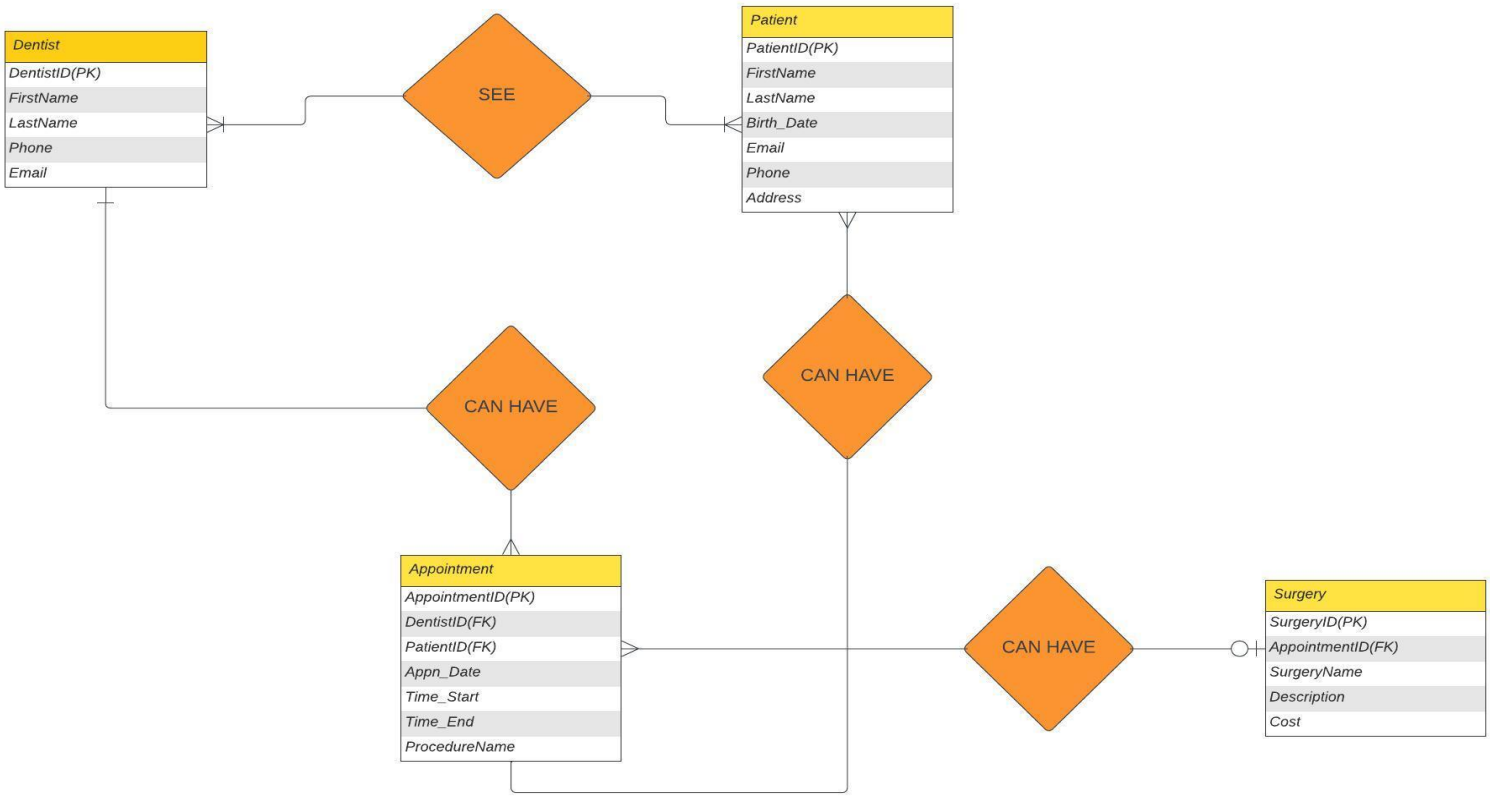
Data engineer was curious on how the dental business do by their workflow and also cashflow on maintaining the business has been active so he makes 5 types of business question that may answer by making ERD and explaining the fact tables derive on it. These are the business question wrote below:

1. How many appointments were scheduled for each dentist in the past month?
2. What is the average waiting time for patients before they are seen by a dentist?
3. Which dental procedures are the most commonly performed?
4. What is the revenue generated by the clinic in the last quarter?
5. How many patients visit the clinic each day, and what is the most common reason for their visit?

