Hospital Management System Database

Teams: Siddhesh,Palash, Luyao, Sneha, Aditi

Overview

1. Introduction
2. Entity Relation Diagram (ERD)
3. Database and Entity Tables Creation, Importing Data
4. Functions
5. Stored Procedures
6. Triggers
7. Column level Encryption
8. Views and Constraint
9. Demo
10. Reference

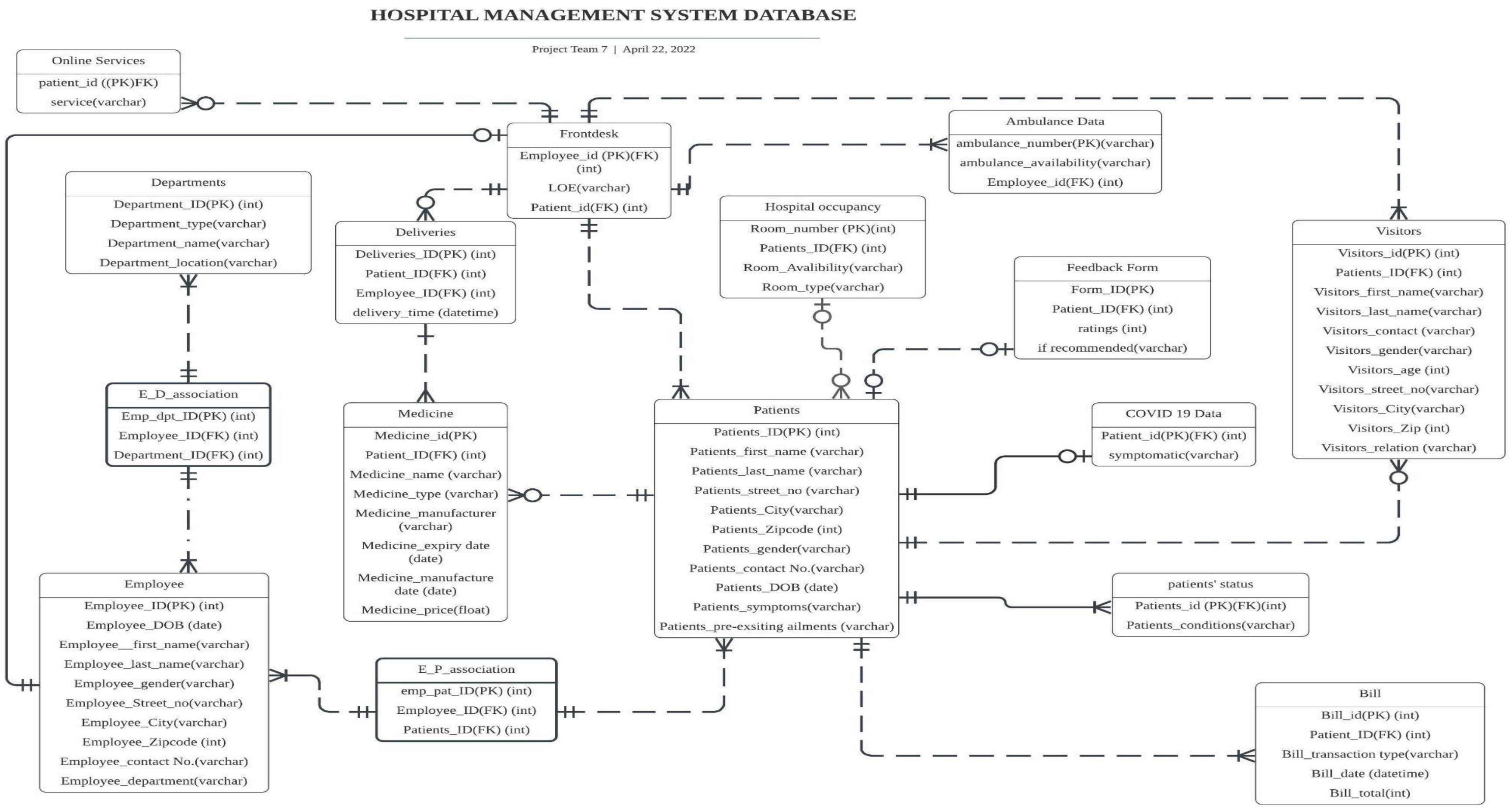
Introduction

Since a long time, hospitals have played a key role in all our lives. With advancement in technology and the on going pandemic, hospitals sizes have been growing constantly.

Our topic is creation and maintenance of hospital database and we have carefully read and analysed the business rules to create our final Entity Relationship Diagram.

This database can be used by the administrative staff of hospitals and data of all the day to day activities can be stored in a database which can be further used for analysis by hospital authorities to make decisions.

ER-DIAGRAM


HOSPITAL MANAGEMENT SYSTEM DATABASE
Project Team 7 | April 22, 2022
Online Services
patient_id ((PK)FK)
service(varchar)
Frontdesk
Employee_id (PK)(FK) (int)
LOE(varchar)
Patient_id(FK) (int)
Ambulance Data
ambulance_number(PK)(varchar)
ambulance_availability(varchar)
Employee_id(FK) (int)
Departments
Department_ID(PK) (int)
Department_type(varchar)
Department_name(varchar)
Department_location(varchar)
Deliveries
Deliveries_ID(PK) (int)
Patient_ID(FK) (int)
Employee_ID(FK) (int)
delivery_time (datetime)
Hospital occupancy
Room_number (PK)(int)
Patients_ID(FK) (int)
Room_Avalibility(varchar)
Room_type(varchar)
Feedback Form
Form_ID(PK)
Patient_ID(FK) (int)
ratings (int)
if recommended(varchar)
Visitors
Visitors_id(PK) (int)
Patients_ID(FK) (int)
Visitors_first_name(varchar)
Visitors_last_name(varchar)
Visitors_contact (varchar)
Visitors_gender(varchar)
Visitors_age (int)
Visitors_street_no(varchar)
Visitors_City(varchar)
Visitors_Zip (int)
Visitors_relation (varchar)
E_D_association
Emp_dpt_ID(PK) (int)
Employee_ID(FK) (int)
Department_ID(FK) (int)
Medicine
Medicine_id(PK)
Patient_ID(FK) (int)
Medicine_name (varchar)
Medicine_type (varchar)
Medicine_manufacturer (varchar)
Medicine_expiry date (date)
Medicine_manufacture date (date)
Medicine_price(float)
Patients
Patients_ID(PK) (int)
Patients_first_name (varchar)
Patients_last_name (varchar)
Patients_street_no (varchar)
Patients_City(varchar)
Patients_Zipcode (int)
Patients_gender(varchar)
Patients_contact No.(varchar)
Patients_DOB (date)
Patients_symptoms(varchar)
Patients_pre-exsiting ailments (varchar)
COVID 19 Data
Patient_id(PK)(FK) (int)
symptomatic(varchar)
patients' status
Patients_id (PK)(FK)(int)
Patients_conditions(varchar)
Employee
Employee_ID(PK) (int)
Employee_DOB (date)
Employee__first_name(varchar)
Employee_last_name(varchar)
Employee_gender(varchar)
Employee_Street_no(varchar)
Employee_City(varchar)
Employee_Zipcode (int)
Employee_contact No.(varchar)
Employee_department(varchar)
E_P_association
emp_pat_ID(PK) (int)
Employee_ID(FK) (int)
Patients_ID(FK) (int)
Bill
Bill_id(PK) (int)
Patient_ID(FK) (int)
Bill_transaction type(varchar)
Bill_date (datetime)
Bill_total(int)

