FINAL PROJECT REPORT

Database of 311 Service Requests for the City of Dallas

By: Rebecca D’souza

INFO 5707 – Data Modeling for Information Professionals

UNIVERSITY OF NORTH TEXAS, DENTON

* **Introduction:**

With the humongous amount of data inundating the internet on an hourly basis, there is a need for advanced technology that can process such data through categorization and organization so that it can be effectively stored in a particular location and is easily retrievable. Databases fulfill such purpose by providing a system to store enormous amounts of data that can easily be searched to access specific records through the use of queries. The purpose of this project is to demonstrate adequate knowledge on data modelling and database design to manage data as information professionals through the skills learnt in this course. The database in this project was built and implemented using MySQL for a City 311 application to service requests in the Dallas area. The database has been built from the base table provided in the City 311 application and has been normalized to create several entities and their attributes for a refined search. Understanding relational databases, normalization, creation of data dictionaries, and data integrity have been the key learning outcomes of this project.

1. **Overview:**

The 311 Service Requests from October 1, 2016 through September 30, 2018 serve as a record of emergency and non-emergency calls made into the service in Dallas, TX. The service has access to over 400 service request types of various natures that affect the quality of life of a neighbourhood, community or a citizen. However, keeping the timeline and the scope of the project in mind, only selected requests have been extracted for inclusion in this project from the base table. Some of the typical calls include feedback or requests about animal services, sidewalk obstruction, sanitation services, street and pothole repair as well as other code violations.

**b. Objectives:**

The objectives of this database are to:

* Provide information on citizens requesting services, modes of request, location of calls, department jurisdiction, case files, and services provided.
* Guide users and software developers in assessing the information and providing solutions.
* Providing transparency and accountability for the City of Dallas with information about emergency and non-emergency (standard) calls made to the city.
* Empower Dallas citizens, businesses, and visitors with information about the city.
* Give citizens access to information on how calls are handled in a timely/responsive manner.
* To improve the services by comparing the number of requests received for a particular service over the span of three years from 2016 to 2018

**c. Scope:**

The City 311 database covers the portion of the Dallas Open Data System that pertains to the 311 system. The information is limited data found within the dates October 01, 2016 to September 30, 2018. The database covers information on citizen, requests made by the citizens, location of requests, case files maintained for each request that has been made, service, and department information. The database limitations still provide ample information for users to understand what types of requests are being made and how often and well they are handled. Moreover, users can also view which requests are being handled by which department and the different requests that have seen a rise or fall over the years from 2016 to 2018. The case files associated with each request maintains information on how the request was served and whether it was resolved in an adequate response time.

* **User Requirements:**

The database allows the authorities of the City of Dallas to monitor the requests that come in from the citizens. These requests should be verified by the Citizen ID that is unique to each individual citizen residing in the city of Dallas. Other details such as the citizen’s first and last name, phone, and email may be recorded. However, the address need not be the citizen’s residential area but any location where a particular service is requested; hence, the location table can be used to obtain the exact GPS coordinates for the area that needs service. The authorities can keep a track of each request that is received by maintaining data on the mode of request and dates recorded on when the request was filed, closed or updated. Information on this can be retrieved using the unique Request No. allocated to each request. These details, in turn, are maintained in the case files recorded for each request so that they can be used to determine the time (in days) taken to respond, priority, outcome, and status of the request. It also serves as a future reference to view how similar requests were handled. Finally, the database allows users to view the different services being handled by different departments, some providing more services than the other.

* **Business Rules:**

a. Each citizen must provide their unique Citizen ID while making a request for a service.

b. Citizens can request for services anywhere in the city of Dallas and are not restricted to their area of residence.

c. Citizens can request emergency and standard services through phone, mobile app or web.

d. For each request made by a citizen, a separate case file is maintained for that particular request.

e. A citizen can make either same multiple requests or different requests for a service but will have case files associated with them equivalent to the numbers of requests made.

f. Each request must begin with the last two digits of the year in which it was made.

g. A request can only be updated on the day it was closed or after it.

h. Case file numbers are unique and must be the last three digits of the request no. that it records.

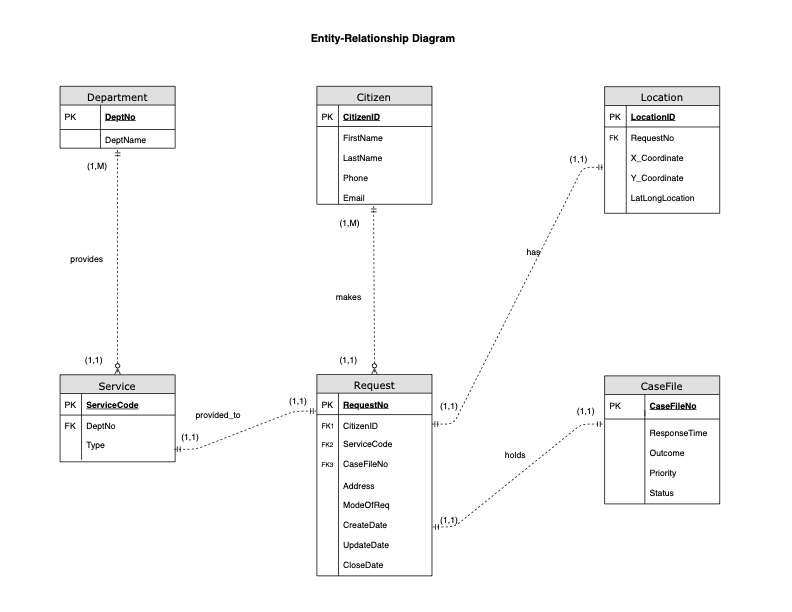
i. All the case files having an outcome that is unresolved must have status recorded as incomplete.

j. Response time is evaluated as the difference between the day the request was received and the day the service was provided.

