**Database for an Educational Institution**

[Mahima Chowdary Kodali](mailto:mahimachowdarykoda@lewisu.edu)

**Student ID -** 30065273

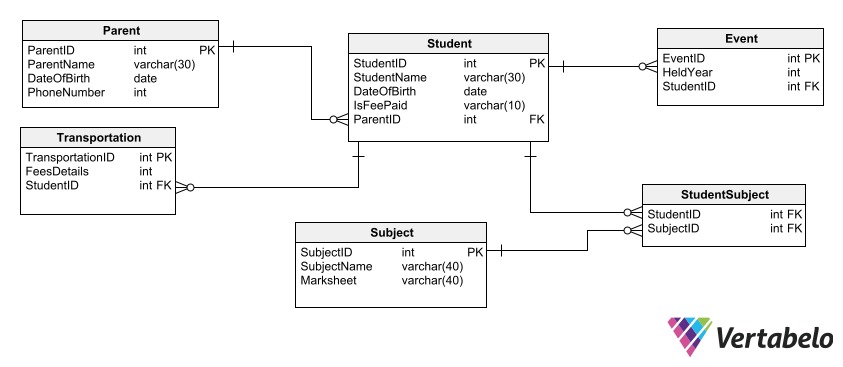
**Email-** mahimachowdarykoda@lewisu.edu

**Data Definition Language Scripts**

**Project Summarization**

In this project, we are generating SQL commands to create the database tables and load data files to database tables. We have used two tools in the project that are Vertabelo and Maria DB. In the first part of the project, vertabelo is used to generate SQL commands to build the database. On first we have loaded the ER diagram of the same project to the Vertabelo software. Using the tab options, select the SQL icon for generating the SQL scripts. An new window will pop up and then click on the generate option to export the SQL file. In the second part of the project we uses the Maria DB for loading the data files. On first, we have to convert the .sql file to .csv file and load the csv file in Maria DB, Also we have written 12 queries for the SQL commands. The queries were based on insert statement, update statement, delete statement and simple select statement.

**ER – Diagram**



**Fig 1: Entity Relationship Diagram**

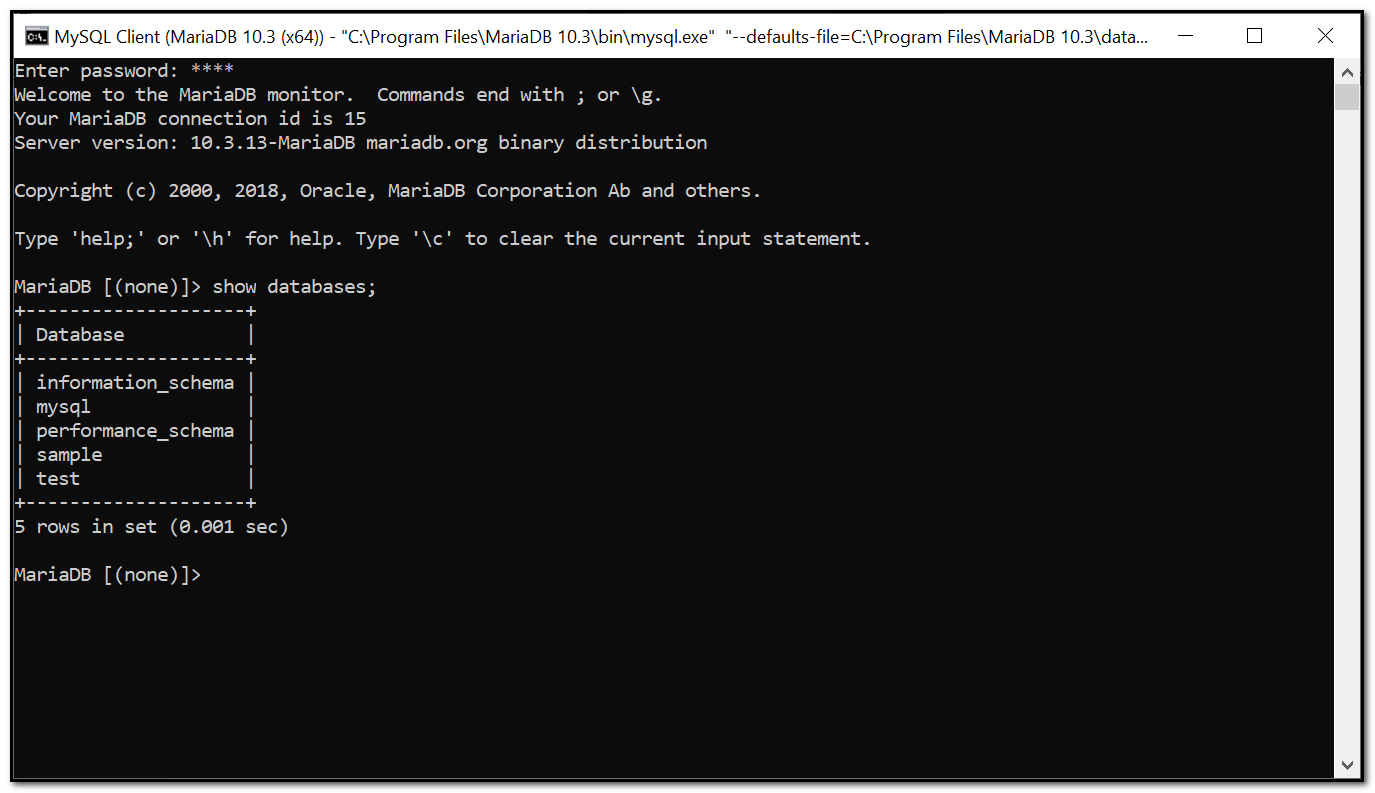
**Entity & Attribute Description**

In the ER diagram we have developed, there are 6 entities have taken place that includes student, parent, event, student subject and transportation. The student entity have provide detailed information about the student. Similarly the all other entities will provide their relevant details of the specific entity. The entity has set of attributes and labels with more parameters stored in a table. Each entity have specific set of attributes which are described in the following.

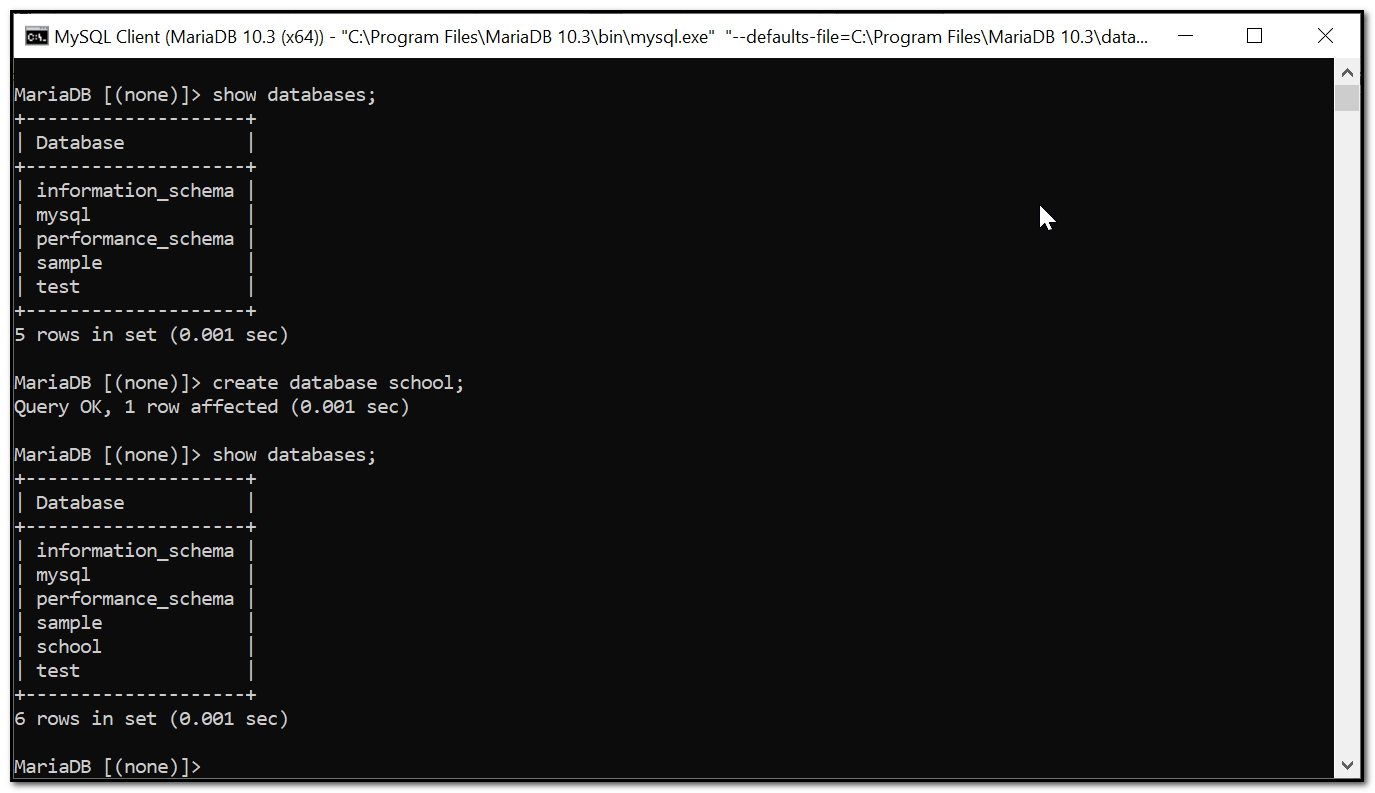
* The “Student” entity consists of attributes such as “StudentID, StudentName, DateOfBirth, IsFeePaid, ParentID.
* The “Parent” entity consists of attributes namely “ParentID”, “ParentName”, “DateOfBirth” and “PhoneNumber”.
* The “Event” entity consists of attributes such as “EventID, “Heldyear” and “StudentID”
* The entity “StudentSubject” consists of entities called “StudentID” and “SubjectID”
* The entity “Subject” consists of attributes called “SubjectID”, “SubjectName” and “Marksheet”.
* The “Transportation” entity consists of “TransportationID”,”FeesDetails, StudentID”

**Part A**

We have use the Vertabelo for generating the script of SQL commands for building the database and table structures.



The above image shows that the database has created and the name of the database is “School”.



This above provides information about the created database named as “school”. It shows that the query is OK and an row has affected.