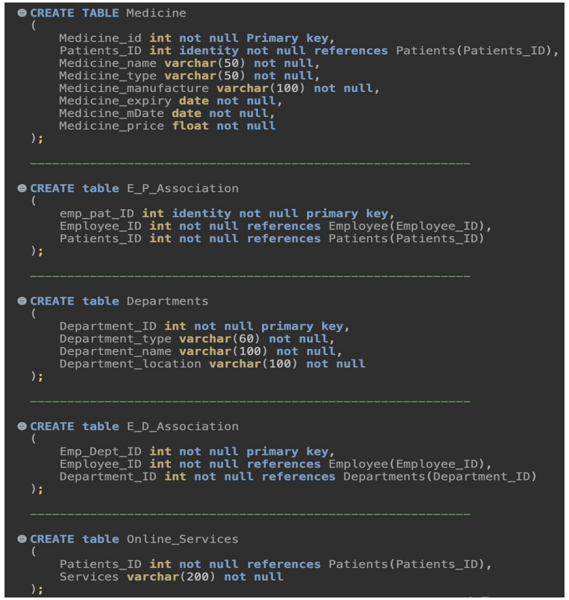
ER-DIAGRAM ENTITIES

There are 14 entities in total and 2 connecting entities which contain data about the keys for the connecting tables.

- Online Services
- Front Desk
- Departments
- Deliveries
- Medicine
- Employee
- Hospital Occupancy
- Patients
- Ambulance Data
- Covid Data
- Feedback Form
- Patient Status
- Visitors
- Bill
- E_D_ ASSOCIATION - Employee Department Association
- E_P_ASSOCIATION - Employee Patient Association

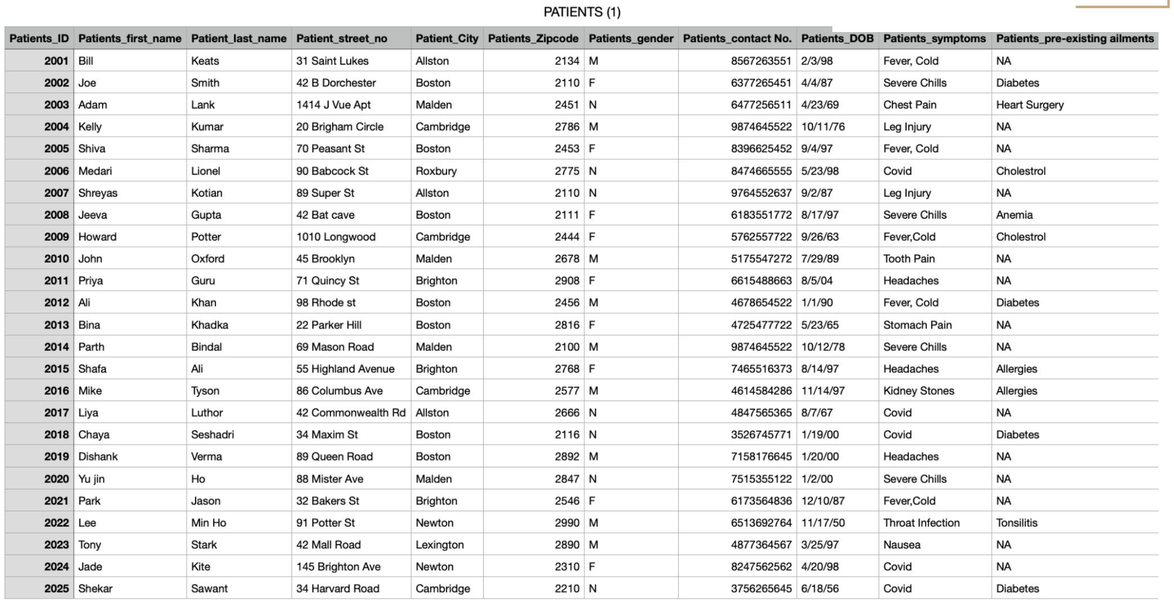
DATA CREATION

Table Creation



```
CREATE TABLE Medicine
(
    Medicine_id int not null Primary key,
    Patients_ID int identity not null references Patients(Patients_ID),
    Medicine_name varchar(50) not null,
    Medicine_type varchar(50) not null,
    Medicine_manufacture varchar(100) not null,
    Medicine_expiry date not null,
    Medicine_mDate date not null,
    Medicine_price float not null
);

---------------------------------------------------------------

CREATE table E_P_Association
(
    emp_pat_ID int identity not null primary key,
    Employee_ID int not null references Employee(Employee_ID),
    Patients_ID int not null references Patients(Patients_ID)
);

---------------------------------------------------------------

CREATE table Departments
(
    Department_ID int not null primary key,
    Department_type varchar(60) not null,
    Department_name varchar(100) not null,
    Department_location varchar(100) not null
);

---------------------------------------------------------------

CREATE table E_D_Association
(
    Emp_Dept_ID int not null primary key,
    Employee_ID int not null references Employee(Employee_ID),
    Department_ID int not null references Departments(Department_ID)
);

---------------------------------------------------------------

CREATE table Online_Services
(
    Patients_ID int not null references Patients(Patients_ID),
    Services varchar(200) not null
);
```

Raw Data



PATIENTS (1)