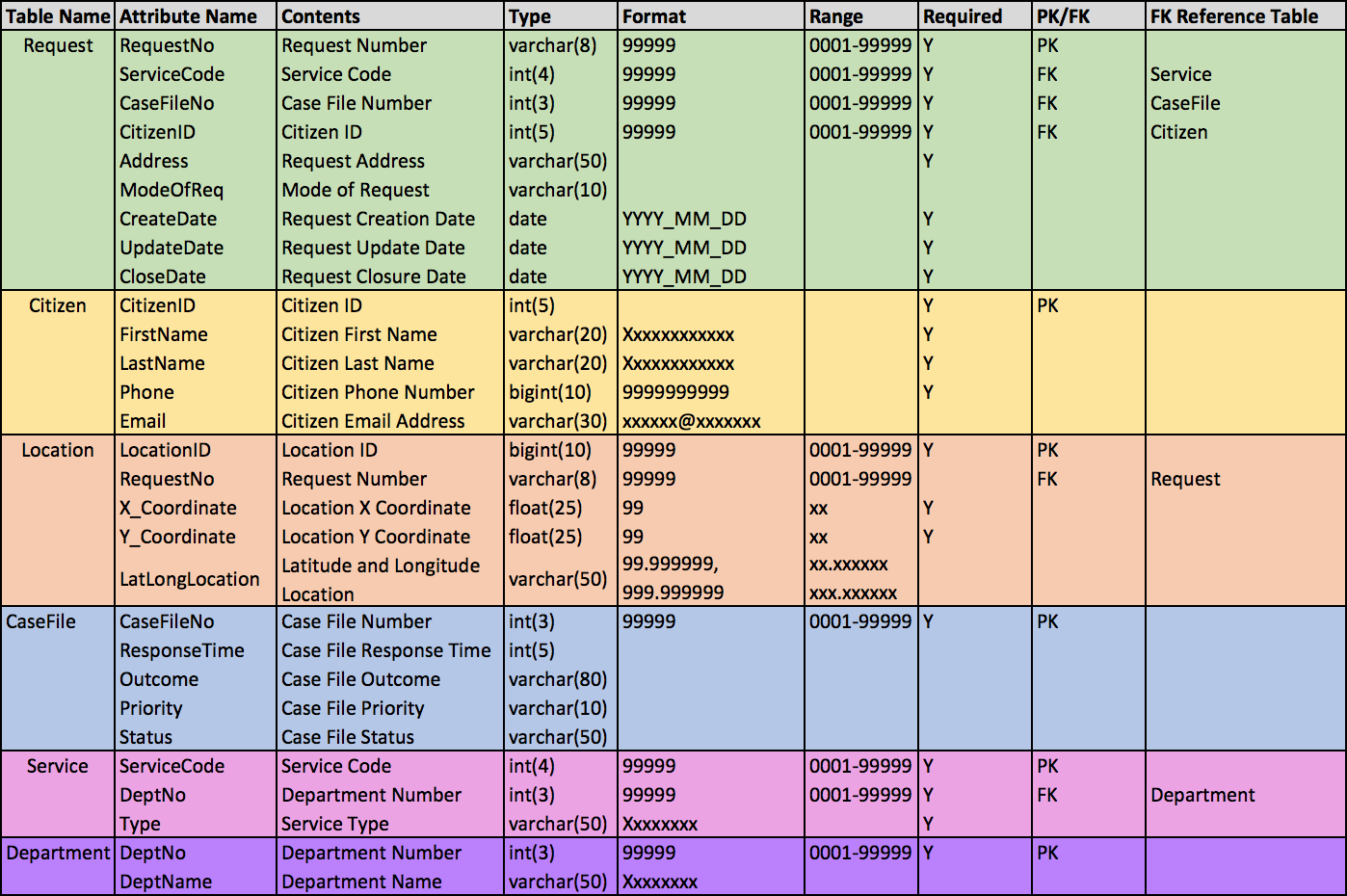
k. A department can be assigned to service several requests on any day.

l. Once service can only be handled by an individual department.

* **Entity-Relationship Diagram**



* **Data Dictionary**



* **Database creation using MySQL statements and Insert statements with real information (including screenshots)**

1. /\* Creating Citizen table \*/

CREATE TABLE Citizen (CitizenID int(5) PRIMARY KEY,

FirstName varchar(20),

LastName varchar(20),

Phone bigint(10),

Email varchar(30));

/\* Inserting values into Citizen table \*/

INSERT INTO Citizen VALUES (1, 'Brian', 'Hans', 9405946050,'brianhans@gmail.com');

INSERT INTO Citizen VALUES (2, 'Tobby', 'Mac', 9405956010,'tobbymac@my.unt.edu');

INSERT INTO Citizen VALUES (3, 'John', 'Paris', 8702954032,'jparis@gmail.com');

INSERT INTO Citizen VALUES (4, 'Danny', 'David', 9405946702,'dannydavid@hotmail.com');

INSERT INTO Citizen VALUES (5, 'Samaira', 'Vaz', 8175943019,'samaira.vaz@gmail.com');

INSERT INTO Citizen VALUES (6, 'Dave', 'Dawson', 9870067804,'dawson.dave@hotmail.com');

INSERT INTO Citizen VALUES (7, 'Rachel', 'Davidson', 7335946067,'rachel.davidson@my.unt.edu');

INSERT INTO Citizen VALUES (8, 'Darren', 'Sequeira', 7827689100,'darrensequeira@gmail.com');

INSERT INTO Citizen VALUES (9, 'Naomi', 'Crasto', 9482032201,'brianhans@gmail.com');

INSERT INTO Citizen VALUES (10, 'Mansi', 'Dsa', 8335986099,'dsa.mansi@frcrce.edu');

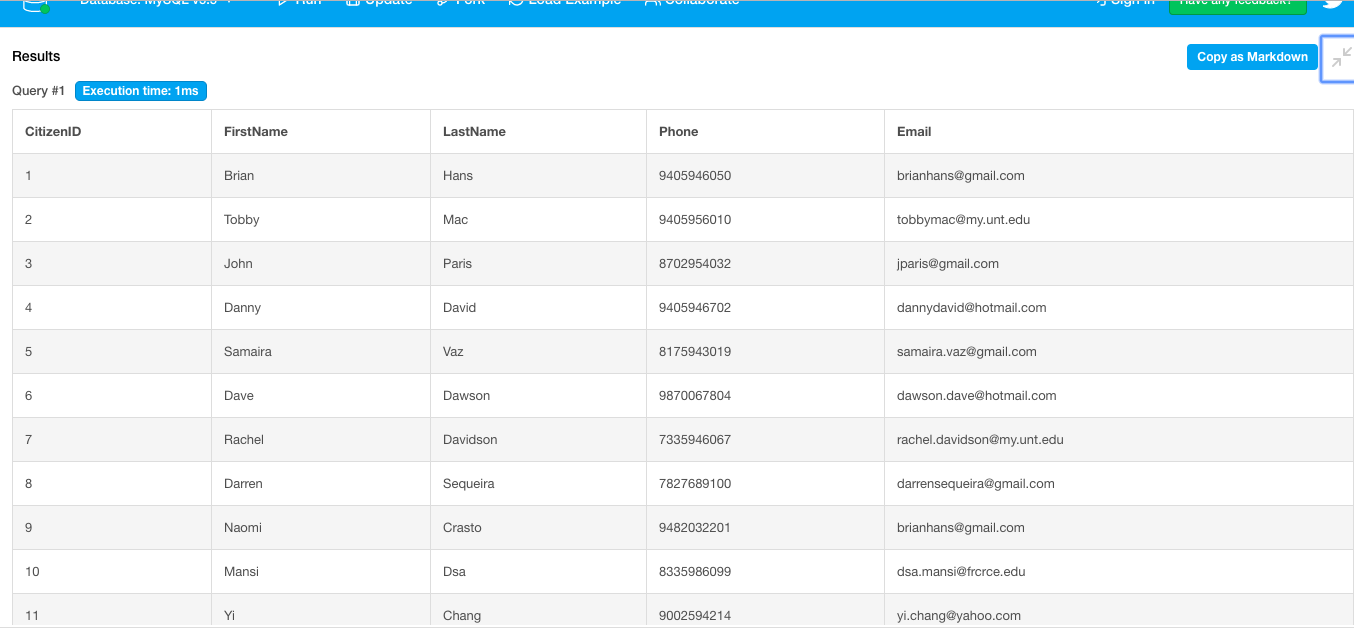
INSERT INTO Citizen VALUES (11, 'Yi', 'Chang', 9002594214,'yi.chang@yahoo.com');