**Create Tables**

**Table: Event**

CREATE TABLE Event (

EventID int NOT NULL,

HeldYear int NOT NULL,

StudentID int NOT NULL,

CONSTRAINT Event\_pk PRIMARY KEY (EventID)

);

**Table: Parent**

CREATE TABLE Parent (

ParentID int NOT NULL,

ParentName varchar(30) NOT NULL,

DateOfBirth date NOT NULL,

PhoneNumber int NOT NULL,

CONSTRAINT Parent\_pk PRIMARY KEY (ParentID)

);

**Table: Student**

CREATE TABLE Student (

StudentID int NOT NULL,

StudentName varchar(30) NOT NULL,

DateOfBirth date NOT NULL,

IsFeePaid varchar(10) NOT NULL,

ParentID int NOT NULL,

CONSTRAINT Student\_pk PRIMARY KEY (StudentID)

);

**Table: StudentSubject**

CREATE TABLE StudentSubject (

StudentID int NOT NULL,

SubjectID int NOT NULL

);

**Table: Subject**

CREATE TABLE Subject (

SubjectID int NOT NULL,

SubjectName varchar(40) NOT NULL,

Marksheet varchar(40) NOT NULL,

CONSTRAINT Subject\_pk PRIMARY KEY (SubjectID)

);

**Table: Transportation**

CREATE TABLE Transportation (

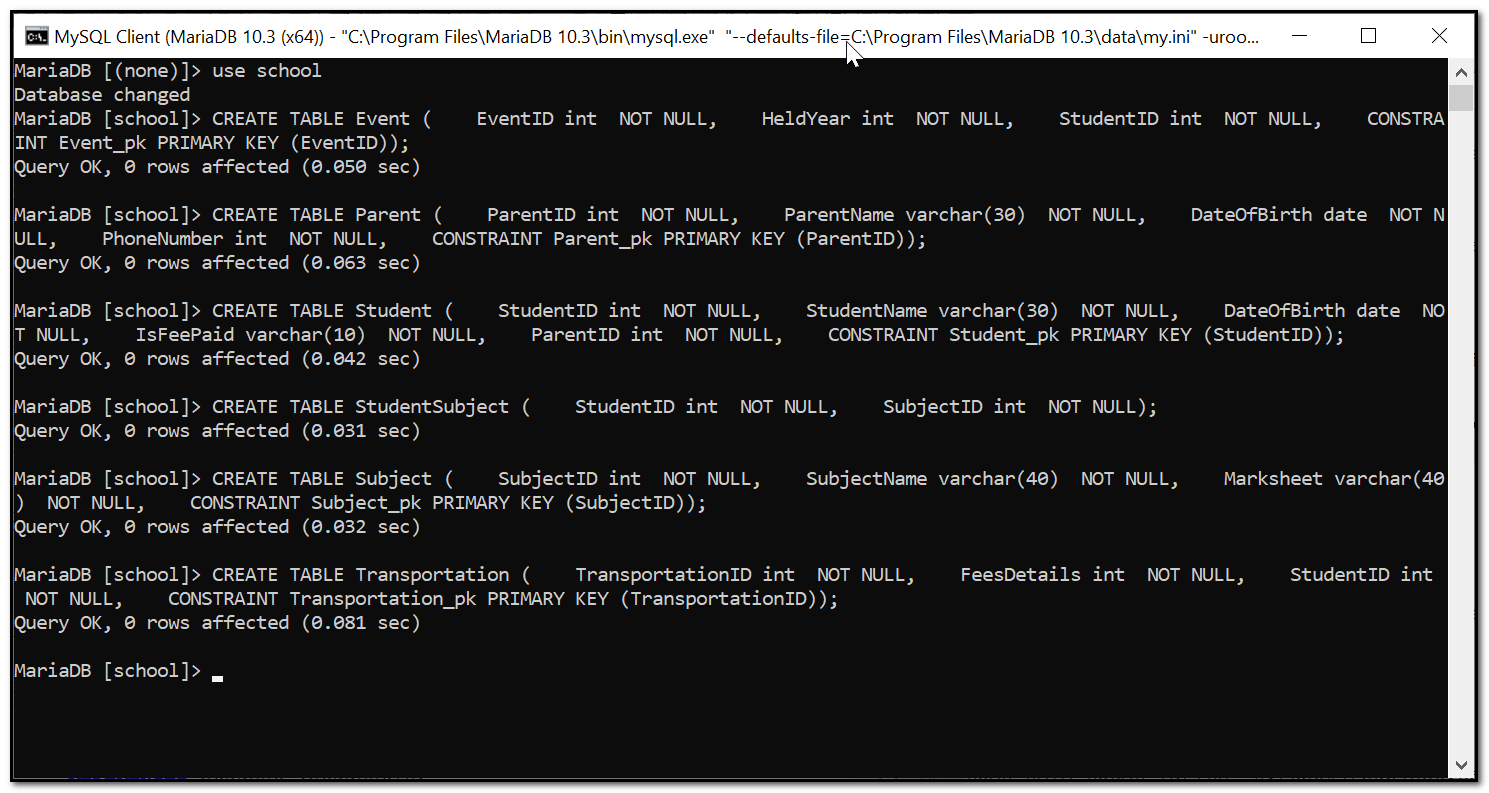
TransportationID int NOT NULL,

FeesDetails int NOT NULL,

StudentID int NOT NULL,

CONSTRAINT Transportation\_pk PRIMARY KEY (TransportationID)

);



The images provides the detailed information of code for creating tables. Also we have attached the code to create each table on the “school” database. The table is created for each entities along with their attributes.

**Foreign keys constraints**

ALTER TABLE Event ADD CONSTRAINT Event\_Student

FOREIGN KEY (StudentID)

REFERENCES Student (StudentID);

ALTER TABLE StudentSubject ADD CONSTRAINT StudentSubject\_Student

FOREIGN KEY (StudentID)

REFERENCES Student (StudentID) ;

ALTER TABLE StudentSubject ADD CONSTRAINT StudentSubject\_Subject

FOREIGN KEY (SubjectID)

REFERENCES Subject (SubjectID);

ALTER TABLE Student ADD CONSTRAINT Student\_Parent

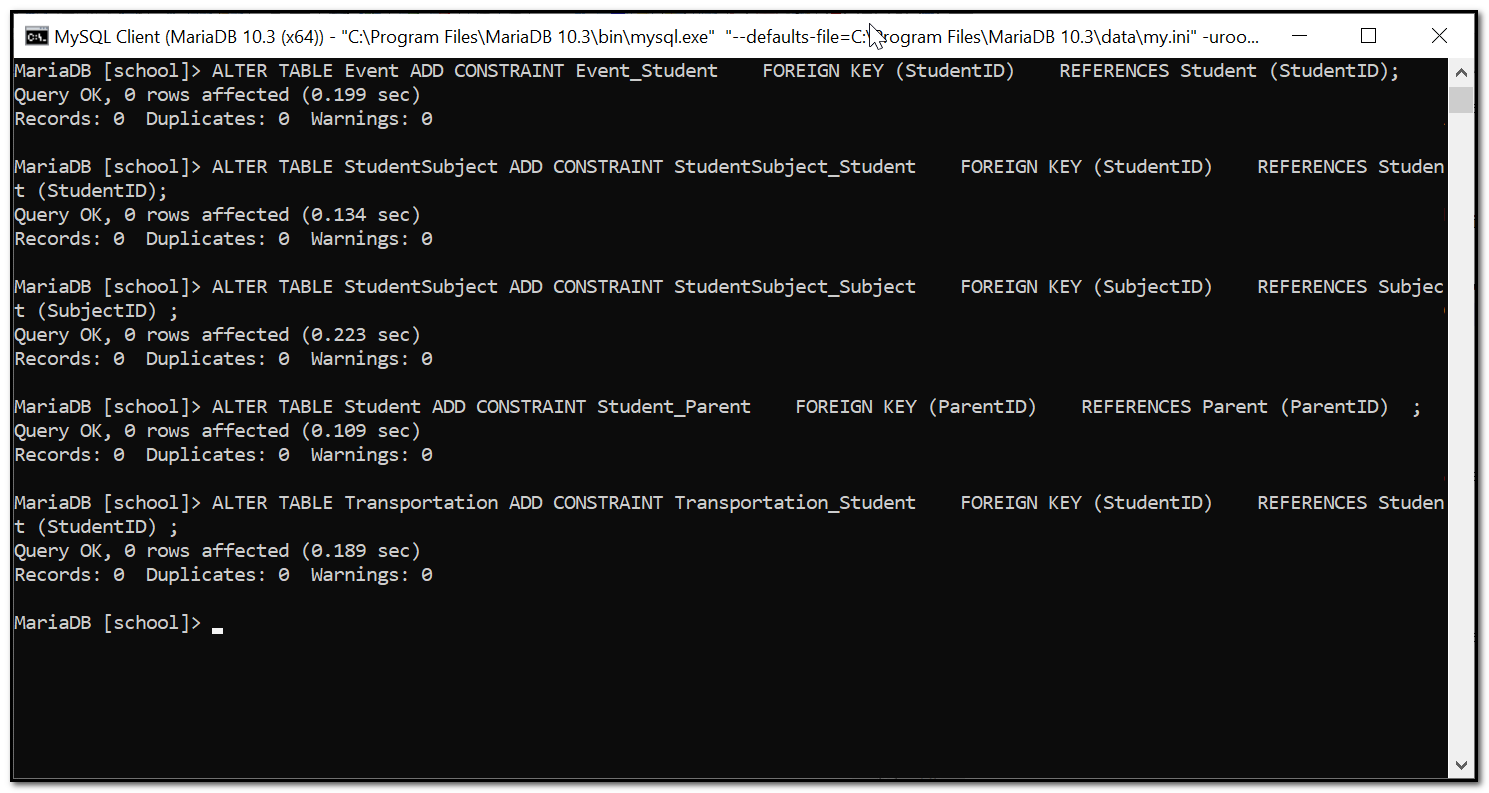
FOREIGN KEY (ParentID)

REFERENCES Parent (ParentID) ;

ALTER TABLE Transportation ADD CONSTRAINT Transportation\_Student

FOREIGN KEY (StudentID)

REFERENCES Student (StudentID) ;



This screen provide information about the foreign key and local key setting procedures on the SQL commands. Each entity foreign key details has been provided. The code has been attached and loaded in the database.