Patients_ID	Patients_first_name	Patient_last_name	Patient_street_no	Patient_City	Patients_Zipcode	Patients_gender	Patients_contact No.	Patients_DOB	Patients_symptoms	Patients_pre-existing ailments
2001	Bill	Keats	31 Saint Lukes	Allston	2134	M	8567263551	2/3/98	Fever, Cold	NA
2002	Joe	Smith	42 B Dorchester	Boston	2110	F	6377265451	4/4/87	Severe Chills	Diabetes
2003	Adam	Lank	1414 J Vue Apt	Malden	2451	N	6477256511	4/23/69	Chest Pain	Heart Surgery
2004	Kelly	Kumar	20 Brigham Circle	Cambridge	2786	M	9874645522	10/11/76	Leg Injury	NA
2005	Shiva	Sharma	70 Peasant St	Boston	2453	F	8396625452	9/4/97	Fever, Cold	NA
2006	Medari	Lionel	90 Babcock St	Roxbury	2775	N	8474665555	5/23/98	Covid	Cholestrol
2007	Shreyas	Kotian	89 Super St	Allston	2110	N	9764552637	9/2/87	Leg Injury	NA
2008	Jeeva	Gupta	42 Bat cave	Boston	2111	F	6183551772	8/17/97	Severe Chills	Anemia
2009	Howard	Potter	1010 Longwood	Cambridge	2444	F	5762557722	9/26/63	Fever,Cold	Cholestrol
2010	John	Oxford	45 Brooklyn	Malden	2678	M	5175547272	7/29/89	Tooth Pain	NA
2011	Priya	Guru	71 Quincy St	Brighton	2908	F	6615488663	8/5/04	Headaches	NA
2012	Ali	Khan	98 Rhode st	Boston	2456	M	4678654522	1/1/90	Fever, Cold	Diabetes
2013	Bina	Khadka	22 Parker Hill	Boston	2816	F	4725477722	5/23/65	Stomach Pain	NA
2014	Parth	Bindal	69 Mason Road	Malden	2100	M	9874645522	10/12/78	Severe Chills	NA
2015	Shafa	Ali	55 Highland Avenue	Brighton	2768	F	7465516373	8/14/97	Headaches	Allergies
2016	Mike	Tyson	86 Columbus Ave	Cambridge	2577	M	4614584286	11/14/97	Kidney Stones	Allergies
2017	Liya	Luthor	42 Commonwealth Rd	Allston	2666	N	4847565365	8/7/67	Covid	NA
2018	Chaya	Seshadri	34 Maxim St	Boston	2116	N	3526745771	1/19/00	Covid	Diabetes
2019	Dishank	Verma	89 Queen Road	Boston	2892	M	7158176645	1/20/00	Headaches	NA
2020	Yu jin	Ho	88 Mister Ave	Malden	2847	N	7515355122	1/2/00	Severe Chills	NA
2021	Park	Jason	32 Bakers St	Brighton	2546	F	6173564836	12/10/87	Fever,Cold	NA
2022	Lee	Min Ho	91 Potter St	Newton	2990	M	6513692764	11/17/50	Throat Infection	Tonsilitis
2023	Tony	Stark	42 Mall Road	Lexington	2890	M	4877364567	3/25/97	Nausea	NA
2024	Jade	Kite	145 Brighton Ave	Newton	2310	F	8247562562	4/20/98	Covid	NA
2025	Shekar	Sawant	34 Harvard Road	Cambridge	2210	N	3756265645	6/18/56	Covid	Diabetes

We used Dbeaver to import data in CSV Format.*

- A major challenge of data creation was to ensure that the tables are referenced correctly and making sure all the relations between the entities have been identified and connected accurately.
- Using the two tables E_D_Association and E_P_Association we were able to successfully create references between the tables and make sure that all the primary keys and foreign keys are correctly marked.
- Key points to keep in mind:

1. Make sure that the data types match in the ERD and the database code.
2. Check for null values of the attributes for each entity.
3. Make sure that the rows are identified and connected properly by joining columns of the table on the right one.
4. Please check all constraints are have been kept in mind while creating the database.

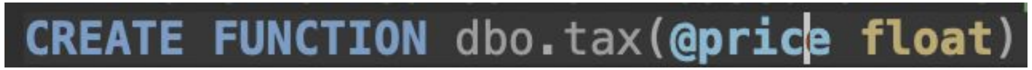
Functions

1. **Tax function**
2. **Visitors function**
3. **Medicine order function**
4. **Employee order function**
5. **Price check function**
6. **Age validation function**
7. **Age calculate function**
8. **Feedback function**
9. **Phone number validation function**

Functions

1.Tax function

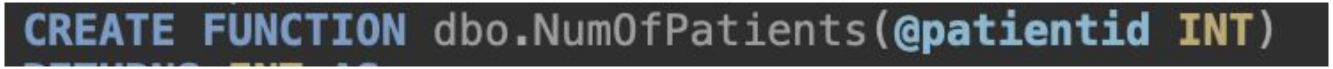
Calculated sale tax based on the price in Medine table.
the tax rate is defined to be 0.06



```
CREATE FUNCTION dbo.tax(@price float)
```

2.Visitors function

Compute columns of number of visitors for each patient.

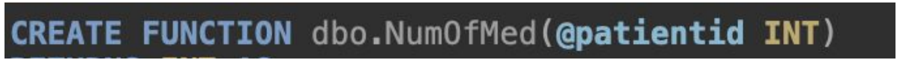


```
CREATE FUNCTION dbo.NumOfPatients(@patientid INT)
```

Functions

3.Medicine order function

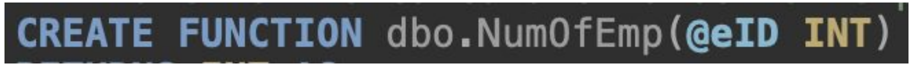
Compute columns for number of medicine orders for each patient.



```
CREATE FUNCTION dbo.NumOfMed(@patientid INT)
```

4. Employee order function

Compute columns for number of employees in each department.

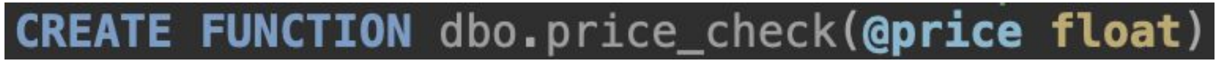


```
CREATE FUNCTION dbo.NumOfEmp(@eID INT)
```

Functions

5.Price check

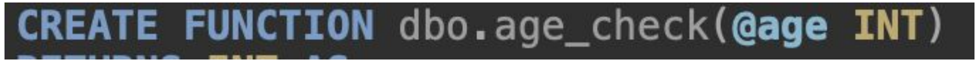
To check if the price of the medicine is valid.



```
CREATE FUNCTION dbo.price_check(@price float)
```

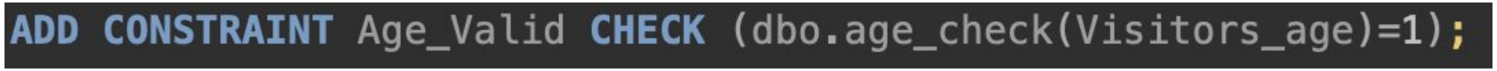
6.Age validation function(with constraint)

Check if the age is a valid age(0-200)



```
CREATE FUNCTION dbo.age_check(@age INT)
```

Constraint

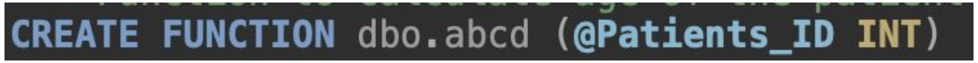


```
ADD CONSTRAINT Age_Valid CHECK (dbo.age_check(Visitors_age)=1);
```

Functions

7.Age calculate function

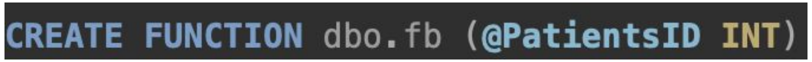
Calculated age of the patient based on patient’s DOB.



```
CREATE FUNCTION dbo.abcd (@Patients_ID INT)
```

8.Feedback function

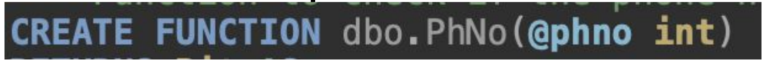
Action on feedback



```
CREATE FUNCTION dbo.fb (@PatientsID INT)
```