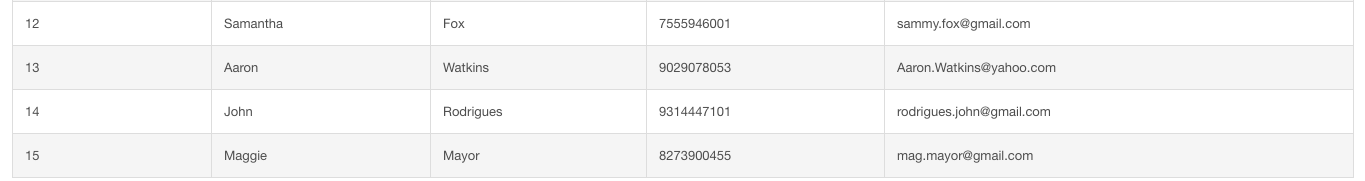
INSERT INTO Citizen VALUES (12, 'Samantha', 'Fox', 7555946001,'sammy.fox@gmail.com');

INSERT INTO Citizen VALUES (13, 'Aaron', 'Watkins', 9029078053,'Aaron.Watkins@yahoo.com');

INSERT INTO Citizen VALUES (14, 'John', 'Rodrigues',9314447101, 'rodrigues.john@gmail.com');

INSERT INTO Citizen VALUES (15, 'Maggie', 'Mayor', 8273900455,'[mag.mayor@gmail.com](mailto:mag.mayor@gmail.com)');





2. /\* Creating Department table \*/

CREATE TABLE Department (DeptNo int(3) PRIMARY KEY,

DeptName varchar(50));

/\* Inserting values into Department table \*/

INSERT INTO Department VALUES (101, 'Feedback Group');

INSERT INTO Department VALUES (102, 'Sanitation Services');

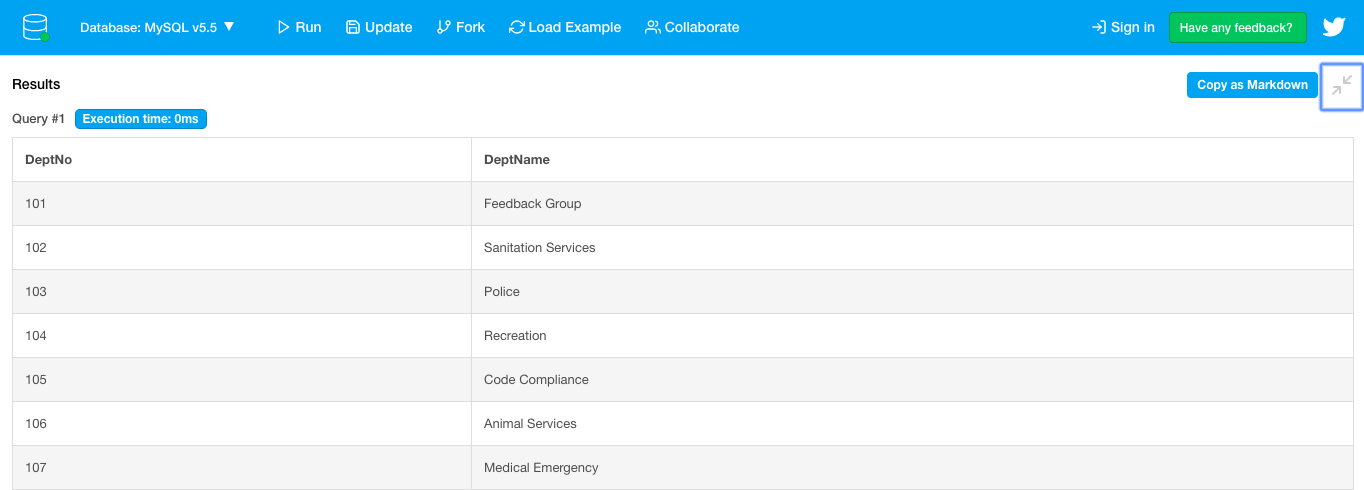
INSERT INTO Department VALUES (103, 'Police');

INSERT INTO Department VALUES (104, 'Recreation');

INSERT INTO Department VALUES (105, 'Code Compliance');

INSERT INTO Department VALUES (106, 'Animal Services');

INSERT INTO Department VALUES (107, 'Medical Emergency');



3. /\* Creating Service table \*/

CREATE TABLE Service (ServiceCode int(4) PRIMARY KEY,

DeptNo int(3),

Type varchar(50));

/\* Inserting values into Service table \*/

INSERT INTO Service VALUES (2325, 101, 'Complaint/Compliment');

INSERT INTO Service VALUES (2326, 102, 'Dead Animal Pickup');

INSERT INTO Service VALUES (2327, 103, 'Extra Police Patrol');

INSERT INTO Service VALUES (2328, 104, 'Parking Maintainance');

INSERT INTO Service VALUES (2329, 102, 'Recycling');