**Part B**

**Load your data files**

LOAD DATA LOCAL INFILE 'D:/CS509/Parent.csv'

INTO TABLE Parent

FIELDS TERMINATED BY','

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(ParentID,ParentName,DateOfBirth,PhoneNumber);

LOAD DATA LOCAL INFILE 'D:/CS509/Student.csv'

INTO TABLE Student

FIELDS TERMINATED BY','

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(StudentID,StudentName,DateOfBirth,IsFeePaid,ParentID);

LOAD DATA LOCAL INFILE 'D:/CS509/Event.csv'

INTO TABLE Event

FIELDS TERMINATED BY','

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(EventID,HeldYear,StudentID);

LOAD DATA LOCAL INFILE 'D:/CS509/Transportation.csv'

INTO TABLE Transportation

FIELDS TERMINATED BY','

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(TransportationID,FeesDetails,StudentID);

LOAD DATA LOCAL INFILE 'D:/CS509/Subject.csv'

INTO TABLE Subject

FIELDS TERMINATED BY','

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(SubjectID,SubjectName,MarkSheet);

LOAD DATA LOCAL INFILE 'D:/CS509/StudentSubject.csv'

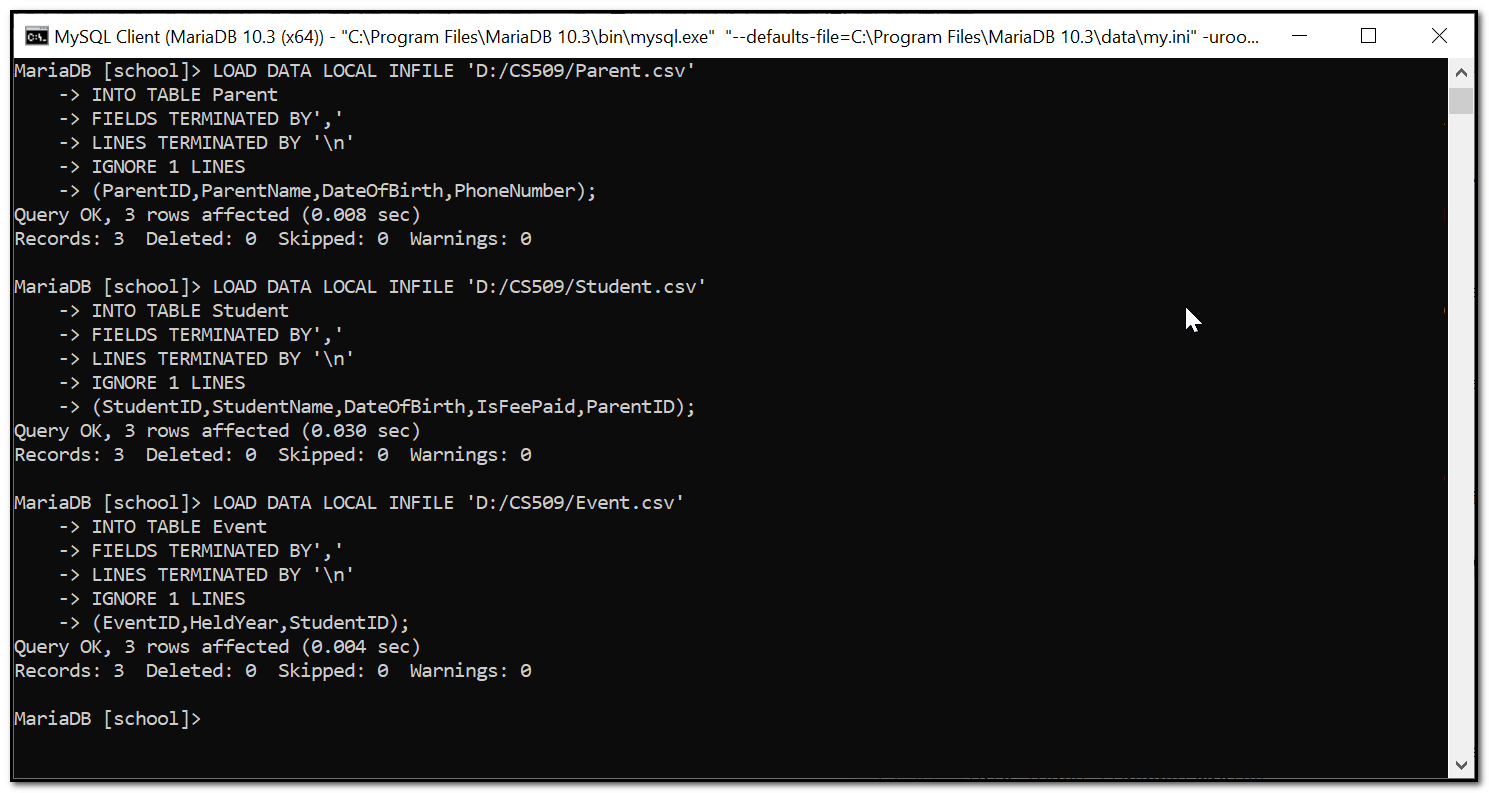
INTO TABLE StudentSubject

FIELDS TERMINATED BY','

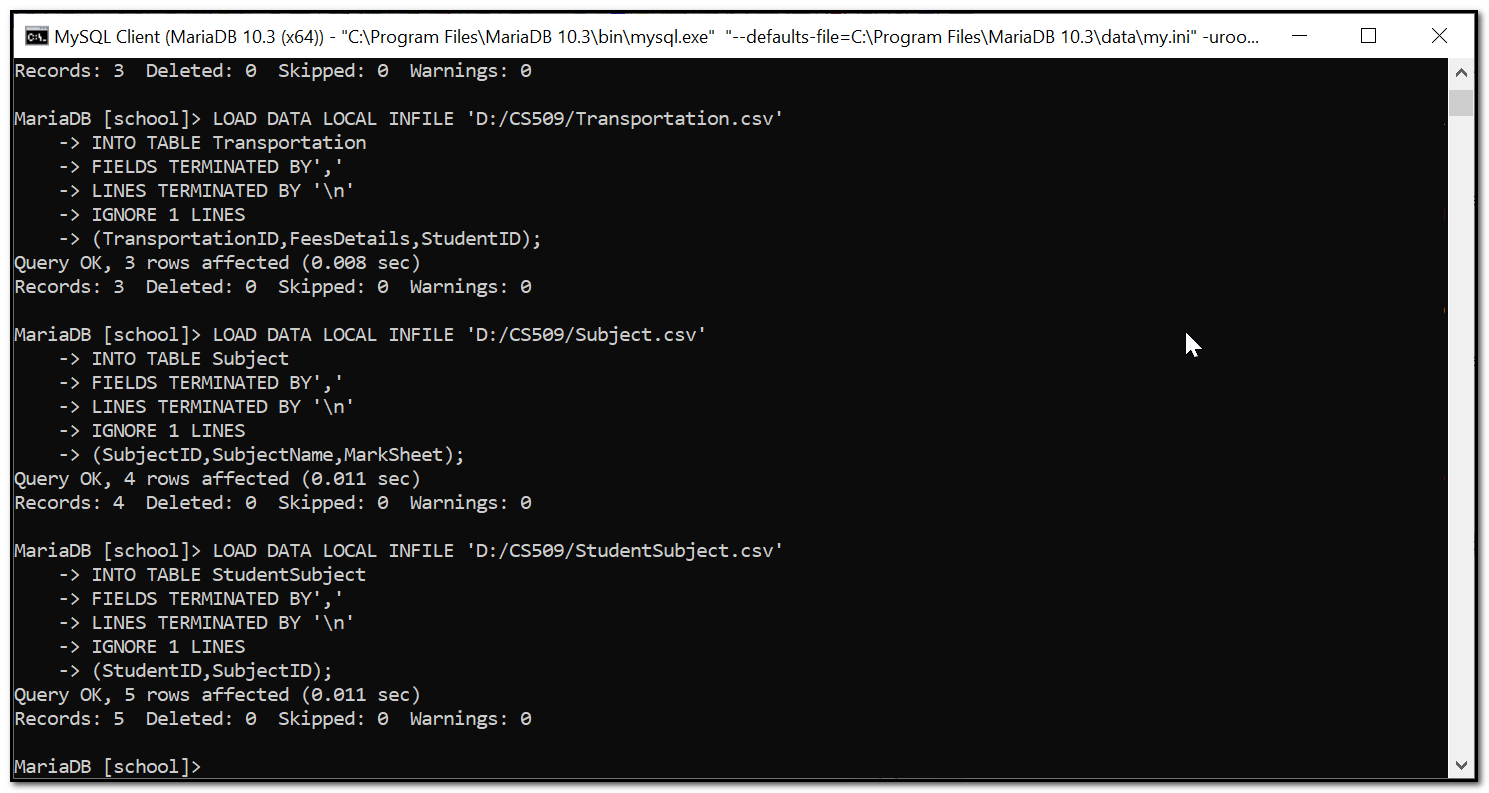
LINES TERMINATED BY '\n'