9.Phone number validation function

Check if the phone number is valid for 10 digits

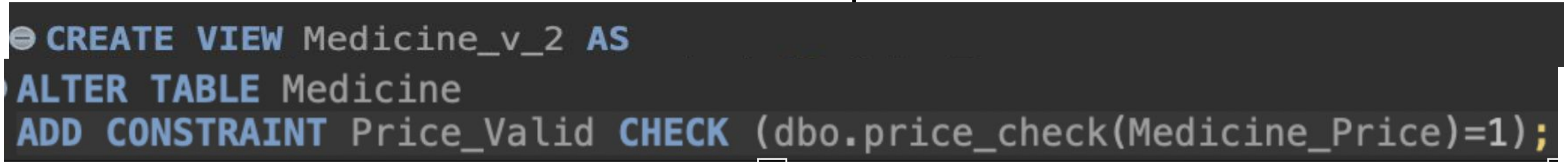


```
CREATE FUNCTION dbo.PhNo(@phno int)
```

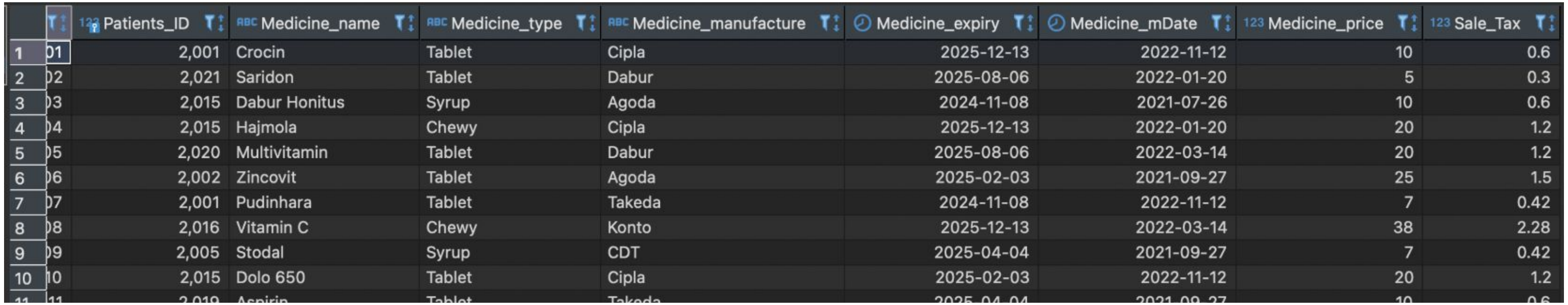
View 1 - Tax for medicine sales(with constraint)

This view camputed a column named “sale_tax” based on the “Price” in the Medicine Table, and we assume the tax rate is 0.06.

Table level check constraint on medicine price



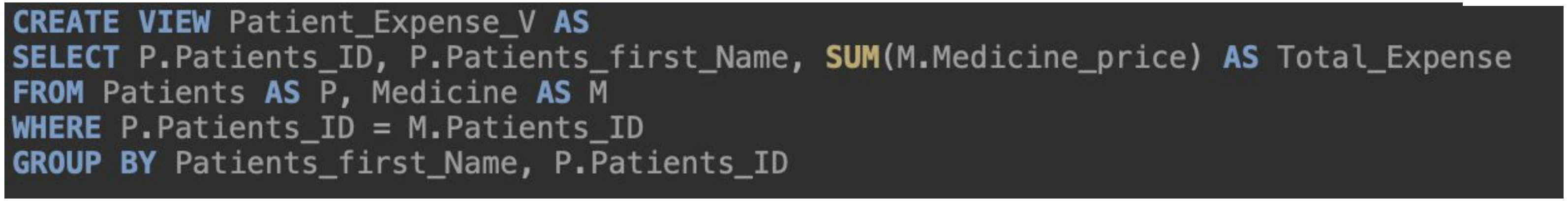
```
CREATE VIEW Medicine_v_2 AS
ALTER TABLE Medicine
ADD CONSTRAINT Price_Valid CHECK (dbo.price_check(Medicine_Price)=1);
```



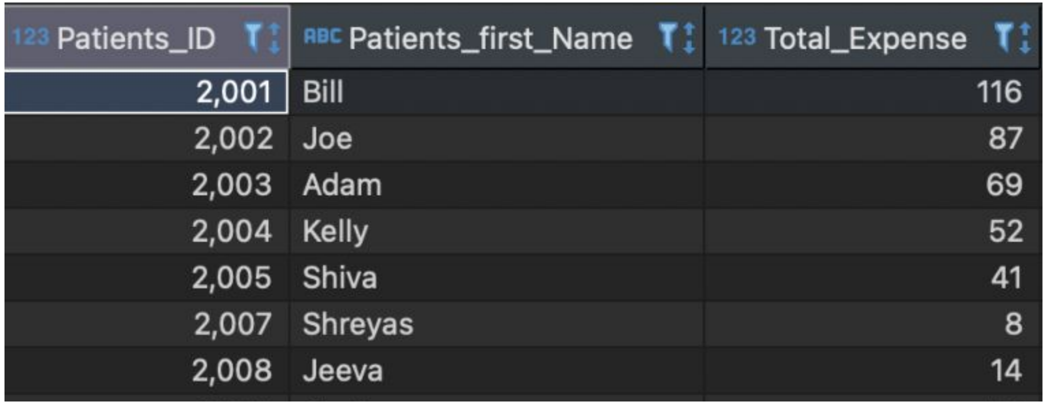
		Patients_ID	Medicine_name	Medicine_type	Medicine_manufacture	Medicine_expiry	Medicine_mDate	Medicine_price	Sale_Tax
1	01	2,001	Crocin	Tablet	Cipla	2025-12-13	2022-11-12	10	0.6
2	02	2,021	Saridon	Tablet	Dabur	2025-08-06	2022-01-20	5	0.3
3	03	2,015	Dabur Honitus	Syrup	Agoda	2024-11-08	2021-07-26	10	0.6
4	04	2,015	Hajmola	Chewy	Cipla	2025-12-13	2022-01-20	20	1.2
5	05	2,020	Multivitamin	Tablet	Dabur	2025-08-06	2022-03-14	20	1.2
6	06	2,002	Zincovit	Tablet	Agoda	2025-02-03	2021-09-27	25	1.5
7	07	2,001	Pudinhara	Tablet	Takeda	2024-11-08	2022-11-12	7	0.42
8	08	2,016	Vitamin C	Chewy	Konto	2025-12-13	2022-03-14	38	2.28
9	09	2,005	Stodal	Syrup	CDT	2025-04-04	2021-09-27	7	0.42
10	10	2,015	Dolo 650	Tablet	Cipla	2025-02-03	2022-11-12	20	1.2
11	11	2,019	Aspirin	Tablet	Takeda	2025-04-04	2021-09-27	10	0.6

View2-Total expense per patient

This view shows a total bill of medicines expenses for each patient.