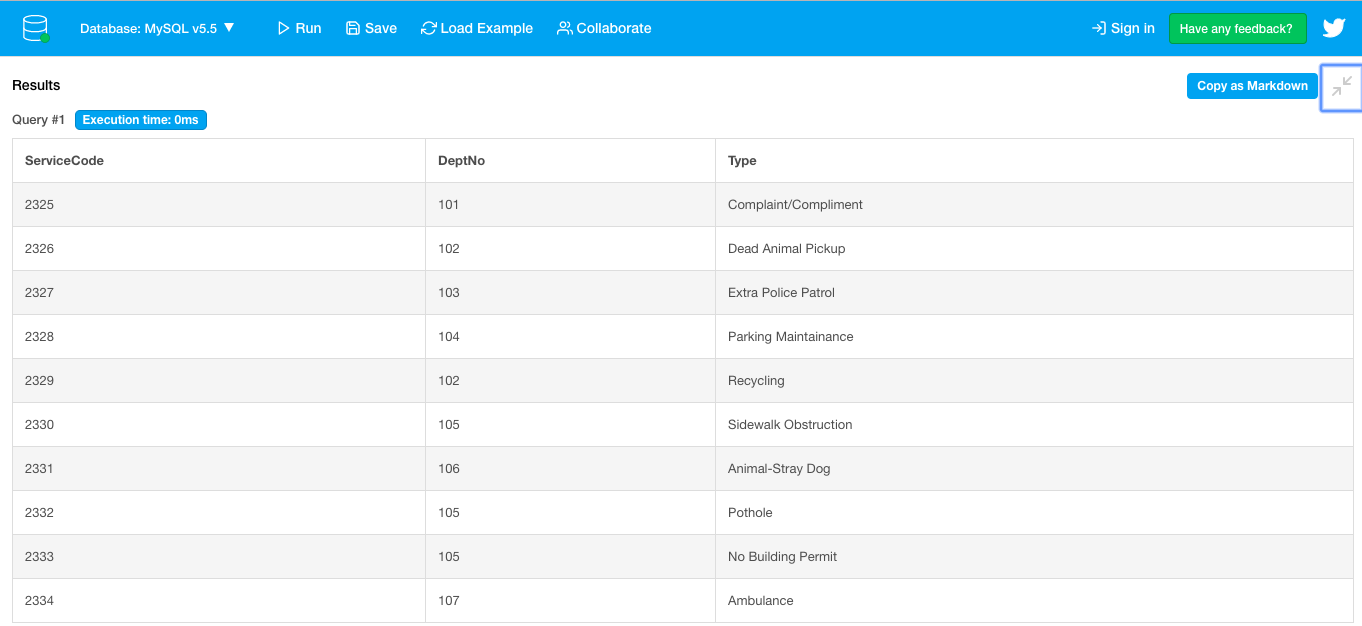
INSERT INTO Service VALUES (2330, 105, 'Sidewalk Obstruction');

INSERT INTO Service VALUES (2331, 106, 'Animal-Stray Dog');

INSERT INTO Service VALUES (2332, 105, 'Pothole');

INSERT INTO Service VALUES (2333, 105, 'No Building Permit');

INSERT INTO Service VALUES (2334, 107, 'Ambulance');



4. /\* Creating Request table \*/

CREATE TABLE Request (RequestNo varchar(8) PRIMARY KEY,

CitizenID int(5),

ServiceCode int(4),

CaseFileNo int(3),

Address varchar(50),

ModeOfReq varchar(10),

CreateDate Date,

UpdateDate Date,

CloseDate Date);

/\* Inserting values into Request table \*/

INSERT INTO Request VALUES('18-70751',10,2334,751,'6420 STRAWBERRY TRL, DALLAS, 75241','Phone','2018-06-05','2018-06-5','2018-06-05');

INSERT INTO Request VALUES('18-70484',7,2329,484,'6933 FLINTCOVE DR, DALLAS, TX 75248','Mobile App','2018-02-10','2018-02-13','2018-02-12');

INSERT INTO Request VALUES('17-28108',2,2326,108,'S ST AUGUSTINE DR & TEAGARDEN RD, DALLAS, TX','Phone','2017-04-03','2017-04-06','2017-04-05');

INSERT INTO Request VALUES('16-42170',14,2332,170,'FOREST LN & WEBB CHAPEL RD, DALLAS, TX','Web','2016-12-30','2017-01-15','2017-01-10');

INSERT INTO Request VALUES('16-25299',15,2330,299,'11442 GLEN CROSS DR, DALLAS, TX 75228','Phone','2016-10-28','2016-10-28','2016-10-28');

INSERT INTO Request VALUES('18-68223',14,2327,223,'3209 SUNNYVALE ST, DALLAS, TX 75216','Phone','2018-05-08','2018-05-20','2018-05-18');

INSERT INTO Request VALUES('17-11596',8,2328,596,'1527 KINGSLEY DR, DALLAS, TX 75216', 'Mobile App','2017-08-08','2017-08-08','2017-08-08');

INSERT INTO Request VALUES('17-28341',9,2331,341,'WILLOW GROVE RD & CRESTLINE AVE, DALLAS, TX','Phone','2017-03-19','2017-03-20','2017-03-19');

INSERT INTO Request VALUES('18-60941',4,2333,941,'7756 EL PENSADOR DR, DALLAS, TX 75248','Phone','2018-08-17','2018-09-01','2018-08-30');

INSERT INTO Request VALUES('16-25527',1,2325,527,'7203 WILCOX DR, DALLAS, TX 75232','Mobile App','2016-11-11','2016-11-11','2016-11-11');

INSERT INTO Request VALUES('18-70665',3,2331,665,'8919 HACKNEY LN, DALLAS, TX 75238','Web','2018-01-15','2018-01-17','2018-01-16');

INSERT INTO Request VALUES('17-36780',12,2329,780,'1504 SAX LEIGH DR, DALLAS, TX 75241','Phone','2017-02-10','2017-02-10','2017-02-10');

INSERT INTO Request VALUES('18-27171',6,2332,171,'E CAMP WISDOM RD & S R L THORNTON FWY, DALLAS, TX','Mobile App','2018-03-01','2018-03-06','2018-03-05');

INSERT INTO Request VALUES('16-40792',5,2334,792,'2021 SKYLARK DR, DALLAS, TX 75216','Phone','2016-12-05','2016-12-05','2016-12-05');

INSERT INTO Request VALUES('18-23756',13,2332,756,'6114 HOLLIS AVE, DALLAS, TX 75227','Mobile App','2018-05-09','2018-05-16','2018-05-15');