The said dental Clinic have a process dealing with their patient treatment and smooth transaction. The ABC Dental Clinic system controls clinic revenue, dentist schedules, patient appointments, and dental operations. Dental operations are performed on patients who have scheduled visits with dentists, and the clinic makes money from these services. Let’s make it simple.

ABC DENTAL CLINIC DATABASE ERD DIAGRAM

Delson M. Rantael , Project Sparta Student



Working_Link:https://lucid.app/lucidchart/e4666da8-44a2-4511-8aad-55b92bc771ac/edit?beaconFlowId=C30B4B9AD60E8F15&invitationId=inv_d752fa5c-f12f-45a8-9dca-77b11e6fffd3&page=0_0#

Normalization Process /Normalized Model

We're creating a normalized database model based on the Entity Relationship Diagram (ERD) of the ABC Dental Clinic system. Normalization is the process of organizing data to minimize redundancy and ensure data integrity. Here's what each part of this process involves:

- **Normalized Tables:** We define individual tables for the different entities in the system, such as Patients, Dentists, Appointments, and Surgery. Each table represents a single type of data and is designed to store related information in a structured manner.
- **Primary Keys:** Each table has a primary key, a unique identifier for each record in the table. For example, PatientID is the primary key for the Patients table, DentistID is the primary key for the Dentists table, and so on.
- **Foreign Keys:** Foreign keys are used to establish relationships between tables. For instance, in the Appointments table, PatientID and DentistID are foreign keys that link to the Patients and Dentists tables, respectively.

Now we can demonstrate this process by making a coding script in SQL. In addition, we eventually insert some dummy data on each entity as required on our Capstone. First, we must create a database for the normalized model, and next the entities, inserting attributes later on my last page.

1.Create first Database for normalize table:

```
CREATE DATABASE ABCDentalClinicNormalized;
```

2. On Created database, click new query and run the following code to create tables

Create the normalized tables (Patients,Dentists,Appointments,Surgery) according to ERD we designed

Create the table for Patients

```sql
CREATE TABLE Patients(
    PatientID INT NOT NULL,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Birth_Date DATE,
    Email VARCHAR(50),
    Phone VARCHAR(50),
    Address VARCHAR(200),
    PRIMARY KEY(PatientID)
);

--Create the table for Dentists
CREATE TABLE Dentists(
    DentistID INT NOT NULL,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Email VARCHAR(50),
    Phone VARCHAR(50),
    PRIMARY KEY(DentistID)
);

--Create the table for Appointments
CREATE TABLE Appointments(
    AppointmentID INT NOT NULL,
    PatientID INT NOT NULL,
    DentistID INT NOT NULL,
    Appn_Date DATE,
    Time_start TIME,
    Time_end TIME,
    Procedure VARCHAR(200),
    PRIMARY KEY(AppointmentID),
    FOREIGN KEY(PatientID) REFERENCES Patients(PatientID),
    FOREIGN KEY(DentistID) REFERENCES Dentists(DentistID)
);

--Create the table for Surgery
CREATE TABLE Surgery(
    SurgeryID INT NOT NULL,
    AppointmentID INT NOT NULL,
    SurgeryName VARCHAR(200),
    Description VARCHAR(200),
    Cost DECIMAL(10,2),
    PRIMARY KEY(SurgeryID),
    FOREIGN KEY(AppointmentID) REFERENCES Appointments(AppointmentID)
);
```

Dimensional Normal Norm Methodology

We take the normalized model and transform it into a dimensional model. Dimensional modeling is a technique used in data warehousing for efficient querying and reporting. Here's what each part of this process involves:

Dimension Tables: Dimension tables contain descriptive attributes that provide context for the measures (facts) in a data warehouse. We create dimension tables for Time, Patients, Dentists, and Procedures. These tables will be used to slice and dice data for reporting.

Fact Tables: Fact tables contain quantitative data, such as measures or facts. In this case, we have two fact tables: Appointment Fact (which captures data related to appointments) and Revenue Fact (which stores information about revenue).

- Surrogate Keys: Dimension tables have surrogate keys (e.g., PatientKey, DentistKey) that provide unique identifiers for each dimension member. These surrogate keys are used in the fact tables to establish relationships.

Foreign Keys: Foreign keys in fact tables link to the surrogate keys in dimension tables, creating relationships between facts and dimensions. For example, in the AppointmentFact table, DateKey links to the TimeDimension.