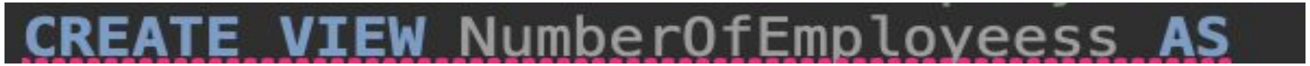
```
CREATE VIEW Patient_Expense_V AS
SELECT P.Patients_ID, P.Patients_first_Name, SUM(M.Medicine_price) AS Total_Expense
FROM Patients AS P, Medicine AS M
WHERE P.Patients_ID = M.Patients_ID
GROUP BY Patients_first_Name, P.Patients_ID
```



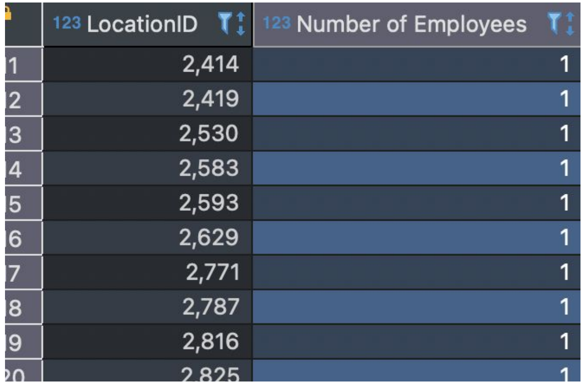
Patients_ID	Patients_first_Name	Total_Expense
2,001	Bill	116
2,002	Joe	87
2,003	Adam	69
2,004	Kelly	52
2,005	Shiva	41
2,007	Shreyas	8
2,008	Jeeva	14

View3-Employees number display

This view shows numbers of employees in each department associated with location id.



```
CREATE VIEW NumberOfEmployeess AS
```

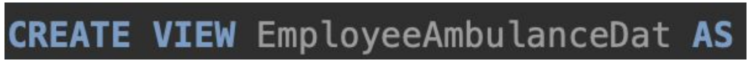


	LocationID	Number of Employees
1	2,414	1
2	2,419	1
3	2,530	1
4	2,583	1
5	2,593	1
6	2,629	1
7	2,771	1
8	2,787	1
9	2,816	1
20	2,825	1

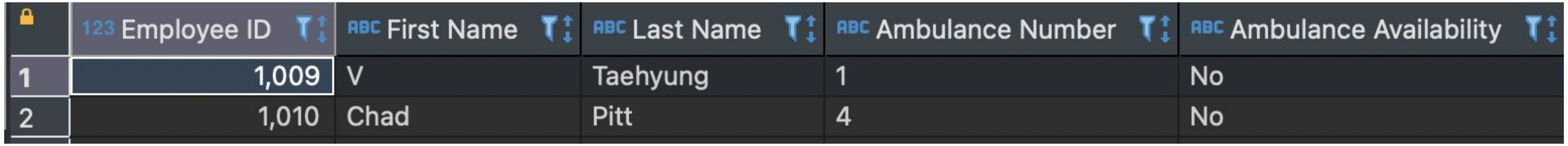
View4-Ambulance assign

Hospital Management System Database

This view shows information on the ambulance assign to employees on duty



```
CREATE VIEW EmployeeAmbulanceDat AS
```

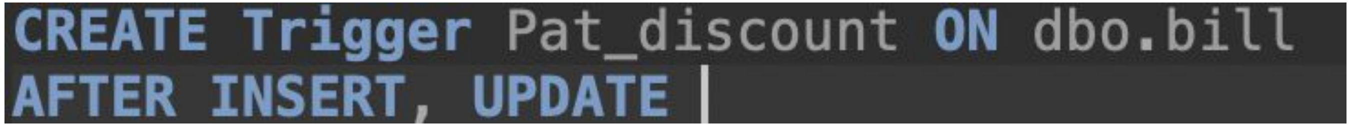


	Employee ID	First Name	Last Name	Ambulance Number	Ambulance Availability
1	1,009	V	Taehyung	1	No
2	1,010	Chad	Pitt	4	No

Triggers

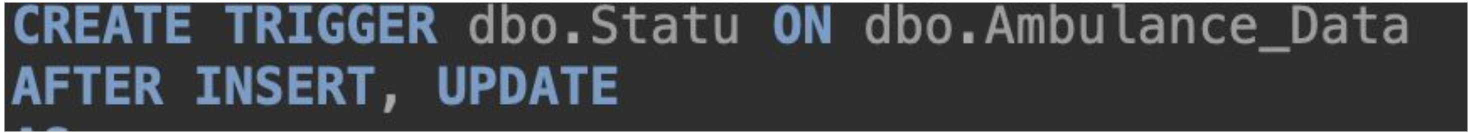
SQL codes that are automatically executed in response to any event on a particular Table.

1. Patient have bill amount over $50,000 has a 10% discount.



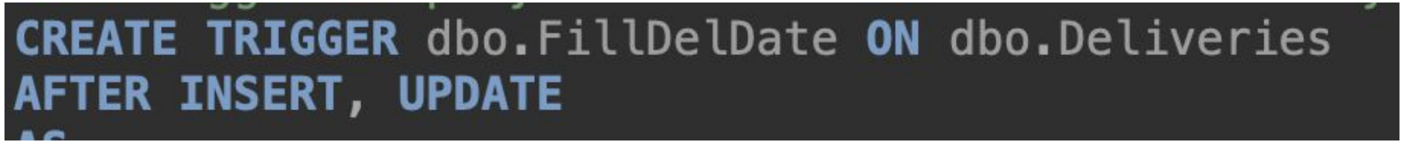
```
CREATE Trigger Pat_discount ON dbo.bill
AFTER INSERT, UPDATE
```

2. Employee availability for an ambulance assign.



```
CREATE TRIGGER dbo.Statu ON dbo.Ambulance_Data
AFTER INSERT, UPDATE
```

3. Date and time display for delivery of medicine after value inserted.

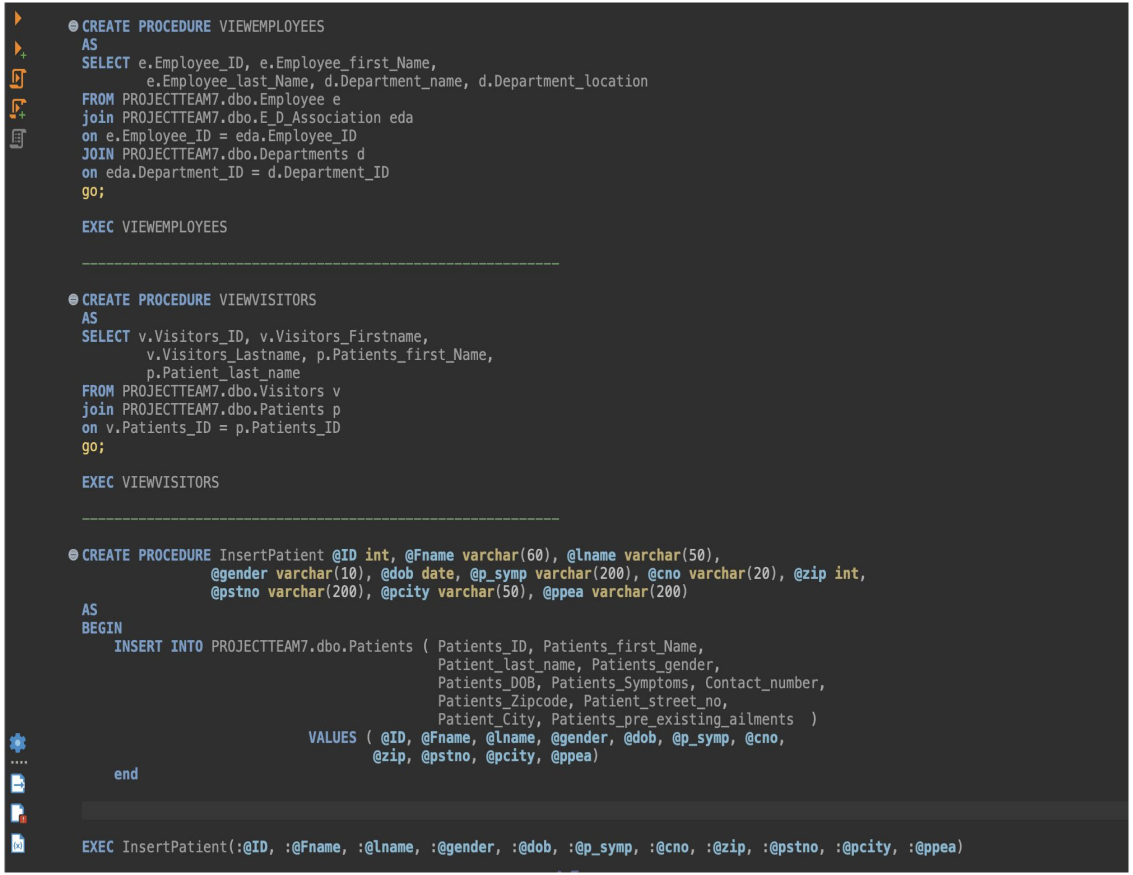


```
CREATE TRIGGER dbo.FillDelDate ON dbo.Deliveries
AFTER INSERT, UPDATE
```