IGNORE 1 LINES

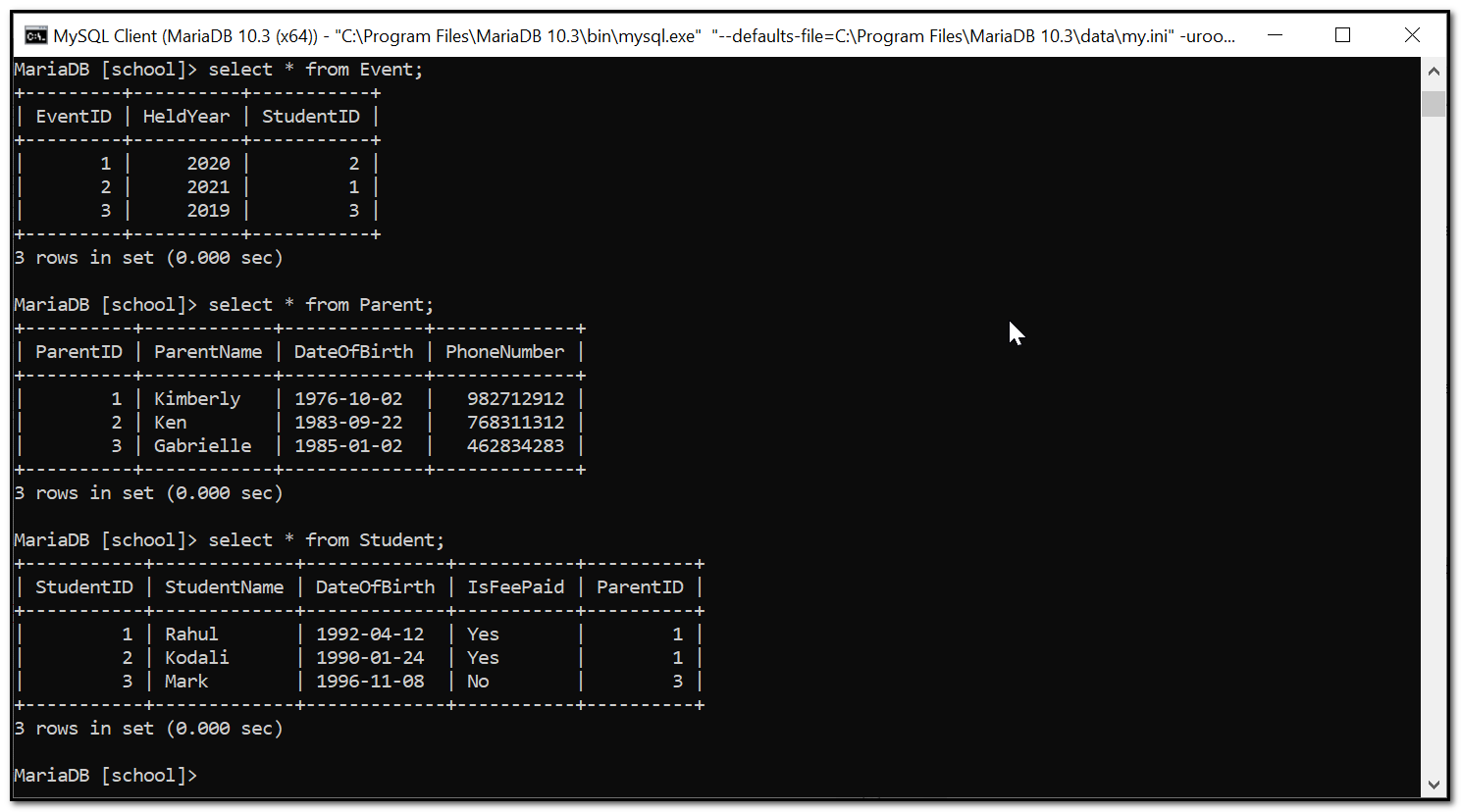
(StudentID,SubjectID);



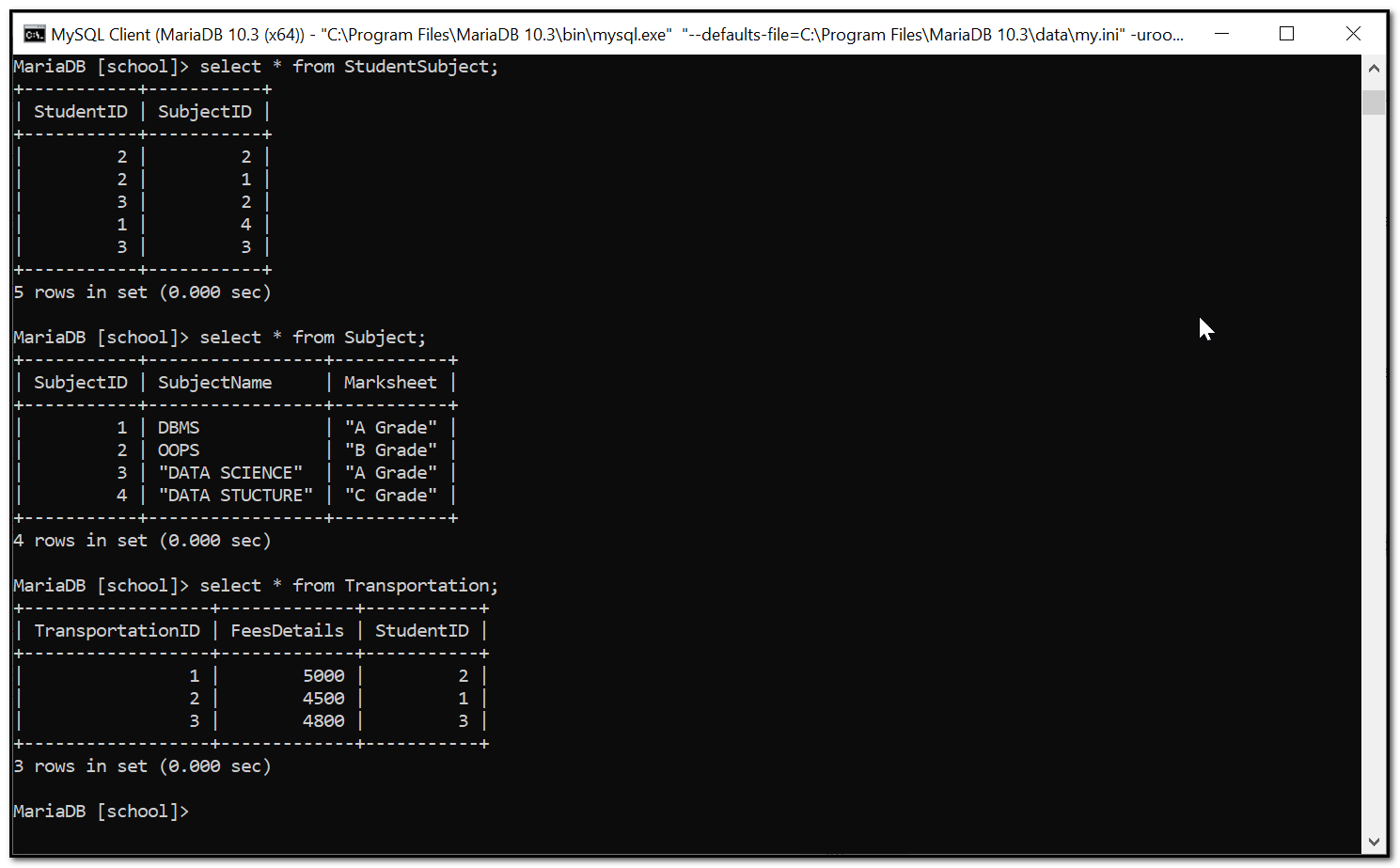
Now load the data files using the load command on “MariaDB” tool. Then uploading the .csv file to separate the delimited files



It shows that the recording of databases and their queries.



This displays first three tables of the databases which shows the Event, parent and student details along with the attributes.



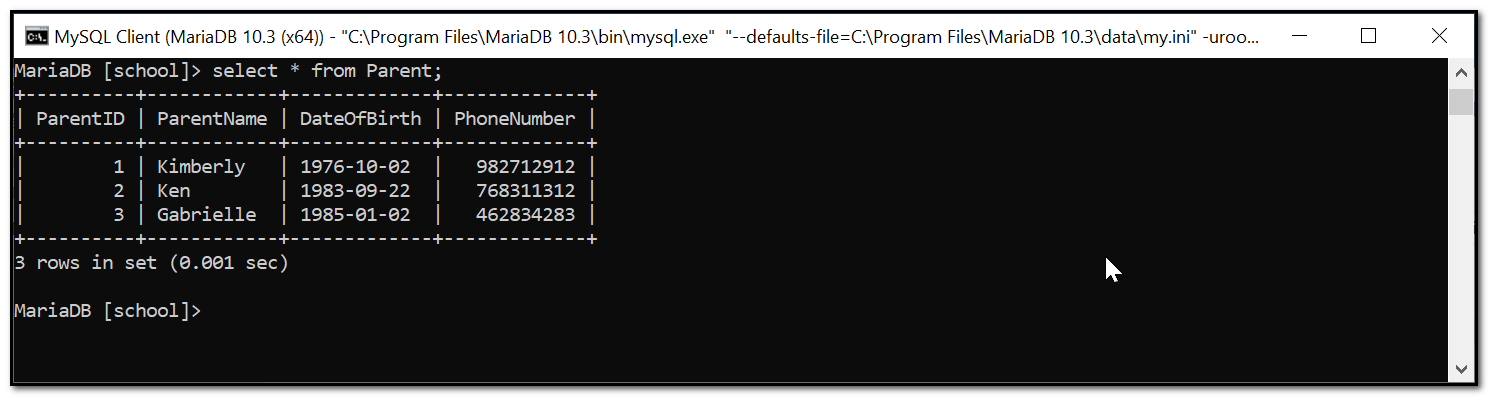
This provides information f the another three tables of database which are studentSubject, Subject and transportation.

