PATAN MULTIPLE CAMPUS
PATANDHOKA, LALITPUR

A LAB REPORT ON INTRODUCTORY DATABASE USING STANDARD QUERYING LANGUAGE

Submitted By
Yadu Sharma
BBA Second Semester
Section ‘B’
Exam Roll No: 28667/20

Submitted To
Binit Shrestha
(Faculty Member)

(Submitted As A Partial Requirement For The Competition Of Introductory Database, Studied By The Students Of BBA Second Semester Student, Tribhuwan University)

February 20, 2022

Table of Contents

CHAPTER-1: SQL BRIEF DESCRIPTION

Structured Query Language(SQL) Is An American National Standard Institute(ANSI) Standard Programming Languages Which Is Designed Specifically For Sorting And Managing The Data In The Relational Database Management System(RDBMS) Using All Kinds Of Data Operations. SQL Is Used To Create, Remove, Alter Database And Database Objects In A Database Management System And Store, Retrieve, Update The Data In A Database .SQL Is A Standard Language For Creating, Accessing, And Manipulating Database Management System. SQL Works For All Modern Relational Database Management System Like, SQL Server, Oracle, and MySQL.

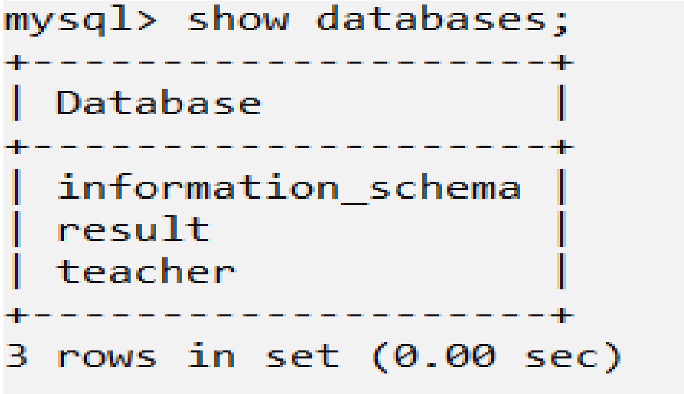
1.1 Introductory Commands In SQL

1.1.1 Show Database

This Command Is Used To Show The List At Database.

Syntax:

```
Show databases;
```



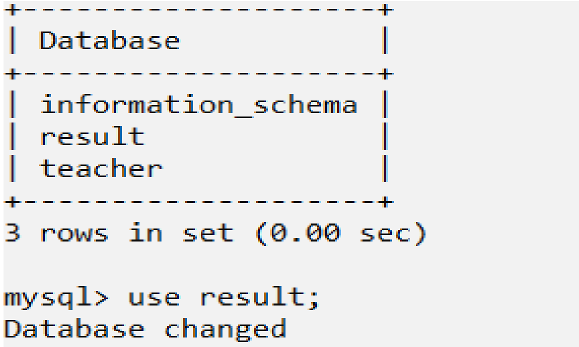
```
mysql> show databases;
+----------------------+
| Database             |
+----------------------+
| information_schema |
| result               |
| teacher              |
+----------------------+
3 rows in set (0.00 sec)
```

1.1.2 Use Database

This Command Is Used To Enter Into A Database Or A Particular Database.

Syntax:

```
Use database_name;
```



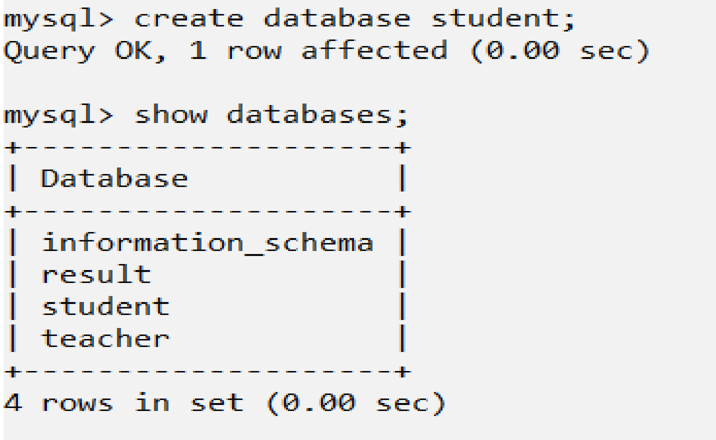
```
+--------------------+
| Database           |
+--------------------+
| information_schema |
| result             |
| teacher            |
+--------------------+
3 rows in set (0.00 sec)

mysql> use result;
Database changed
```

1.1.3 Create Database

This Command Is Used To Create A Database.