By following these steps, you'll have structured and organized databases for both your normalized and dimensional models, ready for data storage and analysis in your Dental Clinic system. The dimensional model is particularly well-suited for reporting and business intelligence purposes.

Dimensional Model (Dimension Table and Fact Tables)

```
-- Next we create database for "ABCDentalClinicDimensional"
CREATE DATABASE ABCDentalClinicDimensional;
--Create the dimensional tables (Patients,Dentists,Appointments,Surgery)
according to erd we designed
--Create the table for Time Dimension
CREATE TABLE TimeDimension (
       DateKey INT PRIMARY KEY,
       Date DATE,
       DayOfWeek VARCHAR(10),
       Month VARCHAR(10),
       Quarter INT,
       Year INT
   );
```

```sql
--Create the dimensional tables (Patients,Dentists,Appointments,Surgery)
according to erd we designed
--Create the table for Time Dimension
CREATE TABLE TimeDimension (
       DateKey INT PRIMARY KEY,
       Date DATE,
       DayOfWeek VARCHAR(10),
       Month VARCHAR(10),
       Quarter INT,
       Year INT
   );
--Create the table for Patients Dimension
CREATE TABLE PatientsDimension (
       PatientKey INT  NOT NULL,
       PatientID INT,
       FirstName VARCHAR(50) NOT NULL,
       LastName VARCHAR(50) NOT NULL,
       Birth_Date DATE,
       Email VARCHAR(50),
       Phone VARCHAR(50),
       Address VARCHAR(200),
       PRIMARY KEY(PatientKey)
);
--Create the table for Dentist Dimension
CREATE TABLE Dentist_Dimension(
    DentistKey INT NOT NULL,
    DentistID INT NOT NULL,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Email VARCHAR(50),
    Phone VARCHAR(50),
    PRIMARY KEY(DentistKey)
);

--Create the table for Surgery Dimension
CREATE TABLE Surgery_Dimension(
    SurgeryKey INT NOT NULL,
    SurgeryID INT NOT NULL,
    SurgeryName VARCHAR(200),
    Description VARCHAR(200),
    Cost DECIMAL(10,2),
    PRIMARY KEY(SurgeryKey)
);
```

Fact Tables

```
--Create the table for Appointment Fact
CREATE TABLE AppointmentFact (
    AppointmentKey INT NOT NULL,
    DateKey INT NOT NULL,
    PatientKey INT NOT NULL,
    DentistKey INT NOT NULL,
    SurgeryKey INT NOT NULL,
    Time_start TIME,
    Time_end TIME,
    PRIMARY KEY(AppointmentKey),
    FOREIGN KEY (DateKey) REFERENCES TimeDimension(DateKey),
    FOREIGN KEY (PatientKey) REFERENCES PatientsDimension(PatientKey),
    FOREIGN KEY (DentistKey) REFERENCES Dentist_Dimension(DentistKey),
    FOREIGN KEY (SurgeryKey) REFERENCES Surgery_Dimension(SurgeryKey)
);

--Create the table for Revenue Fact
CREATE TABLE RevenueFact (
    DateKey INT NOT NULL,
    TotalRevenue DECIMAL(10,2),
    PRIMARY KEY(DateKey),
    FOREIGN KEY (DateKey) REFERENCES TimeDimension(DateKey)
);
```