**Data Manipulation Language Scripts**

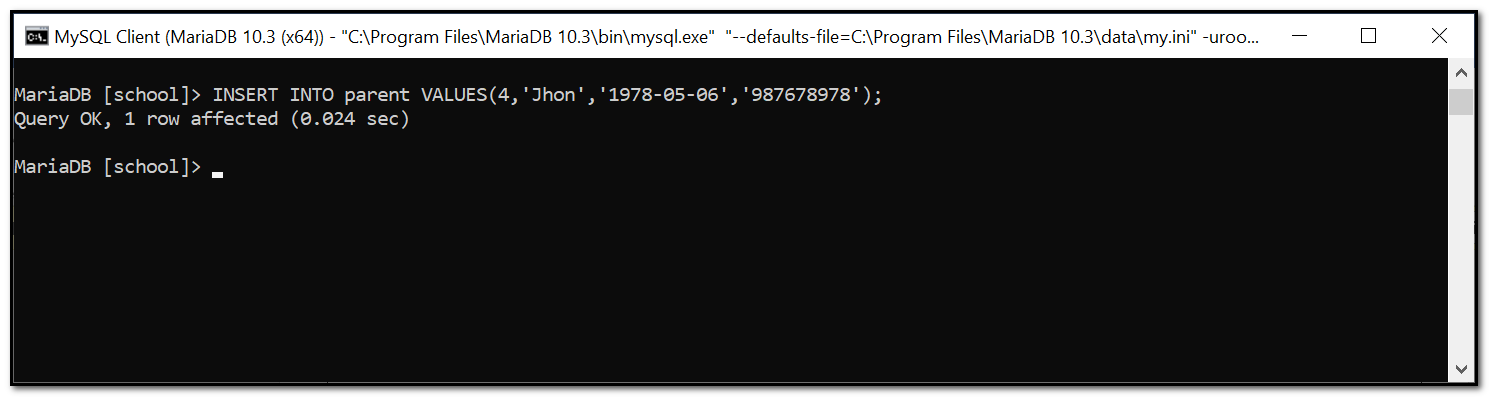
**Insert statements**

**Insert - 1**

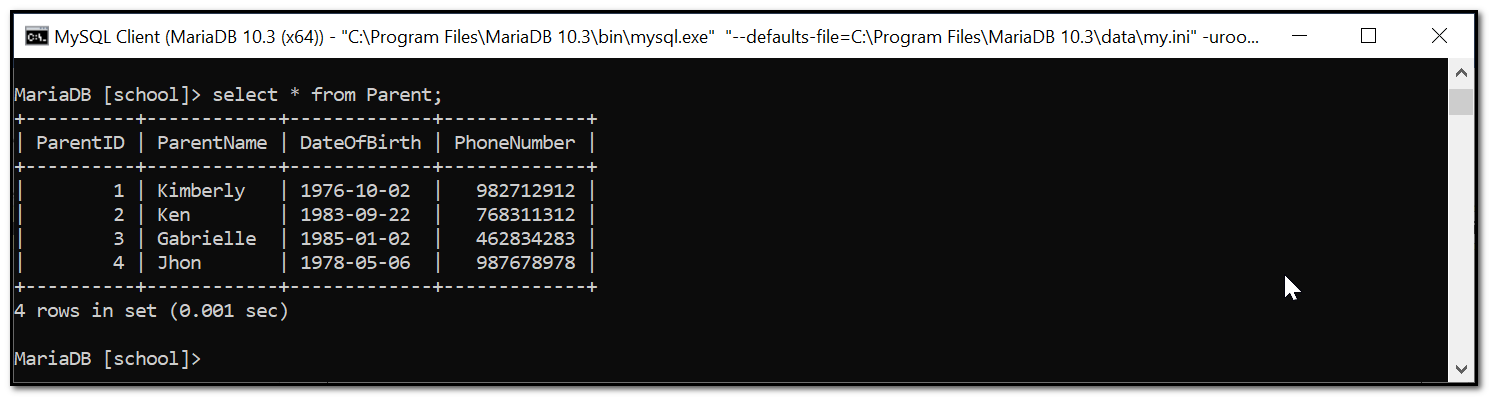
Before insert



INSERT INTO parent VALUES(4,'Jhon','1978-05-06','987678978');



After insert , the table will get all the data of parents along with details.

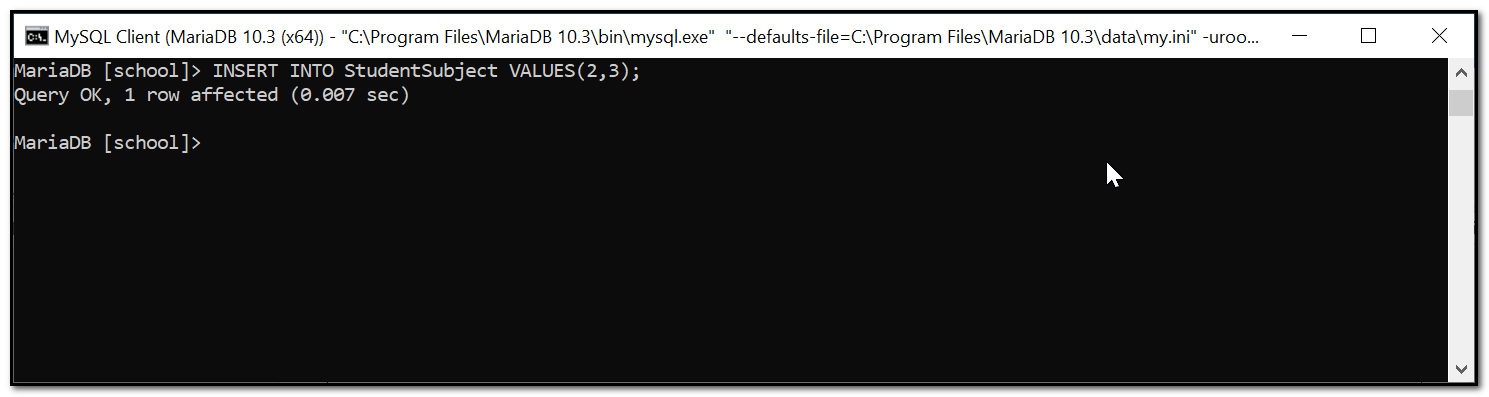


**Insert - 2**

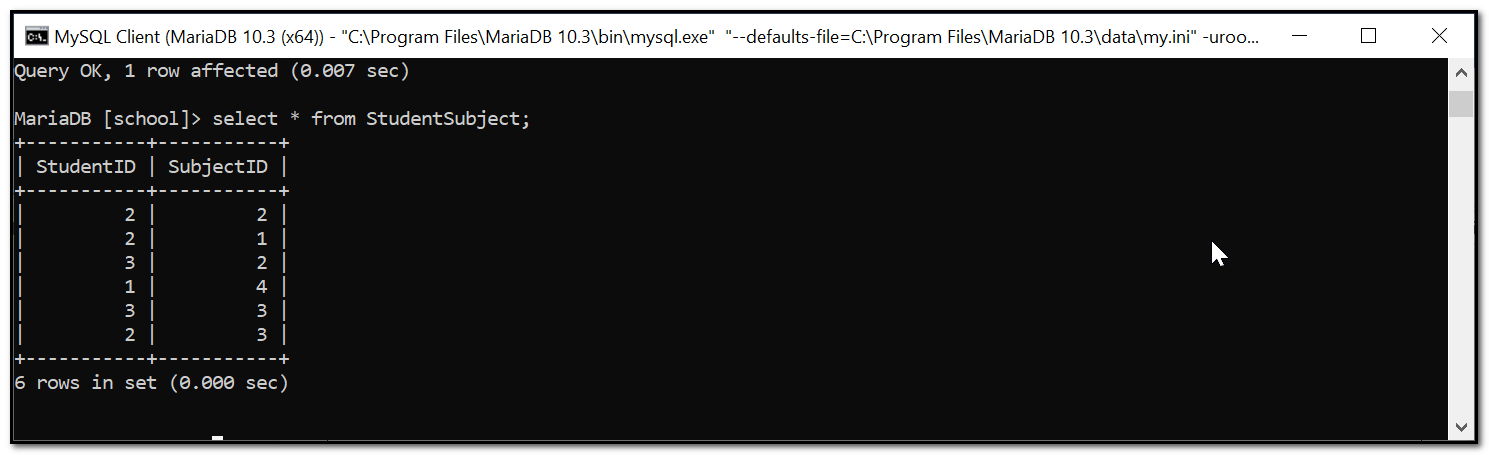
Before insert



INSERT INTO StudentSubject VALUES(2,3);



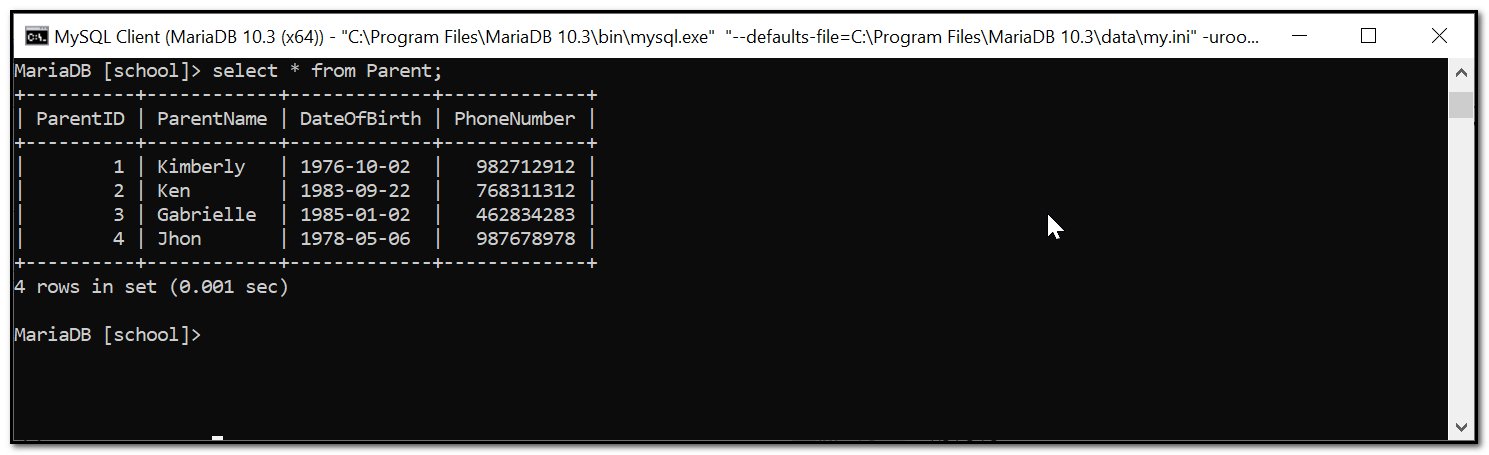
After insert, this will display all information about the subject code and subject ID.