Syntax:

```
Create database database_name;
```



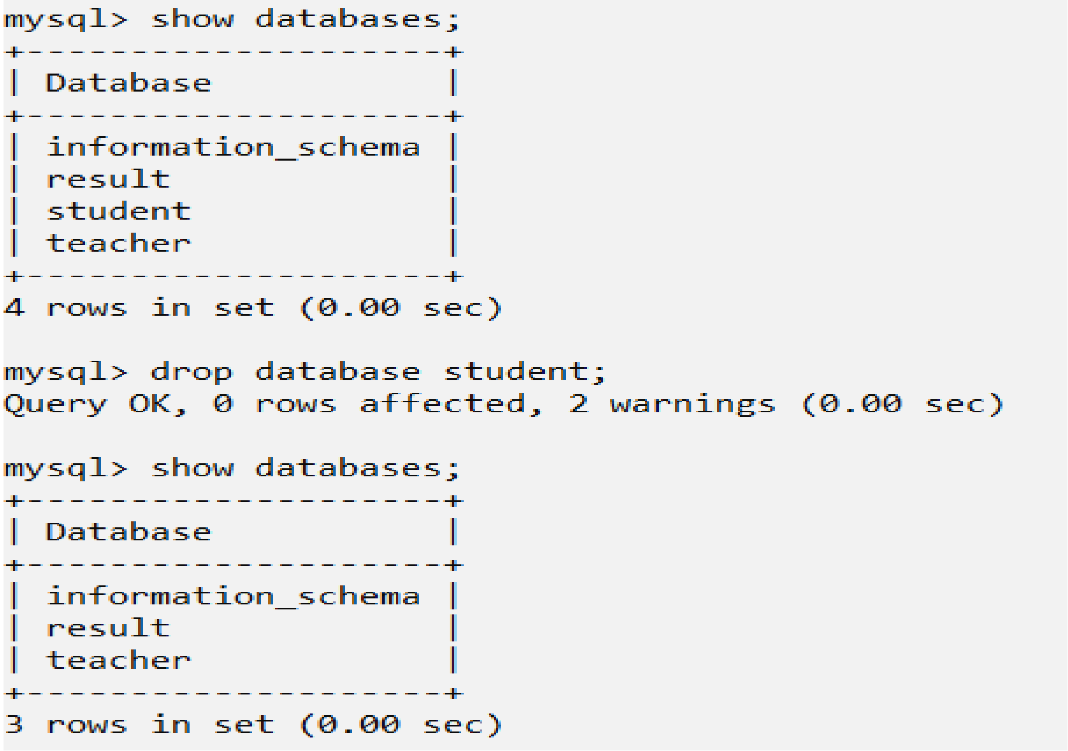
```
mysql> create database student;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| result             |
| student            |
| teacher            |
+--------------------+
4 rows in set (0.00 sec)
```

1.1.4 Drop Table

This Command Is Use To Delete A Particular Database.

Syntax:

```
Drop database database_name;
```



```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| result             |
| student            |
| teacher            |
+--------------------+
4 rows in set (0.00 sec)

mysql> drop database student;
Query OK, 0 rows affected, 2 warnings (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| result             |
| teacher            |
+--------------------+
3 rows in set (0.00 sec)
```

CHAPTER-2: DATA DEFINITION LANGUAGE (DDL)