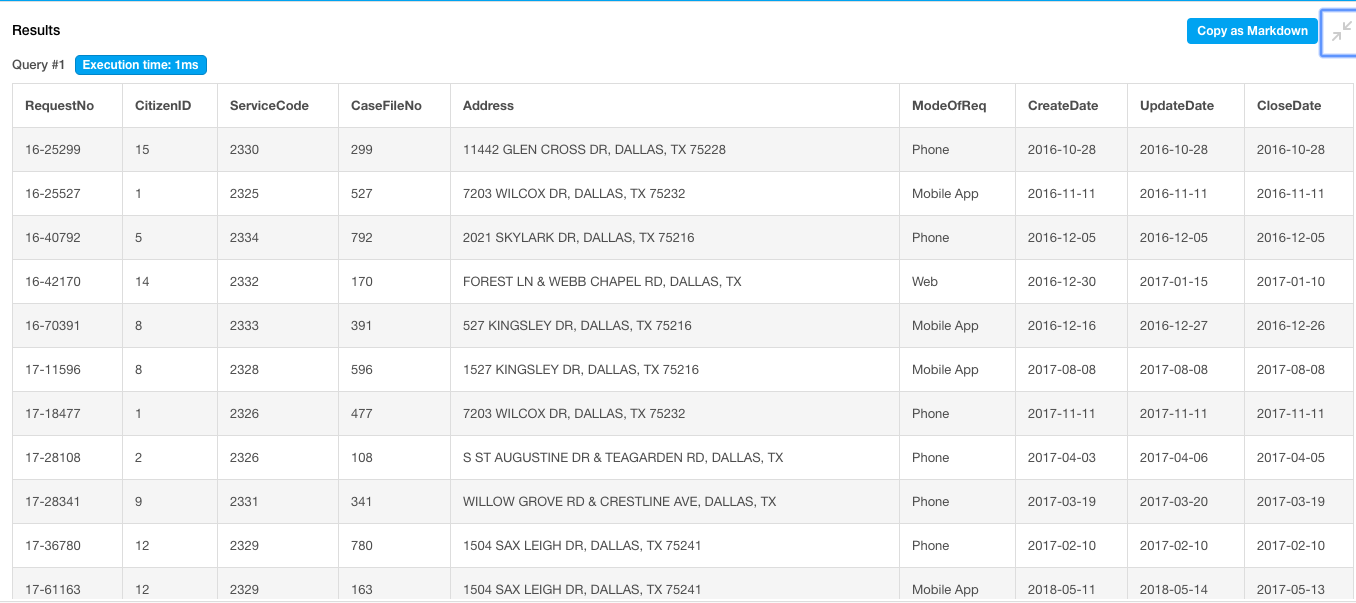
INSERT INTO Request VALUES('17-18477',1,2326,477,'7203 WILCOX DR, DALLAS, TX 75232','Phone','2017-11-11','2017-11-11','2017-11-11');

INSERT INTO Request VALUES('17-63200',11,2329,200,'4969 NASHWOOD LN, DALLAS, TX 75244','Mobile App','2017-04-13','2017-04-14','2017-04-14');

INSERT INTO Request VALUES('16-70391',8,2333,391,'527 KINGSLEY DR, DALLAS, TX 75216','Mobile App','2016-12-16','2016-12-27','2016-12-26');

INSERT INTO Request VALUES('18-43587',1,2331,587,'7203 WILCOX DR, DALLAS, TX 75232','Phone','2018-01-16','2018-01-18','2018-01-17');

INSERT INTO Request VALUES('17-61163',12,2329,163,'1504 SAX LEIGH DR, DALLAS, TX 75241','Mobile App','2018-05-11','2018-05-14','2017-05-13');



5. /\* Creating CaseFile table \*/

CREATE TABLE CaseFile (CaseFileNo int(3) PRIMARY KEY,

ResponseTime int(5),

Outcome varchar(80),

Priority varchar(10),

Status varchar(50));

/\* Inserting values into CaseFile table \*/

INSERT INTO CaseFile VALUES (751,0, 'Resolved - Information Provided','Emergency','Closed');

INSERT INTO CaseFile VALUES (484,2, 'Resolved - Problem Corrected by City','Standard','Closed');

INSERT INTO CaseFile VALUES (108,2, 'Resolved - Problem Corrected by City','Emergency','Closed');

INSERT INTO CaseFile VALUES (170,11,'Unresolved - Unable to Confirm/Locate Problem','Standard','Incomplete');

INSERT INTO CaseFile VALUES (299,0, 'Resolved - Through Citizen Action','Standard','Closed');

INSERT INTO CaseFile VALUES (223,0, 'Resolved - Referred to City Police','Emergency','Closed');

INSERT INTO CaseFile VALUES (596,0, 'Resolved - Through City Action','Standard','Closed');

INSERT INTO CaseFile VALUES (341,0, 'Unresolved - Unable to Confirm/Locate Problem','Standard','Incomplete');

INSERT INTO CaseFile VALUES (941,13,'Resolved - Through City Action','Standard','Closed');

INSERT INTO CaseFile VALUES (527,0, 'Resolved - Information Provided','Standard','Closed');

INSERT INTO CaseFile VALUES (665,1, 'Information received; patrols will be scheduled based on priority','Standard','Under Progress');

INSERT INTO CaseFile VALUES (780,0, 'Resolved - Problem Corrected by City','Standard','Closed');

INSERT INTO CaseFile VALUES (171,4, 'Resolved - Through City Action','Standard','Closed');

INSERT INTO CaseFile VALUES (792,0, 'Resolved - Information Provided','Emergency','Closed');

INSERT INTO CaseFile VALUES (756,6, 'Unresolved - Unable to Confirm/Locate Problem','Standard','Incomplete');

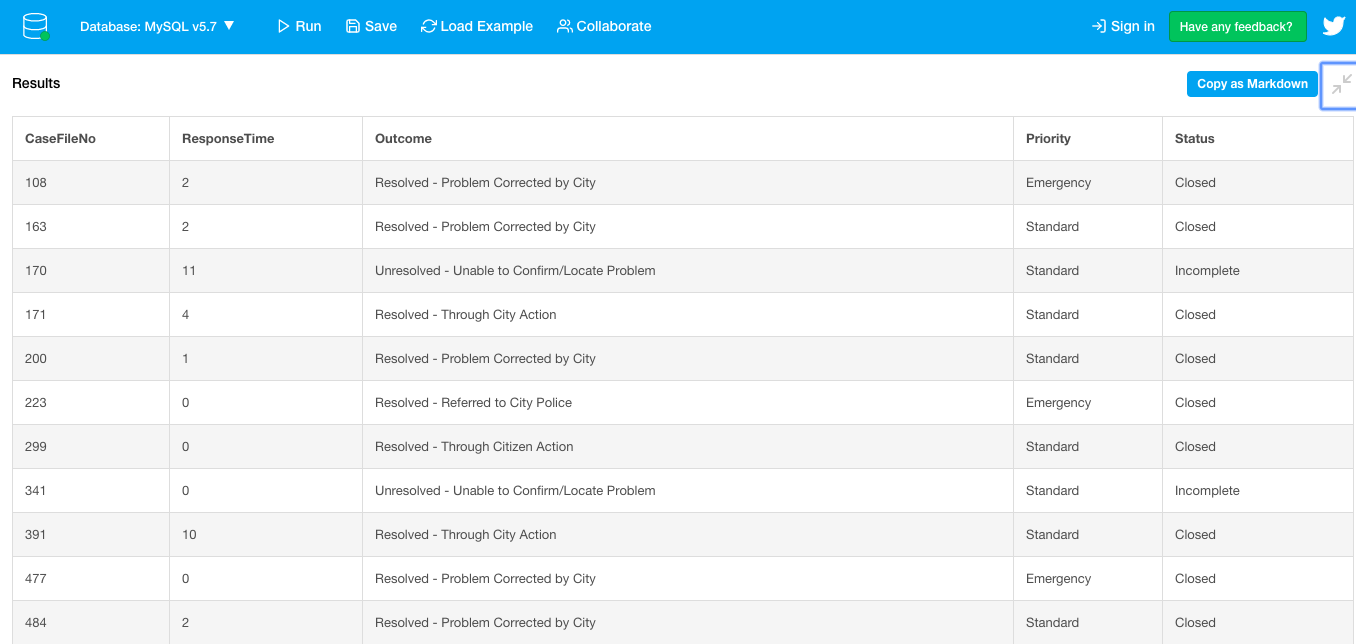
INSERT INTO CaseFile VALUES (477,0, 'Resolved - Problem Corrected by City','Emergency','Closed');