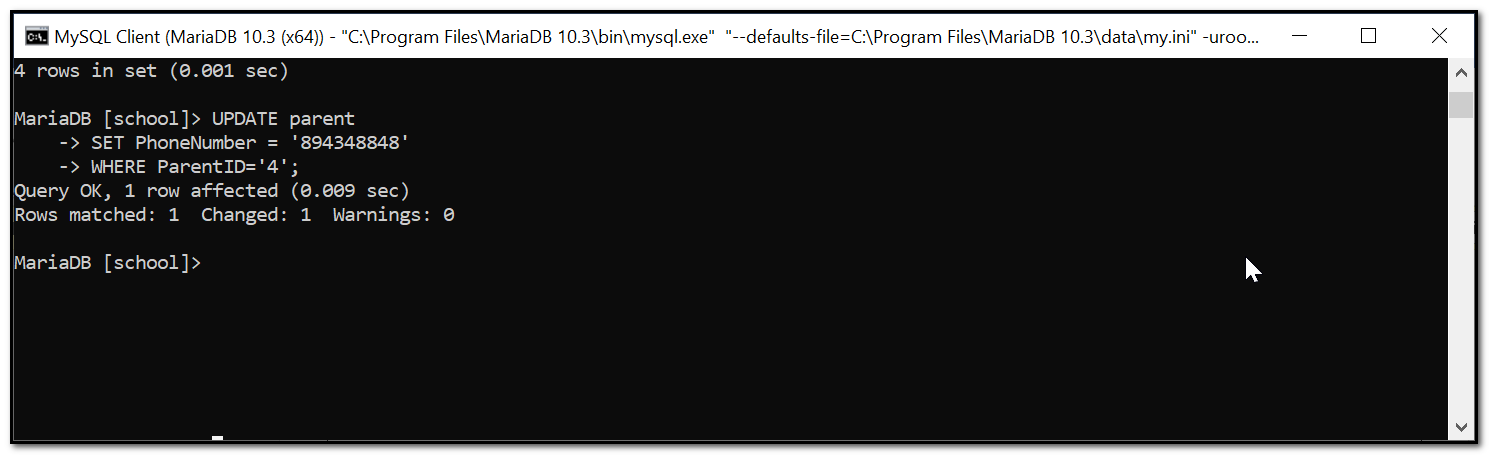


**Update statements**

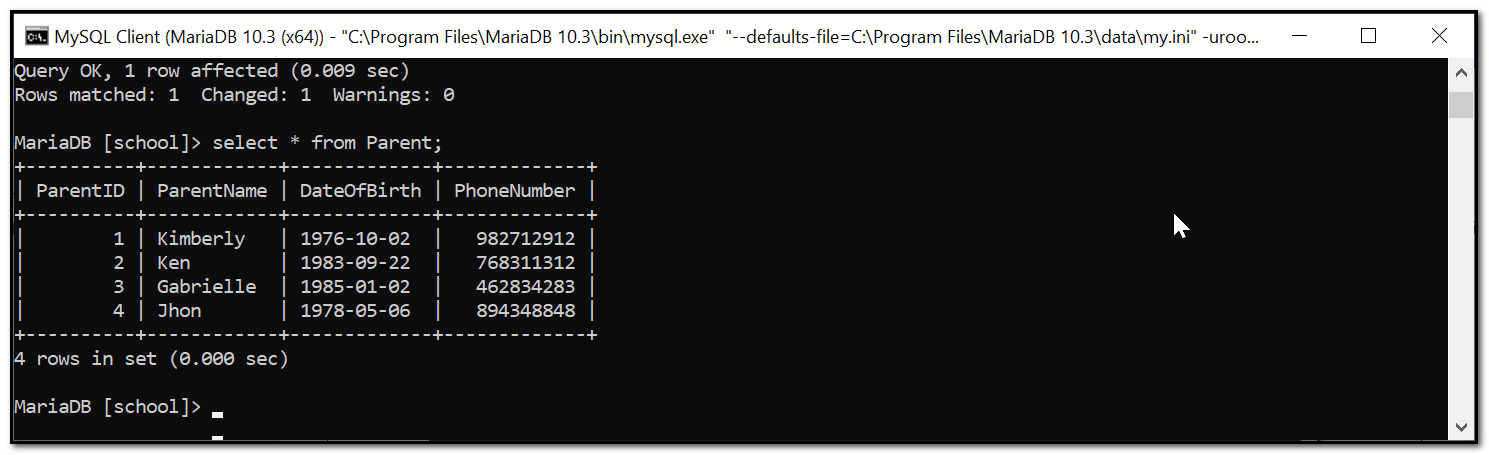
**Update - 1**

Before update



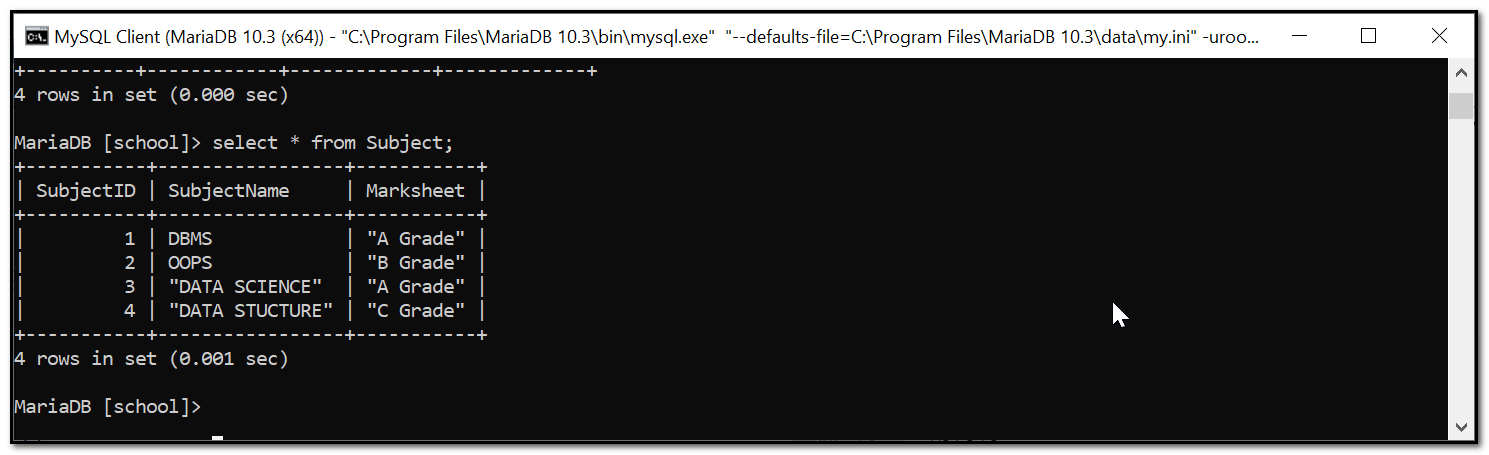


After update, the details are added to the table.



**Update - 2**

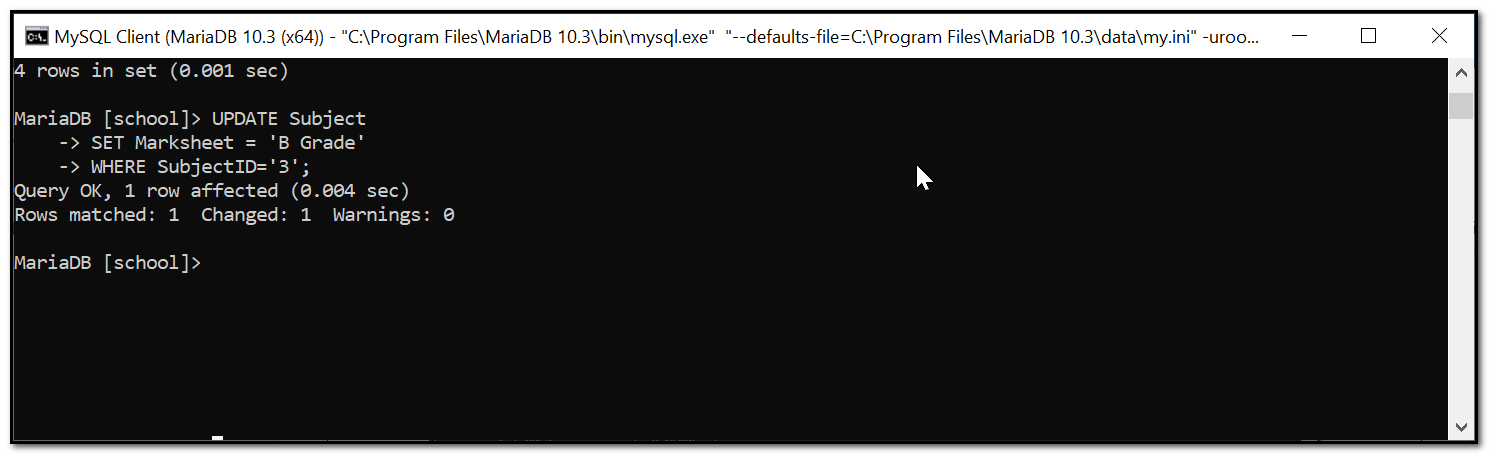
Before update



UPDATE Subject

SET Marksheet = 'B Grade'

WHERE SubjectID='3';