Data Definition Is A Subset Of SQL Statements That Change The Structure Of The Database Schema In Some Way, Typically By Creating, Deleting, Or Modifying Schema Objects Such As Databases, Tables, And Views. The DDL Statements Are Listed Below:

a) Create Statement

b) Drop Statement

c) Alter Statement

d) Truncate Statement

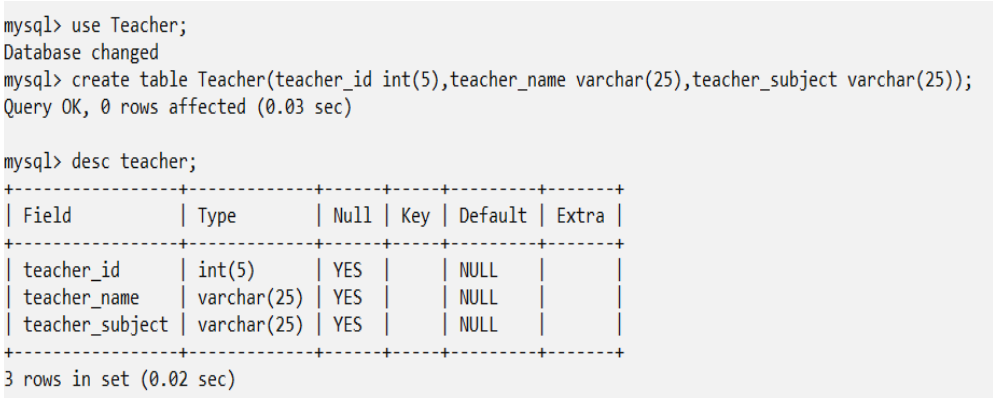
CHAPTER-3: DATA DEFINITION LANGUAGE (DDL) COMMANDS

3.1 Create Table

This Command Is Used To Create A Blank Tables ,Views Or Other Database Object.

Syntax:

```
Create table table_name (column1 datatype(length),column2
datatype(length)…….column3   datatype(length));
```



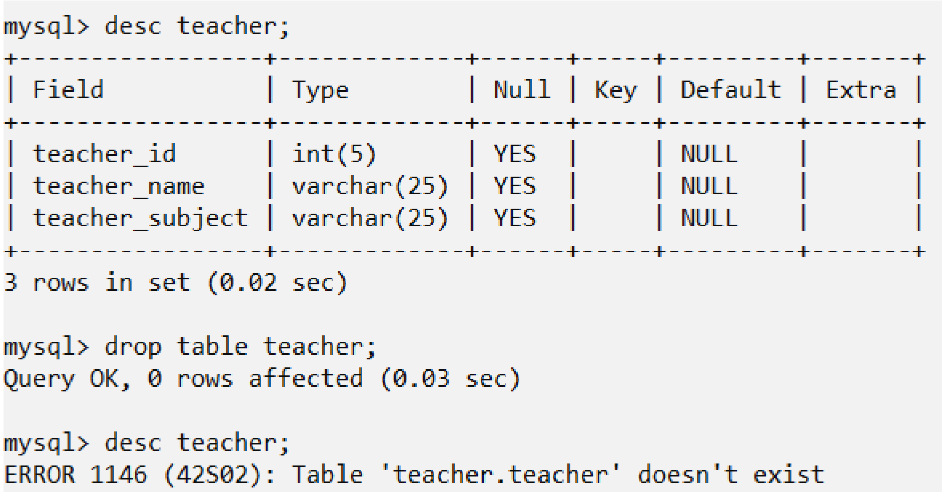
```
mysql> use Teacher;
Database changed
mysql> create table Teacher(teacher_id int(5),teacher_name varchar(25),teacher_subject varchar(25));
Query OK, 0 rows affected (0.03 sec)

mysql> desc teacher;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| teacher_id      | int(5)      | YES  |     | NULL    |       |
| teacher_name    | varchar(25) | YES  |     | NULL    |       |
| teacher_subject | varchar(25) | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
3 rows in set (0.02 sec)
```

3.2 Drop Table

This Command Is Used To Delete Or Remove An Existing Table, Views Or Other Databases Object.