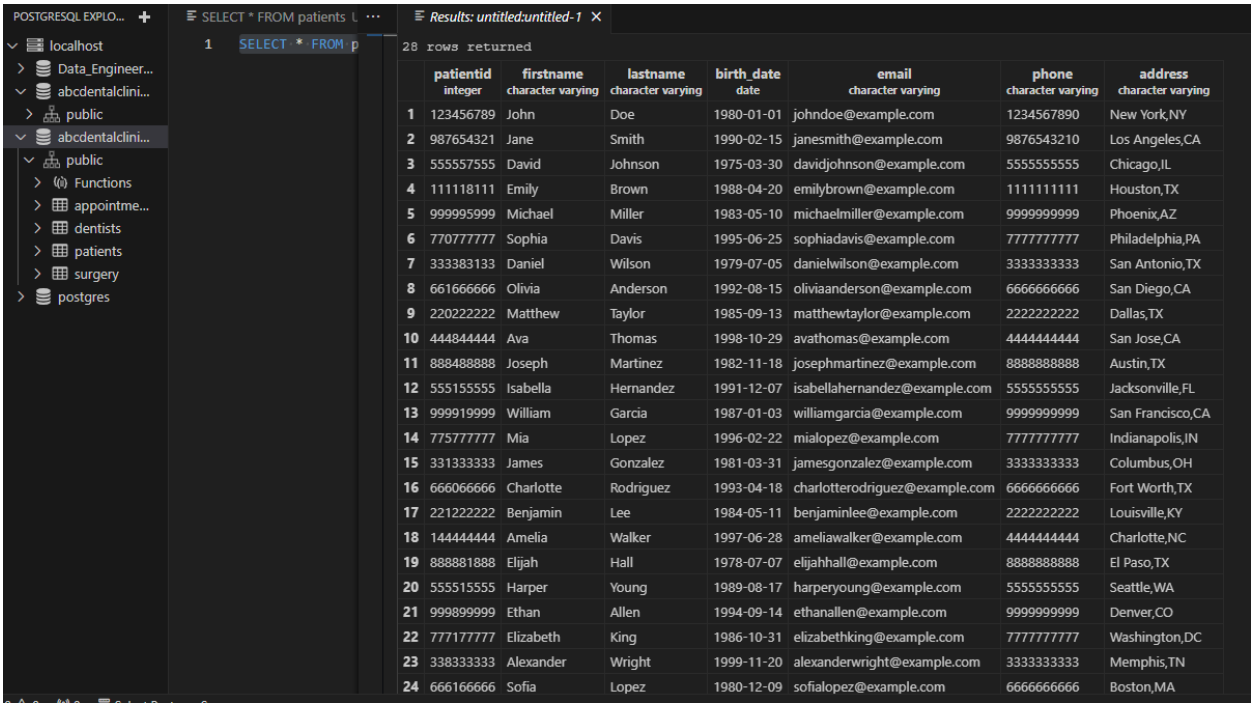
Creating and Inserting Data

On this section we need to make a dummy data using AI namely “jsondataai” website to easily scraping it and convert to csv format .Then we can now insert it on every table on normalized database by using this script ;

```
COPY patients FROM 'C:\Users\Admin\Downloads\SPARTA\SP703_Scripts\patient.csv'
DELIMITER ',' CSV HEADER;
COPY dentists FROM 'C:\Users\Admin\Downloads\SPARTA\SP703_Scripts\Dentists.csv'
DELIMITER ',' CSV HEADER;
COPY appointments FROM
'C:\Users\Admin\Downloads\SPARTA\SP703_Scripts\Appointements.csv' DELIMITER ','
CSV HEADER;
COPY surgery FROM  'C:\Users\Admin\Downloads\SPARTA\SP703_Scripts\Surgery.csv'
DELIMITER ',' CSV HEADER;
```

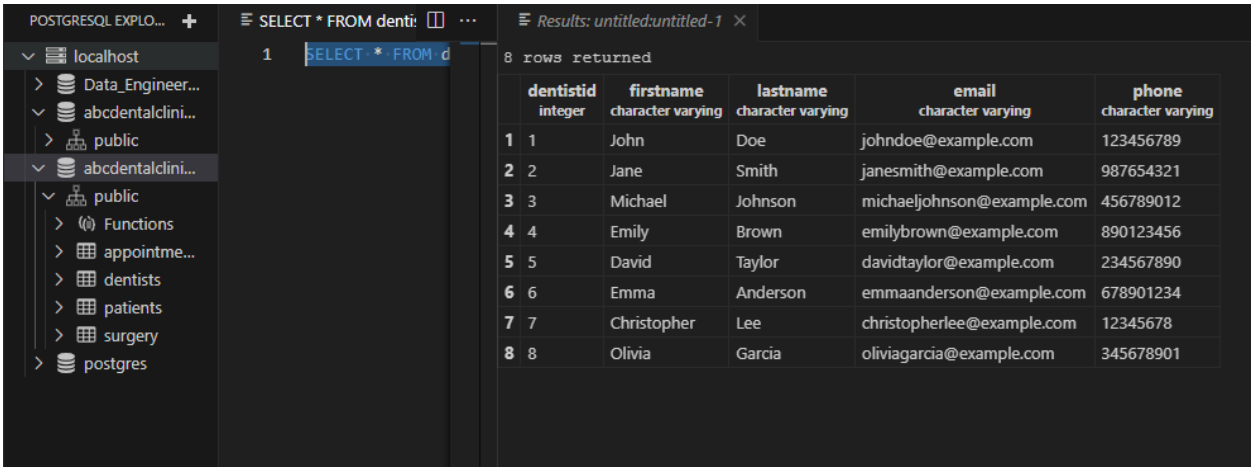
Output Shows;