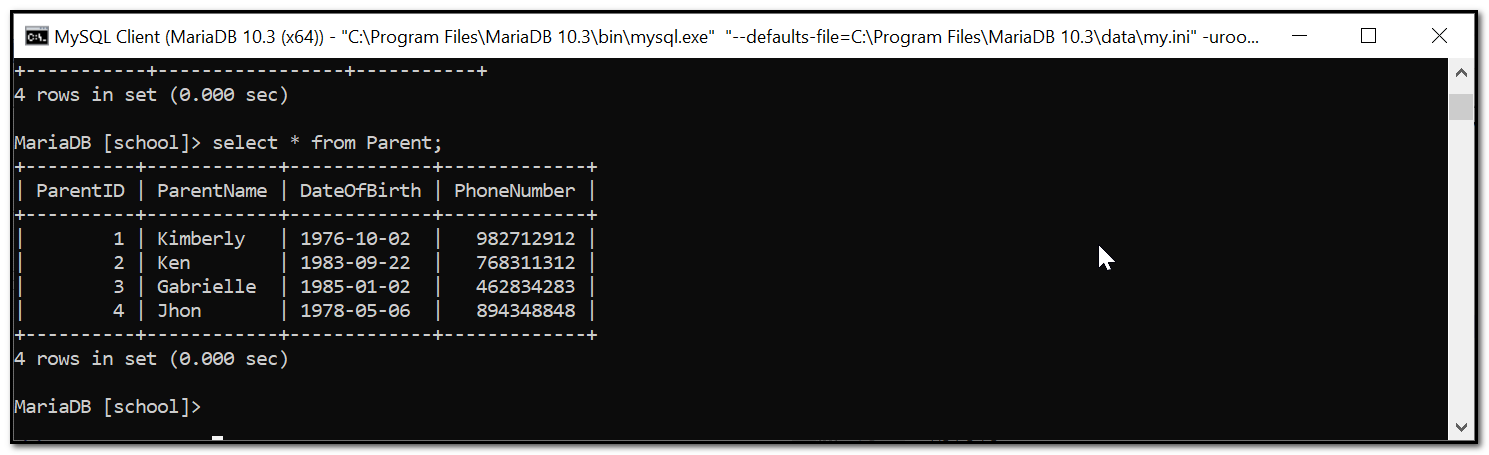


After update

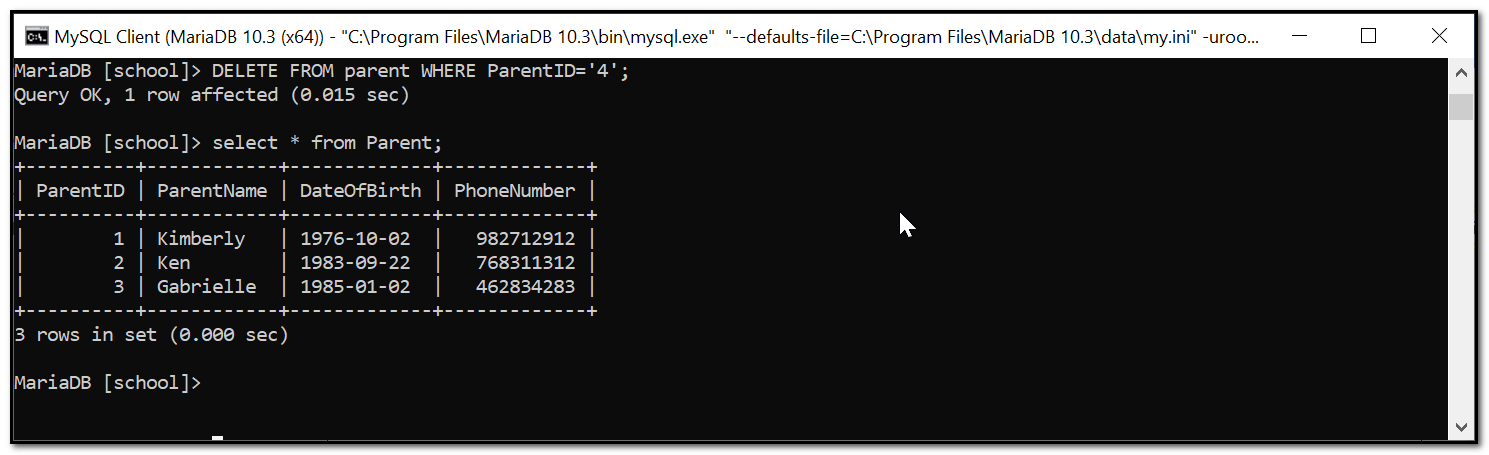


**Delete statement**

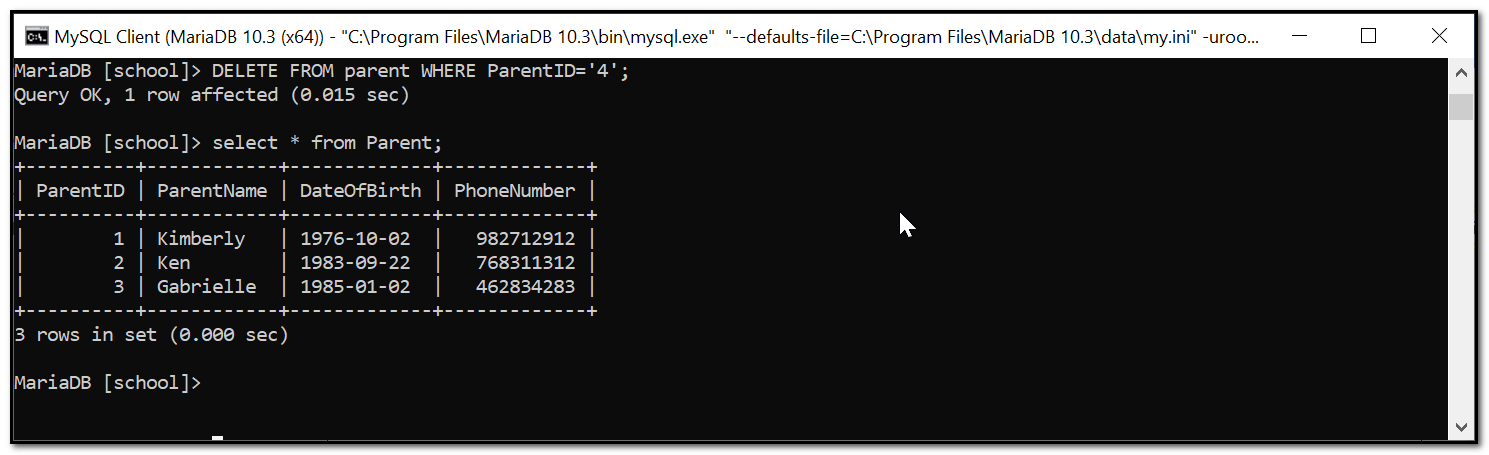
Before delete



DELETE FROM parent WHERE ParentID='4';

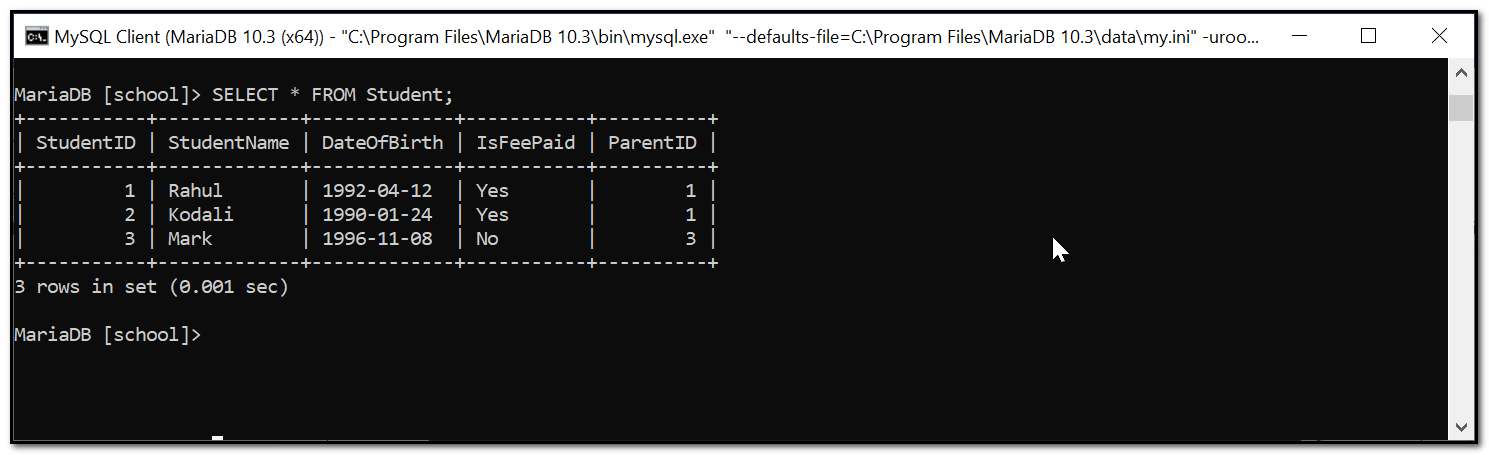


After delete



**Select statement**

SELECT \* FROM Student;