Syntax:

```
Drop table table_name;
```



```
mysql> desc teacher;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| teacher_id      | int(5)      | YES  |     | NULL    |       |
| teacher_name    | varchar(25) | YES  |     | NULL    |       |
| teacher_subject | varchar(25) | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
3 rows in set (0.02 sec)

mysql> drop table teacher;
Query OK, 0 rows affected (0.03 sec)

mysql> desc teacher;
ERROR 1146 (42S02): Table 'teacher.teacher' doesn't exist
```

3.3 Alter Table

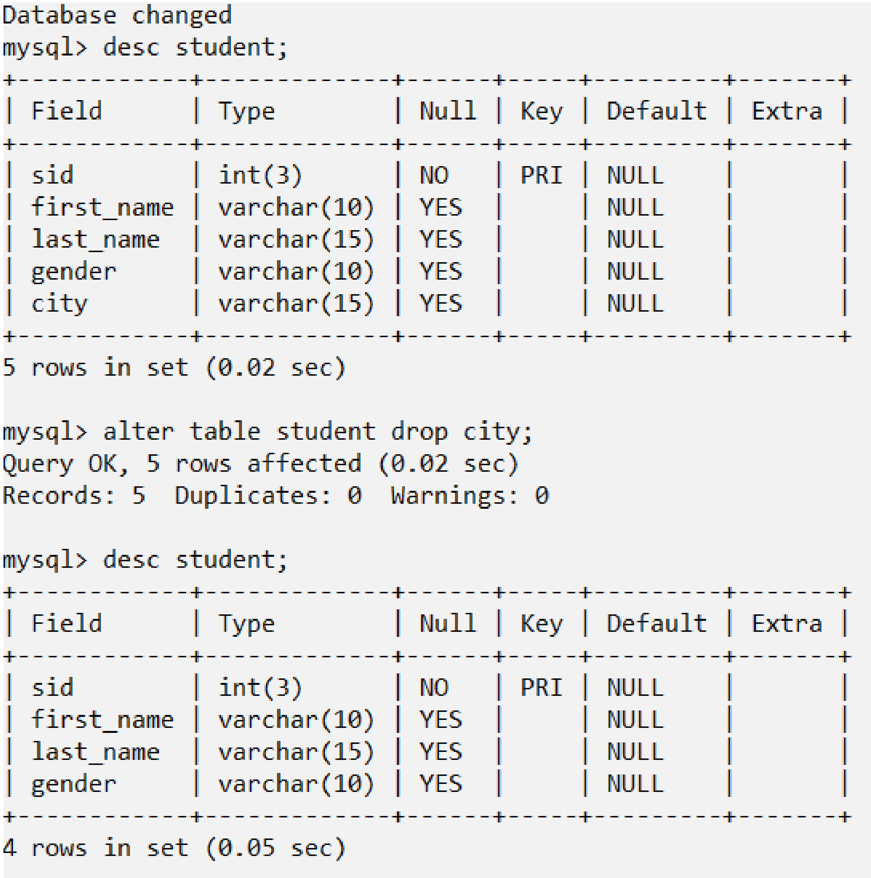
This Command Is Used To Modify A Given Table.

3.3.1 Drop Column

This Command Is Used To Delete A Column In A Given Table.

Syntax:

```
Alter table table_name drop column_name;
```



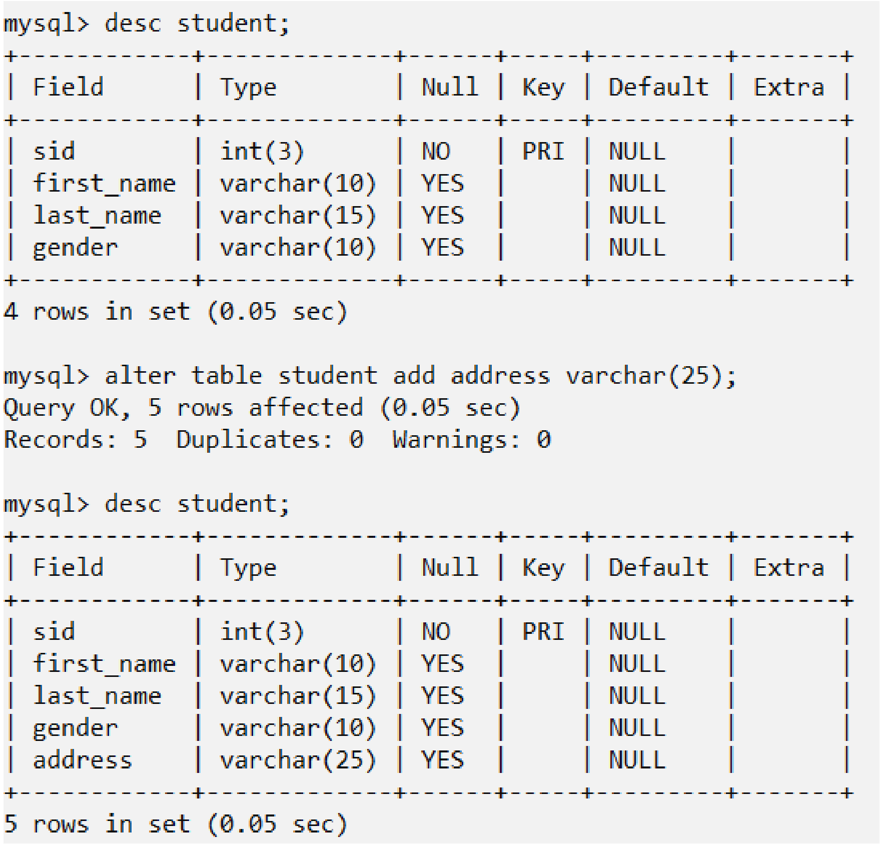
```
Database changed
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| city       | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.02 sec)

mysql> alter table student drop city;
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)
```