INSERT INTO CaseFile VALUES (200,1, 'Resolved - Problem Corrected by City','Standard','Closed');

INSERT INTO CaseFile VALUES (391,10,'Resolved - Through City Action','Standard','Closed');

INSERT INTO CaseFile VALUES (587,1, 'Resolved - Animal Picked Up','Emergency','Closed');

INSERT INTO CaseFile VALUES (163,2, 'Resolved - Problem Corrected by City','Standard','Closed');



6. /\* Creating Location table \*/

CREATE TABLE Location (LocationID bigint(10) PRIMARY KEY,

RequestNo varchar(8),

X\_Coordinate float(10),

Y\_Coordinate float(10),

LatLongLocation varchar(50));

/\* Inserting values into Location table \*/

INSERT INTO Location VALUES (1313282561,'18-70751',2503560.54120737, 6929065.81747519,'(32.664372°, -96.761264°)');

INSERT INTO Location VALUES (1313066763,'18-70484',2493716.28601761, 7040873.67059449,'(32.971992°, -96.78488°)');

INSERT INTO Location VALUES (1300299422,'17-28108',2535451.20544399, 6933876.42107811,'(32.676079°, -96.65728°)');

INSERT INTO Location VALUES (1285121111,'16-42170',2518247.77026552, 6988591.84857261,'(32.827434°, -96.710155°)');

INSERT INTO Location VALUES (1313256299,'16-25299',2485085.24979944, 6987491.24356711,'(32.8243°, -96.819701°)');

INSERT INTO Location VALUES (1300379046,'18-68223',2472662.29359705, 7049664.55706696,'(32.997263°, -96.855501°)');

INSERT INTO Location VALUES (1299746961,'17-11596',2498007.85075559, 6988576.70370295,'(32.8282°, -96.774933°)');

INSERT INTO Location VALUES (1300395904,'17-28341',2501382.74313274, 6978910.35960111,'(32.801631°, -96.765406°)');

INSERT INTO Location VALUES (1300655768,'18-60941',2496227.04199791, 6973055.87637027,'(28.060474°, -80.693431°)');

INSERT INTO Location VALUES (1300733242,'16-25527',2490727.20770729, 6970747.50199133,'(37.411966°, -79.138291°)');

INSERT INTO Location VALUES (1300074434,'18-70665',2486682.77872619, 6971202.00749142,'(32.781464°, -96.814226°)');

INSERT INTO Location VALUES (1299214592,'17-36780',2486296.51644893, 6959236.75759438,'(32.747811°, -96.81563°)');

INSERT INTO Location VALUES (1301135735,'18-27171',2518765.60281629, 7004333.5754407,'(32.87052°, -96.707508°)');

INSERT INTO Location VALUES (1299806168,'16-40792',2548272.99251031, 6933650.36392709,'(32.674748°, -96.615628°)');

INSERT INTO Location VALUES (1311990744,'18-23756',2521991.40688814, 6965153.82184088,'(32.762732°, -96.699076°)');

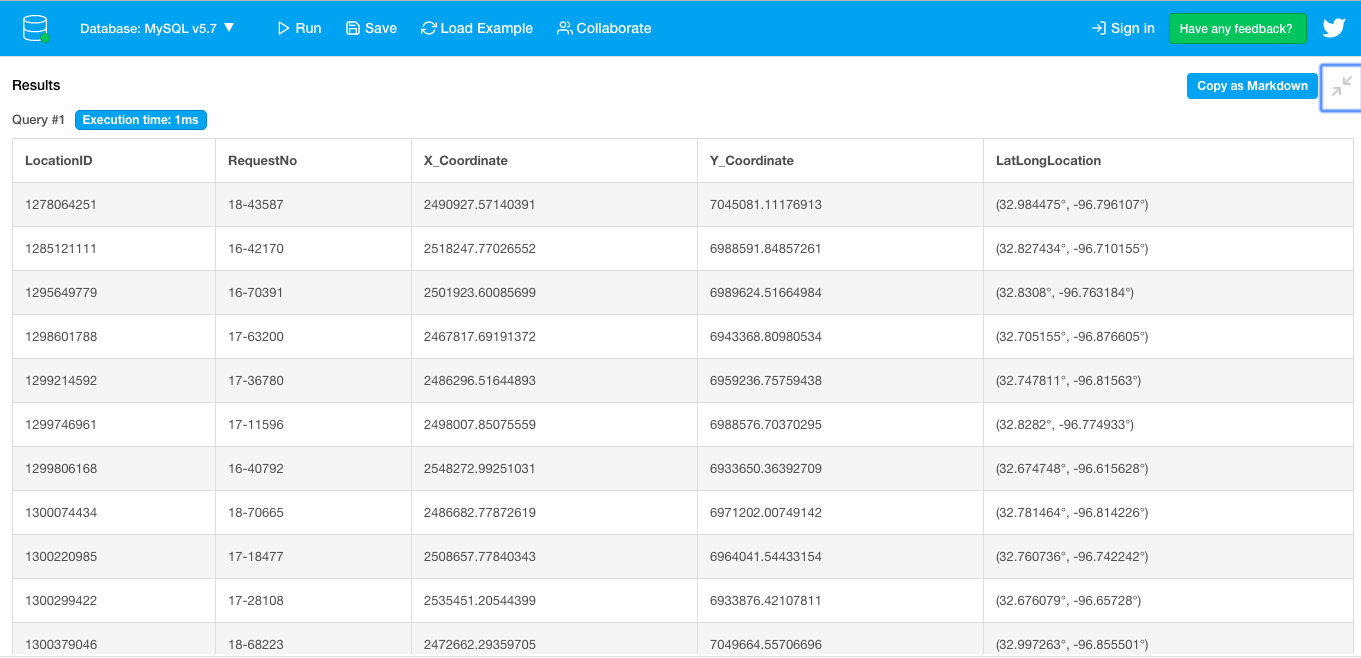
INSERT INTO Location VALUES (1300220985,'17-18477',2508657.77840343, 6964041.54433154,'(32.760736°, -96.742242°)');