Stored Procedures

The Stored Procedures in our database

- ALLPATIENTS
- AvailableAmbulance
- InsertEmployee
- InsertPatient
- VIEWCOVID
- VIEWVISITORS
- VIEWEMPLOYEES
- VIEWPATIENTS

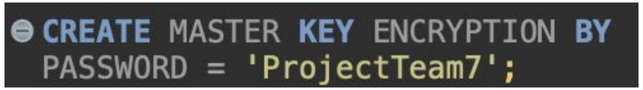


```
CREATE PROCEDURE VIEWEMPLOYEES
AS
SELECT e.Employee_ID, e.Employee_first_Name,
        e.Employee_last_Name, d.Department_name, d.Department_location
FROM PROJECTTEAM7.dbo.Employee e
join PROJECTTEAM7.dbo.E_D_Association eda
on e.Employee_ID = eda.Employee_ID
JOIN PROJECTTEAM7.dbo.Departments d
on eda.Department_ID = d.Department_ID
go;

EXEC VIEWEMPLOYEES

-----------------------------------------------------------

CREATE PROCEDURE VIEWVISITORS
AS
SELECT v.Visitors_ID, v.Visitors_Firstname,
        v.Visitors_Lastname, p.Patients_first_Name,
        p.Patient_last_name
FROM PROJECTTEAM7.dbo.Visitors v
join PROJECTTEAM7.dbo.Patients p
on v.Patients_ID = p.Patients_ID
go;

EXEC VIEWVISITORS

-----------------------------------------------------------

CREATE PROCEDURE InsertPatient @ID int, @Fname varchar(60), @lname varchar(50),
                @gender varchar(10), @dob date, @p_symp varchar(200), @cno varchar(20), @zip int,
                @pstno varchar(200), @pcity varchar(50), @ppea varchar(200)
AS
BEGIN
    INSERT INTO PROJECTTEAM7.dbo.Patients ( Patients_ID, Patients_first_Name,
                                        Patient_last_name, Patients_gender,
                                        Patients_DOB, Patients_Symptoms, Contact_number,
                                        Patients_Zipcode, Patient_street_no,
                                        Patient_City, Patients_pre_existing_ailments  )
                        VALUES ( @ID, @Fname, @lname, @gender, @dob, @p_symp, @cno,
                                @zip, @pstno, @pcity, @ppea)
    end


EXEC InsertPatient(:@ID, :@Fname, :@lname, :@gender, :@dob, :@p_symp, :@cno, :@zip, :@pstno, :@pcity, :@ppea)
```

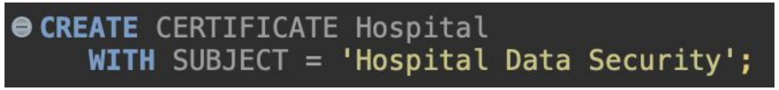
Column Level Encryption

We use the following steps for column level Encryption

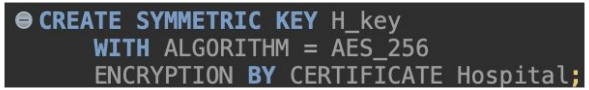
1. Create a database Master Key
2. Create a self- signed certificate
3. Config symmetric key
4. Encrypt the Data
5. Query and Verify Encryption



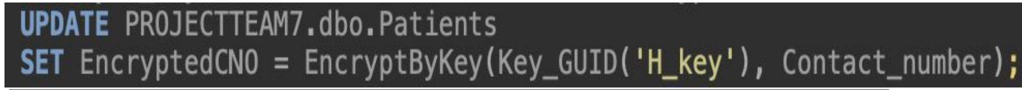
```sql
CREATE MASTER KEY ENCRYPTION BY
PASSWORD = 'ProjectTeam7';
```



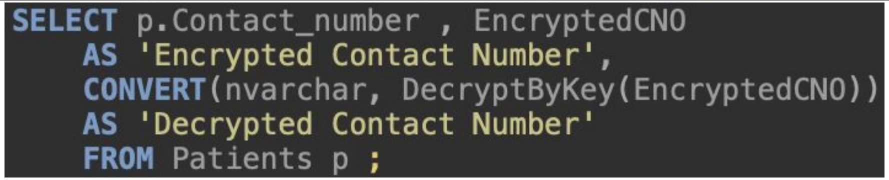
```sql
CREATE CERTIFICATE Hospital
    WITH SUBJECT = 'Hospital Data Security';
```



```sql
CREATE SYMMETRIC KEY H_key
    WITH ALGORITHM = AES_256
    ENCRYPTION BY CERTIFICATE Hospital;
```