Patient table



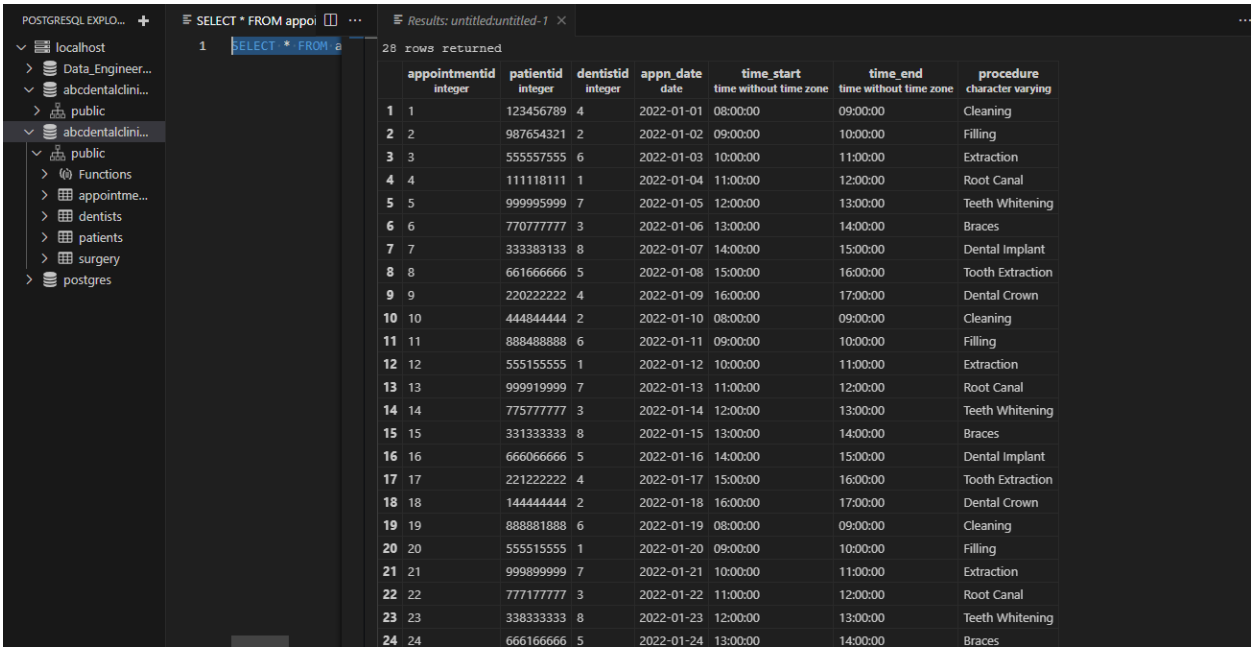
	patientid integer	firstname character varying	lastname character varying	birth_date date	email character varying	phone character varying	address character varying
1	123456789	John	Doe	1980-01-01	johndoe@example.com	1234567890	New York,NY
2	987654321	Jane	Smith	1990-02-15	janesmith@example.com	9876543210	Los Angeles,CA
3	555557555	David	Johnson	1975-03-30	davidjohnson@example.com	5555555555	Chicago,IL
4	111118111	Emily	Brown	1988-04-20	emilybrown@example.com	1111111111	Houston,TX
5	999995999	Michael	Miller	1983-05-10	michaelmiller@example.com	9999999999	Phoenix,AZ
6	770777777	Sophia	Davis	1995-06-25	sophiadavis@example.com	7777777777	Philadelphia,PA
7	333383133	Daniel	Wilson	1979-07-05	danielwilson@example.com	3333333333	San Antonio,TX
8	661666666	Olivia	Anderson	1992-08-15	oliviaanderson@example.com	6666666666	San Diego,CA
9	220222222	Matthew	Taylor	1985-09-13	matthewtaylor@example.com	2222222222	Dallas,TX
10	444844444	Ava	Thomas	1998-10-29	avathomas@example.com	4444444444	San Jose,CA
11	888488888	Joseph	Martinez	1982-11-18	josephmartinez@example.com	8888888888	Austin,TX
12	555155555	Isabella	Hernandez	1991-12-07	isabellahernandez@example.com	5555555555	Jacksonville,FL
13	999919999	William	Garcia	1987-01-03	williamgarcia@example.com	9999999999	San Francisco,CA
14	775777777	Mia	Lopez	1996-02-22	mialopez@example.com	7777777777	Indianapolis,IN
15	331333333	James	Gonzalez	1981-03-31	jamesgonzalez@example.com	3333333333	Columbus,OH
16	666066666	Charlotte	Rodriguez	1993-04-18	charlotterodriguez@example.com	6666666666	Fort Worth,TX
17	221222222	Benjamin	Lee	1984-05-11	benjaminlee@example.com	2222222222	Louisville,KY
18	144444444	Amelia	Walker	1997-06-28	ameliawalker@example.com	4444444444	Charlotte,NC
19	888881888	Elijah	Hall	1978-07-07	elijahhall@example.com	8888888888	El Paso,TX
20	555515555	Harper	Young	1989-08-17	harperyoung@example.com	5555555555	Seattle,WA
21	999899999	Ethan	Allen	1994-09-14	ethanallen@example.com	9999999999	Denver,CO
22	777177777	Elizabeth	King	1986-10-31	elizabethking@example.com	7777777777	Washington,DC
23	338333333	Alexander	Wright	1999-11-20	alexanderwright@example.com	3333333333	Memphis,TN
24	666166666	Sofia	Lopez	1980-12-09	sofialopez@example.com	6666666666	Boston,MA