3.3.2 Add Column

This Command Is Used To Add Column In A Given Table.

Syntax:

```
Alter table  table_name add column_name datatype (length);
```



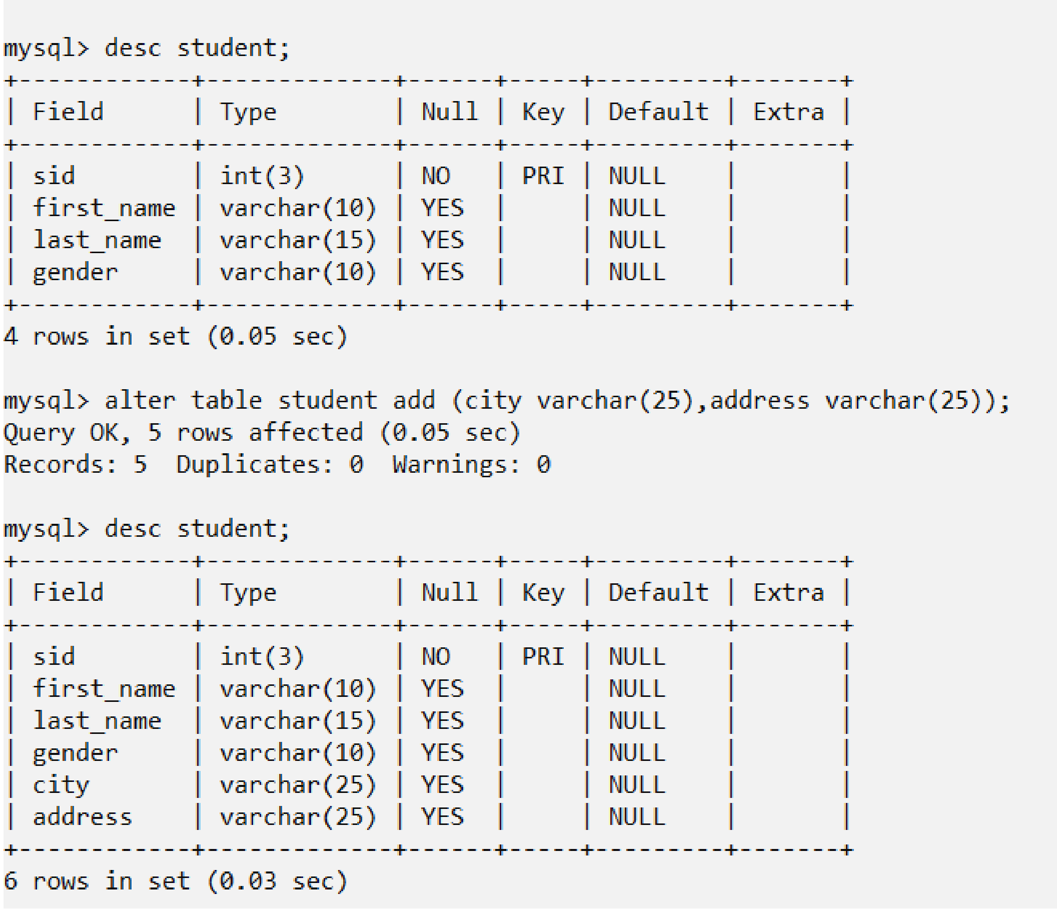
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)