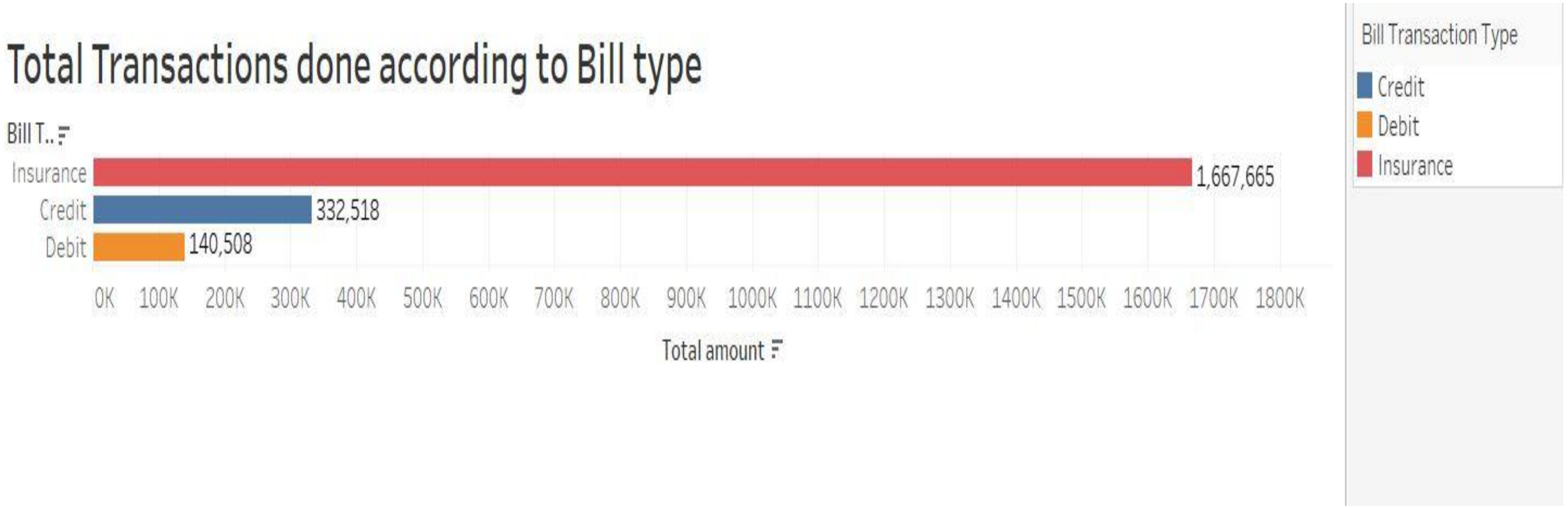


```sql
UPDATE PROJECTTEAM7.dbo.Patients
SET EncryptedCNO = EncryptByKey(Key_GUID('H_key'), Contact_number);
```

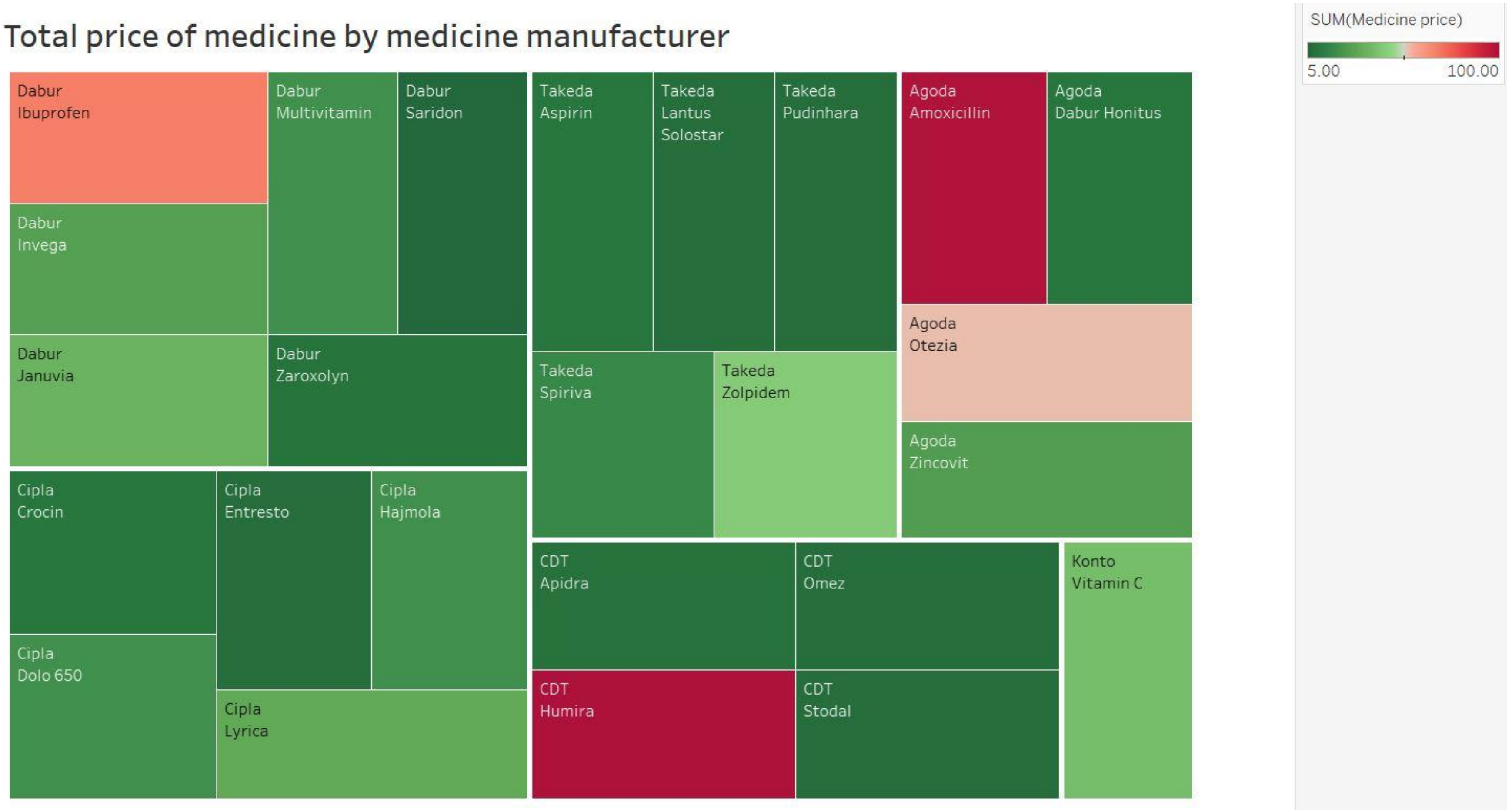


```sql
SELECT p.Contact_number , EncryptedCNO
    AS 'Encrypted Contact Number',
    CONVERT(nvarchar, DecryptByKey(EncryptedCNO))
    AS 'Decrypted Contact Number'
    FROM Patients p ;
```