INSERT INTO Location VALUES (1298601788,'17-63200',2467817.69191372, 6943368.80980534,'(32.705155°, -96.876605°)');

INSERT INTO Location VALUES (1295649779,'16-70391',2501923.60085699, 6989624.51664984,'(32.8308°, -96.763184°)');

INSERT INTO Location VALUES (1278064251,'18-43587',2490927.57140391, 7045081.11176913,'(32.984475°, -96.796107°)');

INSERT INTO Location VALUES (1300718255,'17-61163',2467933.04836044, 7017800.14934219,'(32.909642°, -96.872589°)');



* **MySQL Query Statements using DB Fiddle MySQL v5.5 (including screenshots)**

/\* Extracting data from multiple tables using JOIN \*/

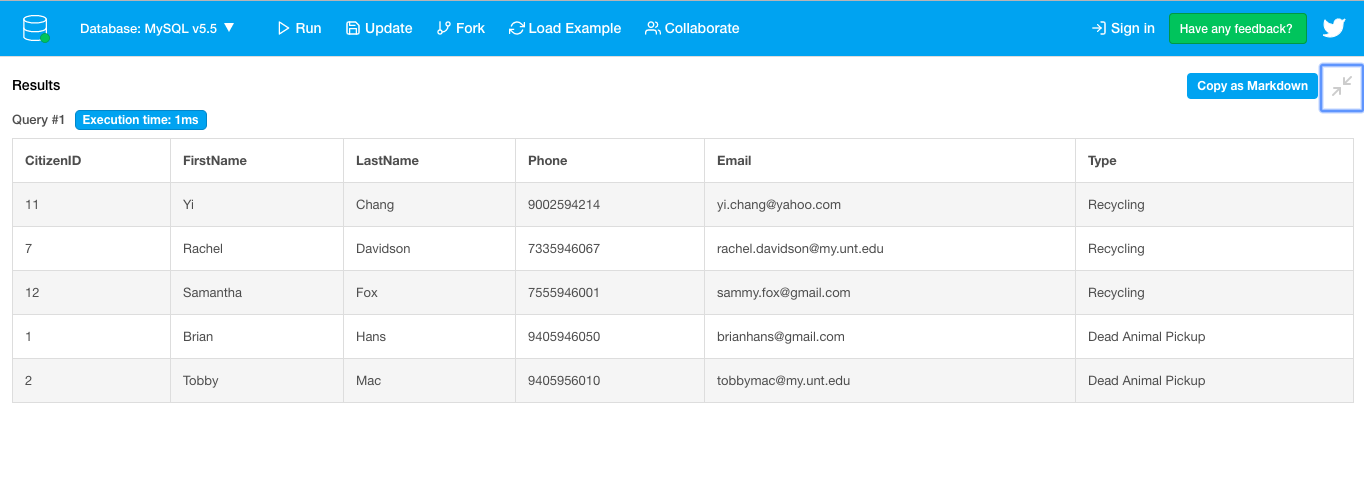
1. Display details of those Citizens along with the Service types who have requested services from the department named Sanitation Services. Sort the data by the citizen’s last name. Make sure to eliminate duplicates.

SELECT DISTINCT C.\*, S.Type

FROM Citizen C JOIN Request R ON C.CitizenID = R.CitizenID JOIN Service S ON R.ServiceCode = S.ServiceCode JOIN Department D ON S.DeptNo = D.DeptNo

WHERE DeptName = 'Sanitation Services'

ORDER BY LastName;



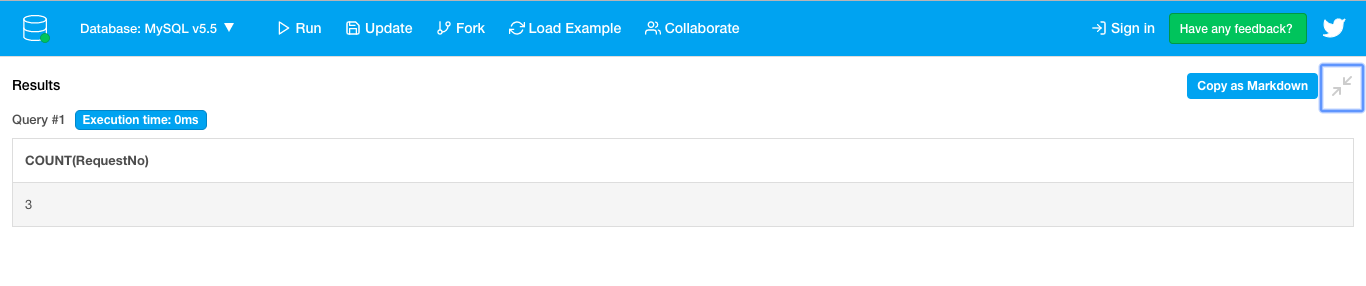
/\* Using Aggregate Function COUNT\*/

2. How many requests services have been provided to the citizen named Brian Hans?

SELECT COUNT(RequestNo)

FROM Citizen C JOIN Request R ON C.CitizenID = R.CitizenID

WHERE FirstName = 'Brian' AND LastName = 'Hans';



/\* Using Built-In functions : Column Operation CONCAT and String Function SUBSTRING along with IN operator to determine Set Membership \*/

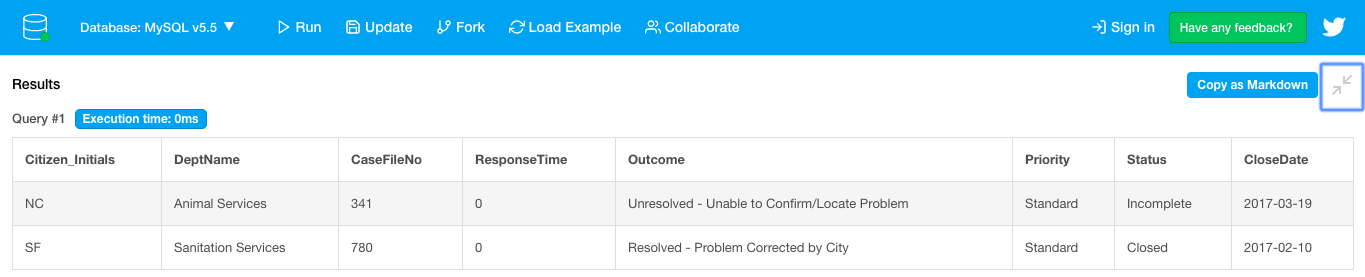
3. Display the concatenated initials of the Citizen’s first and last names along with the Department name, CaseFile details, and date of when the requests were closed in February and March of 2017. Label the column for the Citizen’s initials as Citizen\_Initials.