Dentist table



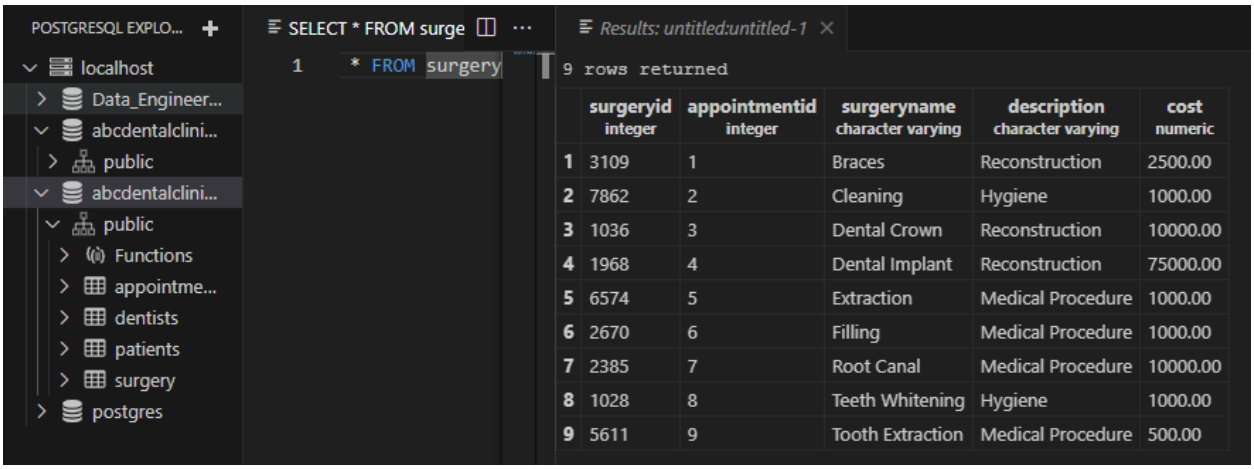
	dentistid integer	firstname character varying	lastname character varying	email character varying	phone character varying
1	1	John	Doe	johndoe@example.com	123456789
2	2	Jane	Smith	janesmith@example.com	987654321
3	3	Michael	Johnson	michaeljohnson@example.com	456789012
4	4	Emily	Brown	emilybrown@example.com	890123456
5	5	David	Taylor	davidtaylor@example.com	234567890
6	6	Emma	Anderson	emmaanderson@example.com	678901234
7	7	Christopher	Lee	christopherlee@example.com	12345678
8	8	Olivia	Garcia	oliviagarcia@example.com	345678901