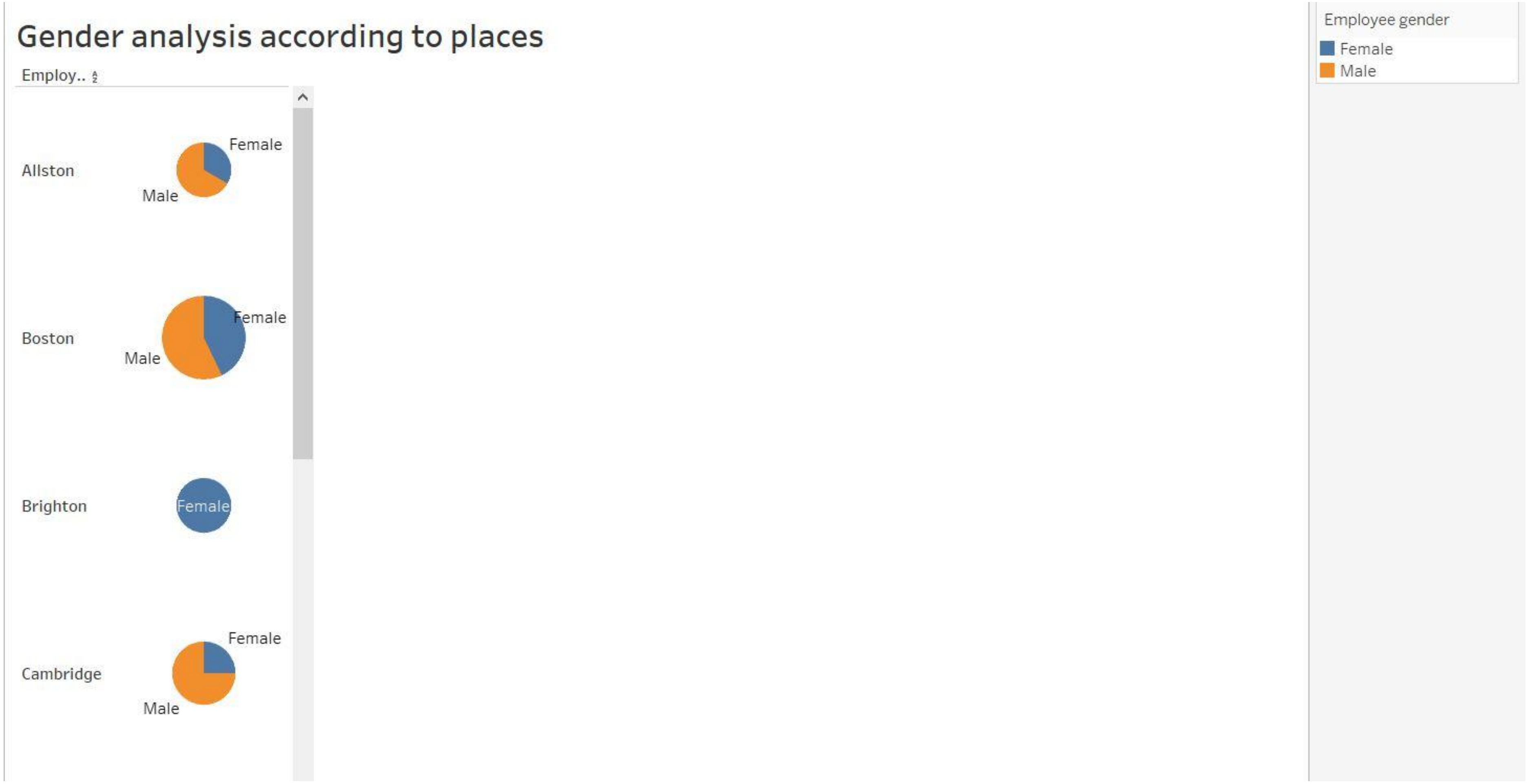
Visualizations



Visualizations

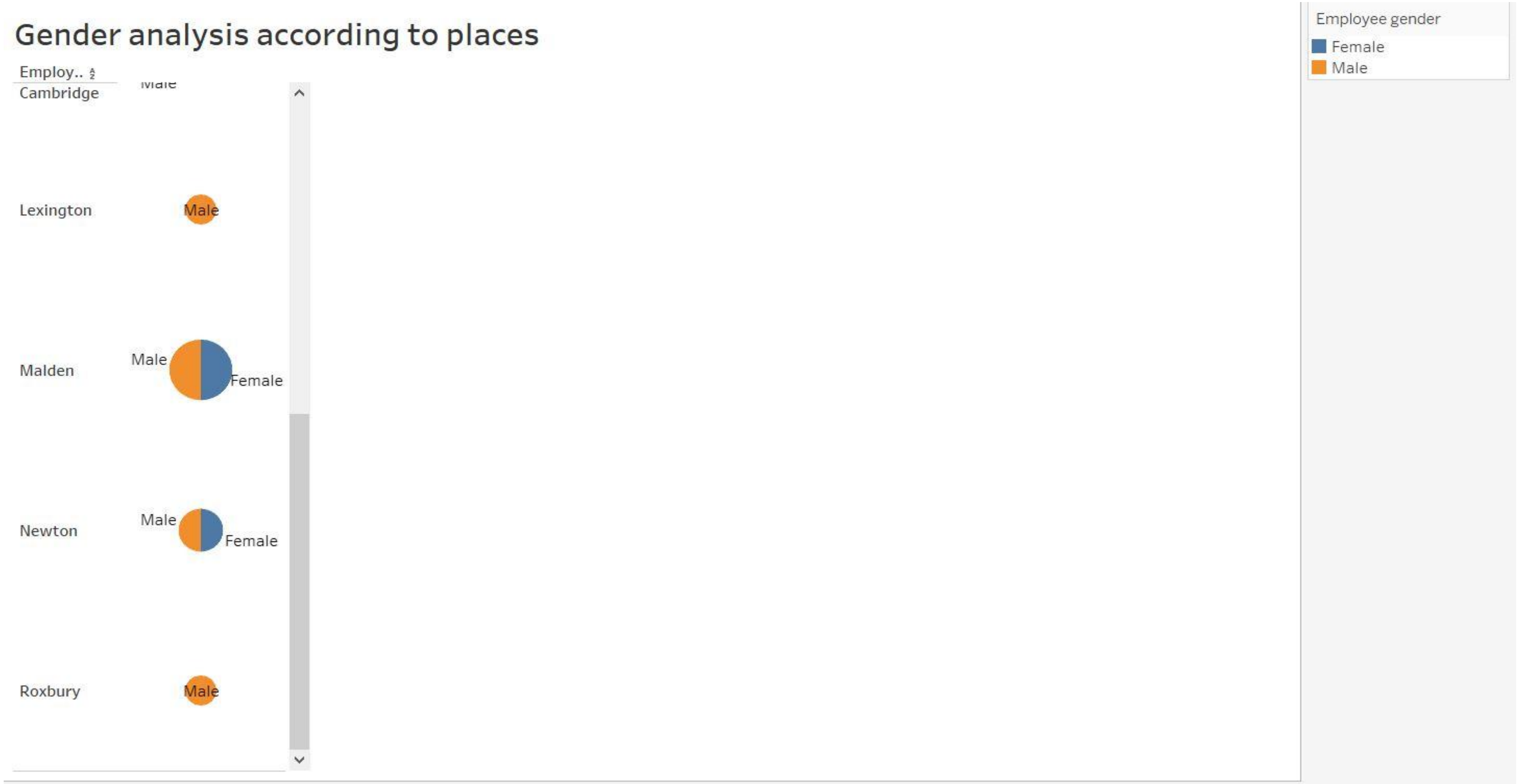


Visualizations



Gender analysis according to places

Visualizations



Demo!!!

References

We referred to Professor's class recordings and Youtube channel. Also used Google search and youtube for error solving.

https://youtube.com/playlist?list=PLDT2VyyU52ooeMWtqD2MWV9_AbtzhDEjc

https://youtube.com/playlist?list=PLDT2VyyU52opF2g7ZT3MJvFYvVMv-0ZNI

https://youtube.com/playlist?list=PL3gqFGmaw_0HAmuLaK0a5-fSXc0Gdvo4v

https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/encrypt-a-column-of-data?view=sql-server-ver15

https://www.google.com/

THANK YOU

Q&A