SELECT CONCAT(SUBSTRING(C.FirstName,1,1),SUBSTRING(C.LastName,1,1)) AS Citizen\_Initials,

DeptName, CF.\*, CloseDate

FROM Citizen C JOIN Request R ON C.CitizenID = R.CitizenID JOIN Service S ON S.ServiceCode=R.ServiceCode JOIN Department D ON D.DeptNo=S.DeptNo JOIN CaseFile CF ON CF.CaseFileNo = R.CaseFileNo

WHERE Month(CloseDate) IN (2,3) AND Year(CloseDate) = 2017;



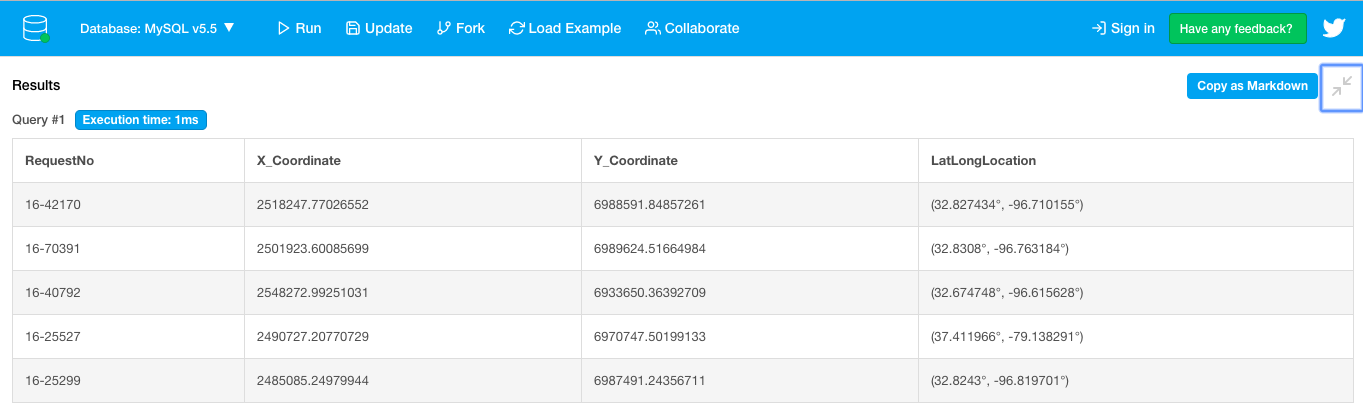
/\* Using LIKE operator in Pattern Matching \*/

4. Using only a single table named Location, display the Request numbers along with the X and Y coordinates and the Latitude-Longitude location for all the requests made in the year 2016. (Hint: All request numbers begin with the last two digits of the year in which the requests were made)

SELECT DISTINCT RequestNo, X\_Coordinate, Y\_Coordinate, LatLongLocation

FROM Location

WHERE RequestNo LIKE '16%';



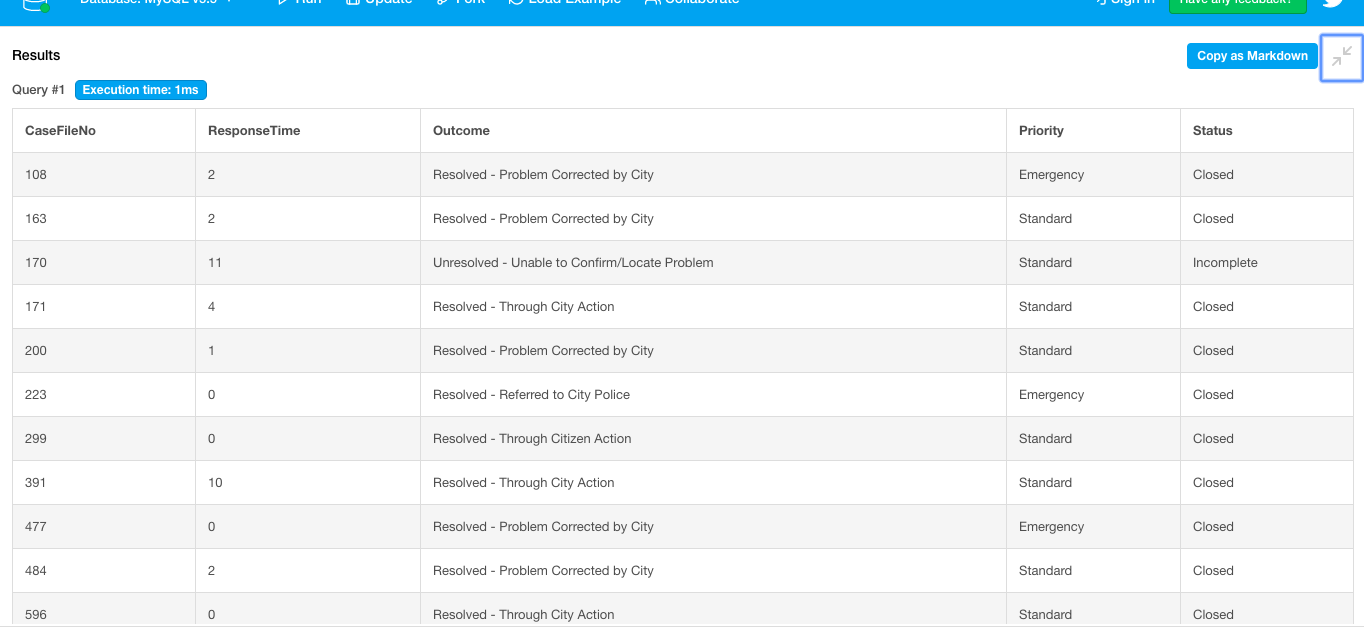
/\* Using Subqueries in WHERE clause along with BETWEEN \*/

5. Display CaseFile details for the requests served by Department Numbers greater than 101 but less than 106. (Hint: Use BETWEEN)

SELECT \*

FROM CaseFile

WHERE CaseFileNo IN (SELECT CaseFileNo FROM Request WHERE ServiceCode IN (SELECT ServiceCode FROM Service WHERE DeptNo BETWEEN 102 AND 105));



/\* Using GROUP BY clause \*/

6. List the number of services provided by each department along with their names and department numbers. Label this column as “No of Services”. (Hint: Use GROUP BY)

SELECT S.DeptNo, D.DeptName, COUNT(ServiceCode) AS "No of Services"

FROM Service S JOIN Department D ON S.DeptNo = D.DeptNo

GROUP BY D.DeptNo, D.DeptName;