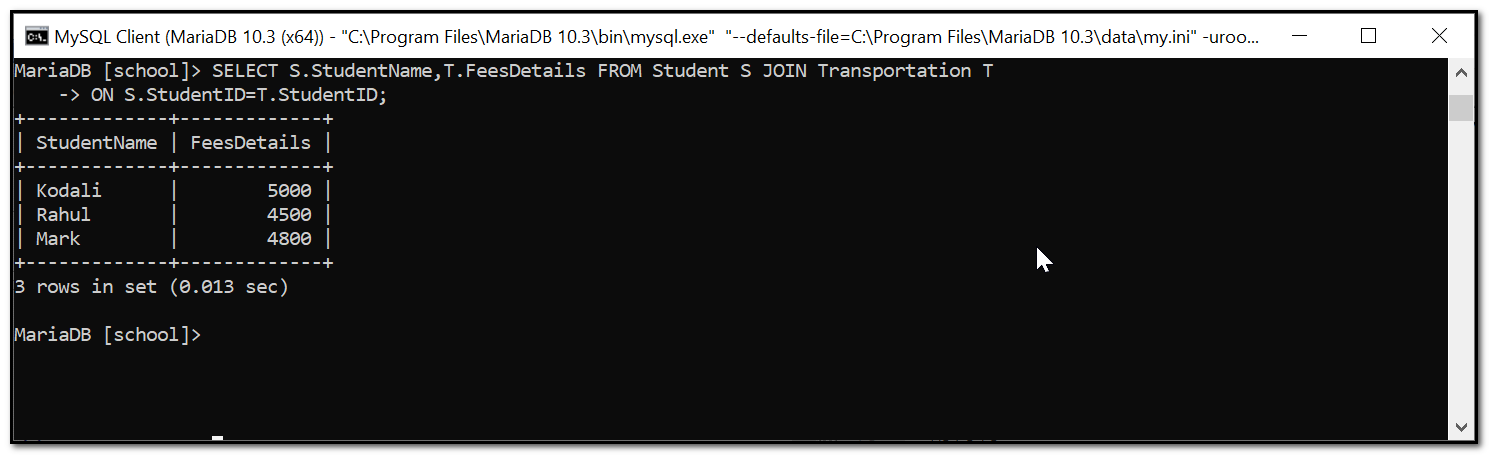


**Two join statements**

**Join 1**

SELECT S.StudentName,T.FeesDetails FROM Student S JOIN Transportation T

ON S.StudentID=T.StudentID

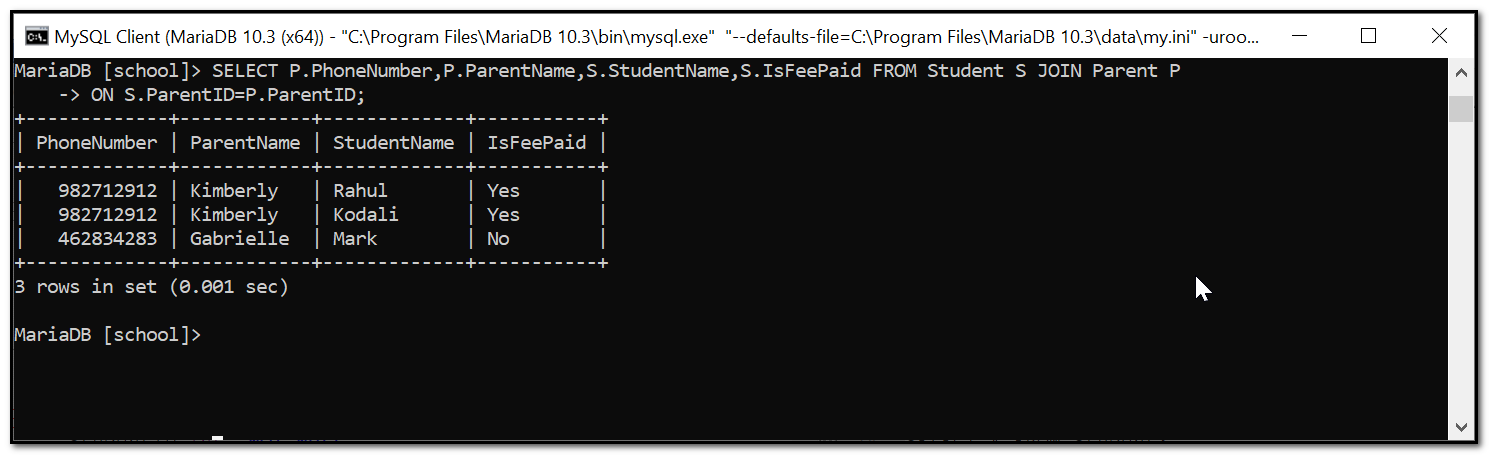


**Join 2**

SELECT P.PhoneNumber,P.ParentName,S.StudentName,S.IsFeePaid

FROM Student S JOIN Parent P

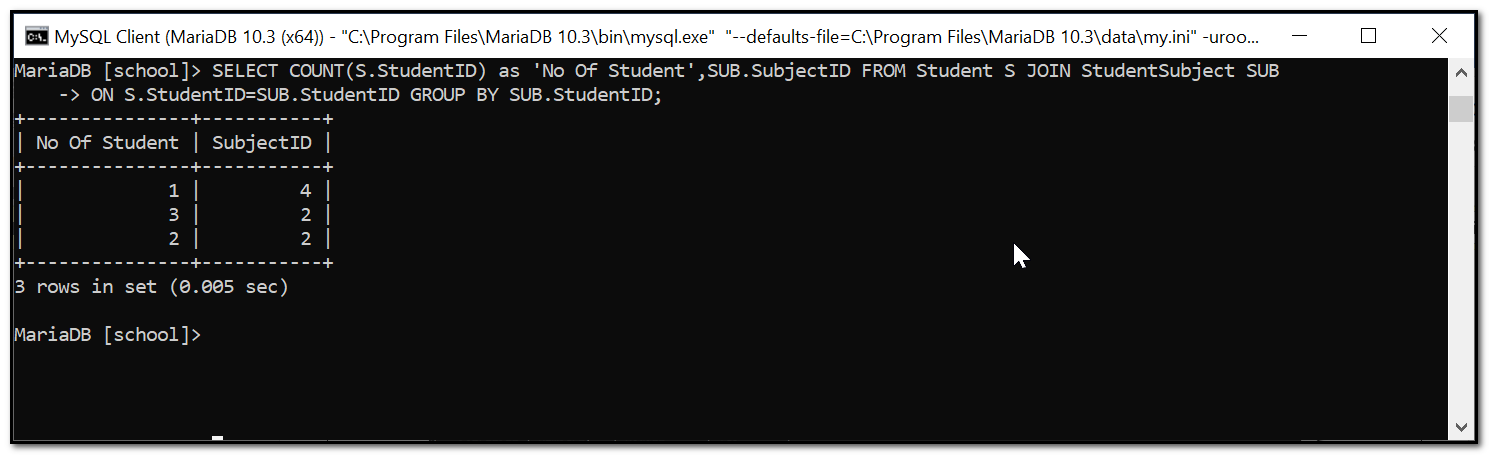
ON S.ParentID=P.ParentID;



**Summary statements**

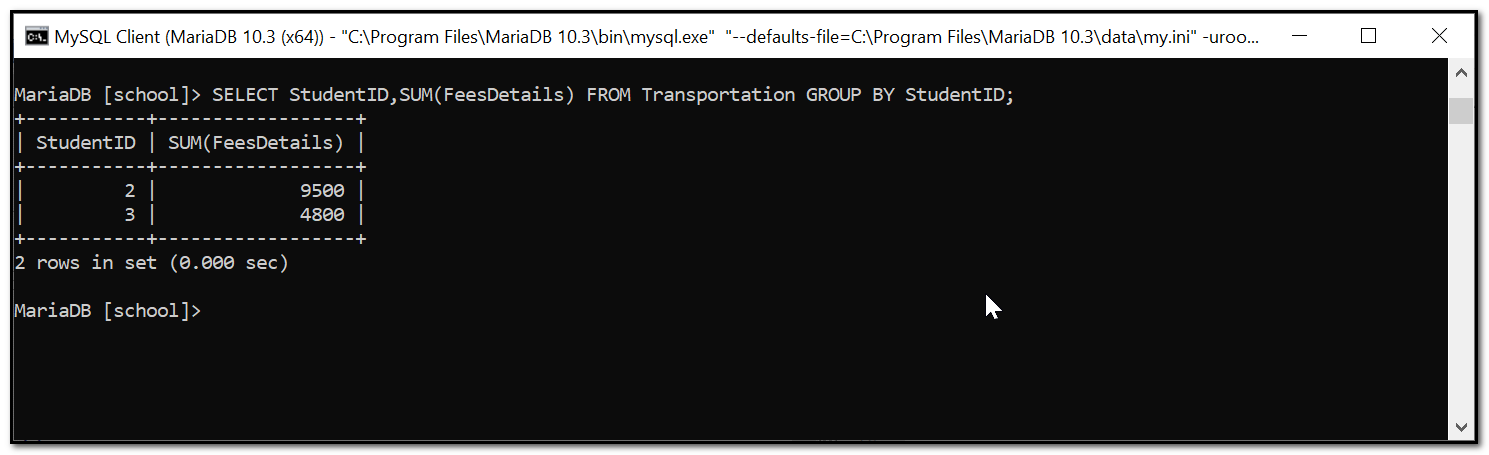
**Summary 1**

SELECT COUNT(S.StudentID) as 'No Of Student',SUB.SubjectID FROM Student S JOIN StudentSubject SUB ON S.StudentID=SUB.StudentID GROUP BY SUB.StudentID;



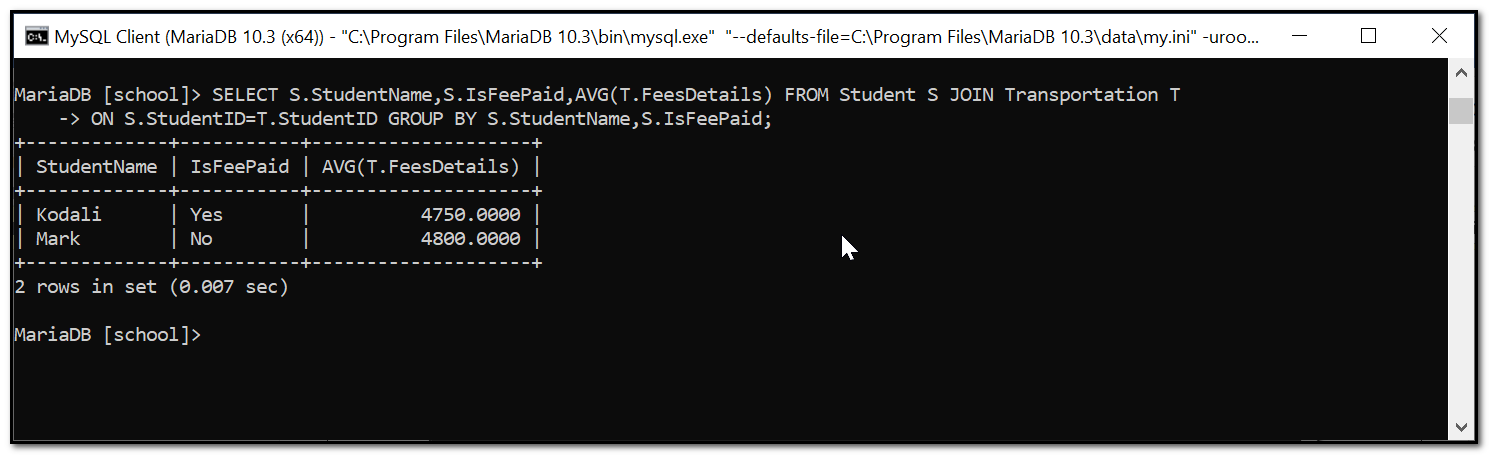
**Summary 2**

SELECT StudentID,SUM(FeesDetails) FROM Transportation GROUP BY StudentID



**Summary 3**

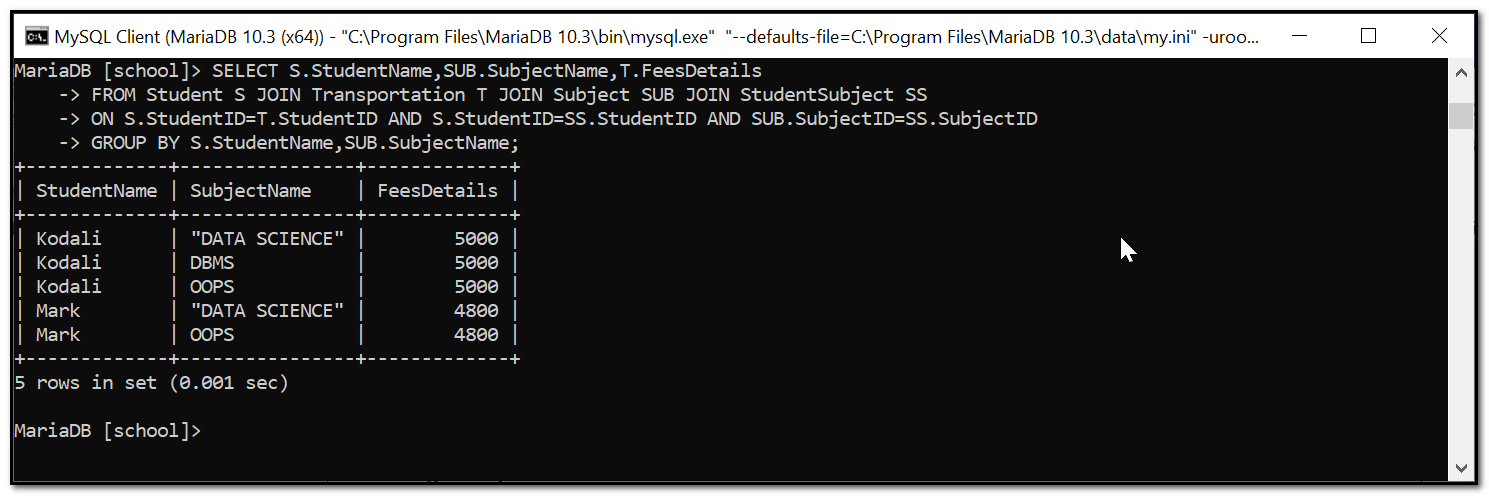
SELECT S.StudentName,S.IsFeePaid,AVG(T.FeesDetails) FROM Student S JOIN Transportation T ON S.StudentID=T.StudentID GROUP BY S.StudentName,S.IsFeePaid



**Multi-table query**

**Query 1**

SELECT S.StudentName,SUB.SubjectName,T.FeesDetails FROM Student S JOIN Transportation T JOIN Subject SUB JOIN StudentSubject SS ON S.StudentID=T.StudentID AND S.StudentID=SS.StudentID AND SUB.SubjectID=SS.SubjectID GROUP BY S.StudentName,SUB.SubjectName;



This now all the 12 query has been executed.