Appointments Table



	appointmentid integer	patientid integer	dentistid integer	appn_date date	time_start time without time zone	time_end time without time zone	procedure character varying
1	1	123456789	4	2022-01-01	08:00:00	09:00:00	Cleaning
2	2	987654321	2	2022-01-02	09:00:00	10:00:00	Filling
3	3	555557555	6	2022-01-03	10:00:00	11:00:00	Extraction
4	4	111118111	1	2022-01-04	11:00:00	12:00:00	Root Canal
5	5	999995999	7	2022-01-05	12:00:00	13:00:00	Teeth Whitening
6	6	770777777	3	2022-01-06	13:00:00	14:00:00	Braces
7	7	333383133	8	2022-01-07	14:00:00	15:00:00	Dental Implant
8	8	661666666	5	2022-01-08	15:00:00	16:00:00	Tooth Extraction
9	9	220222222	4	2022-01-09	16:00:00	17:00:00	Dental Crown
10	10	444844444	2	2022-01-10	08:00:00	09:00:00	Cleaning
11	11	888488888	6	2022-01-11	09:00:00	10:00:00	Filling
12	12	555155555	1	2022-01-12	10:00:00	11:00:00	Extraction
13	13	999919999	7	2022-01-13	11:00:00	12:00:00	Root Canal
14	14	775777777	3	2022-01-14	12:00:00	13:00:00	Teeth Whitening
15	15	331333333	8	2022-01-15	13:00:00	14:00:00	Braces
16	16	666066666	5	2022-01-16	14:00:00	15:00:00	Dental Implant
17	17	221222222	4	2022-01-17	15:00:00	16:00:00	Tooth Extraction
18	18	144444444	2	2022-01-18	16:00:00	17:00:00	Dental Crown
19	19	888881888	6	2022-01-19	08:00:00	09:00:00	Cleaning
20	20	555515555	1	2022-01-20	09:00:00	10:00:00	Filling
21	21	999899999	7	2022-01-21	10:00:00	11:00:00	Extraction
22	22	777177777	3	2022-01-22	11:00:00	12:00:00	Root Canal
23	23	338333333	8	2022-01-23	12:00:00	13:00:00	Teeth Whitening
24	24	666166666	5	2022-01-24	13:00:00	14:00:00	Braces

Surgery Table:



	surgeryid integer	appointmentid integer	surgeryname character varying	description character varying	cost numeric
1	3109	1	Braces	Reconstruction	2500.00
2	7862	2	Cleaning	Hygiene	1000.00
3	1036	3	Dental Crown	Reconstruction	10000.00
4	1968	4	Dental Implant	Reconstruction	75000.00
5	6574	5	Extraction	Medical Procedure	1000.00
6	2670	6	Filling	Medical Procedure	1000.00
7	2385	7	Root Canal	Medical Procedure	10000.00
8	1028	8	Teeth Whitening	Hygiene	1000.00
9	5611	9	Tooth Extraction	Medical Procedure	500.00

Now we can able to insert this table to a dimension table while also mapping the original primary key each table to a new surrogate key each table in the dimensional database. However transferring and populating data in connecting to database into data warehouse are super dangly extensive . I will explain it thoroughly ,first we need to connect two database using MySQL or sometime python .After that all attributes and entities are acceptable to the destination table match on it. We can use this step below to populate some data in dimension database:

TimeDimension Table:

Populate the TimeDimension table with relevant time-related data. Depending on your reporting needs, you may insert dates, days of the week, months, quarters, and years into this table.

PatientDimension Table:

Insert data from the Patients table into the PatientDimension table. You will need to map the PatientID to a unique surrogate key (PatientKey) in this table.

DentistDimension Table:

Insert data from the Dentists table into the DentistDimension table, similarly mapping DentistID to DentistKey.

SurgeryDimension Table:

Populate the SurgeryDimension table with data from the Surgery table, again mapping SurgeryID to SurgeryKey.

AppointmentFact Table:

To populate the AppointmentFact table, you would insert data from the Appointments table. However, you'll need to map PatientID, DentistID, and SurgeryID to their corresponding surrogate keys in the respective dimension tables.

RevenueFact Table:

Insert data into the RevenueFact table, including DateKey and TotalRevenue. Here, DateKey would map to the TimeDimension's surrogate key.

You would continue this process for each record you want to add to the database. Remember to maintain referential integrity by ensuring that foreign keys in fact tables reference the correct surrogate keys in dimension tables.Inserting data into a dimensional model can be more complex due to the need to map foreign keys to dimension surrogate keys, but it allows for efficient reporting and analysis of data.