mysql> alter table student add address varchar(25);
Query OK, 5 rows affected (0.05 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(25) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.05 sec)
```

3.3.3 Add Multiple Column

This Command Is Used To Add Multiple Columns In A Given Table At Once.

Syntax:

```
Alter table table _name add(column_name1
datatype(length),column_name2 datatype(length);
```



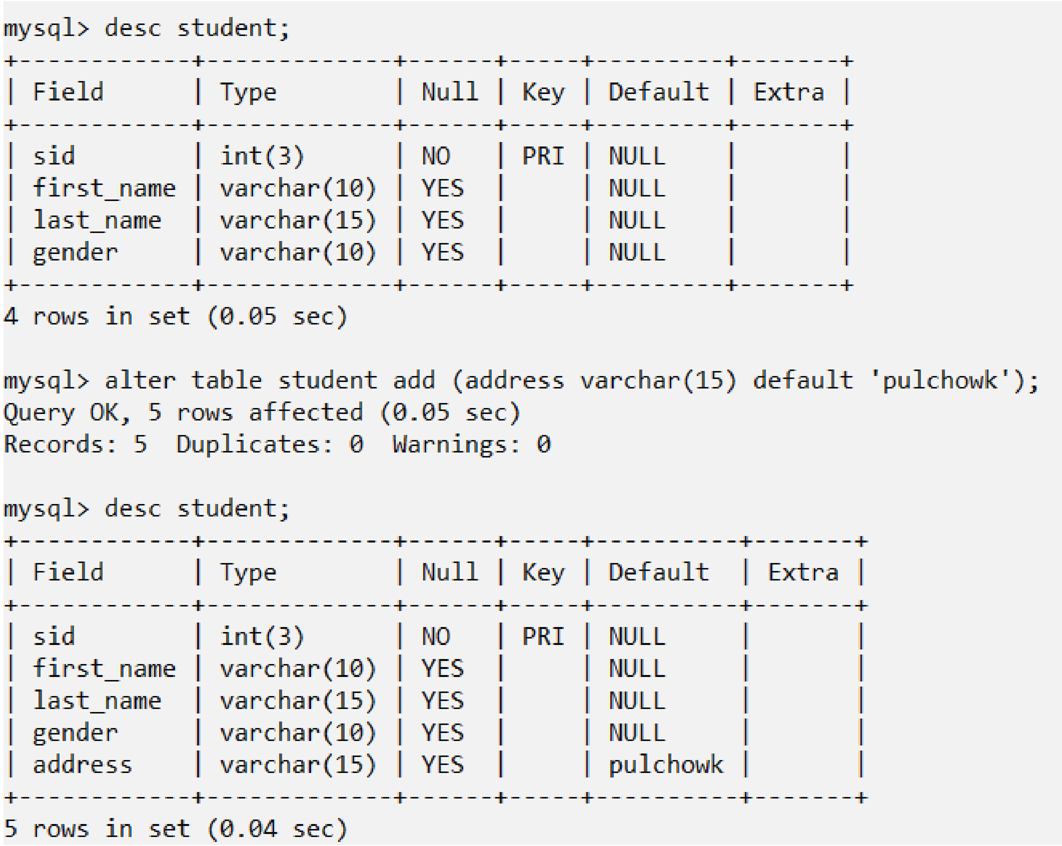
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)

mysql> alter table student add (city varchar(25),address varchar(25));
Query OK, 5 rows affected (0.05 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| city       | varchar(25) | YES  |     | NULL    |       |
| address    | varchar(25) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
6 rows in set (0.03 sec)
```

3.3.4 Add Column With Default Value

This Command Is Used To Add Value In Default Column.

Syntax:

```
Alter table table_name add (column_name datatype(length)
default default_value);
```



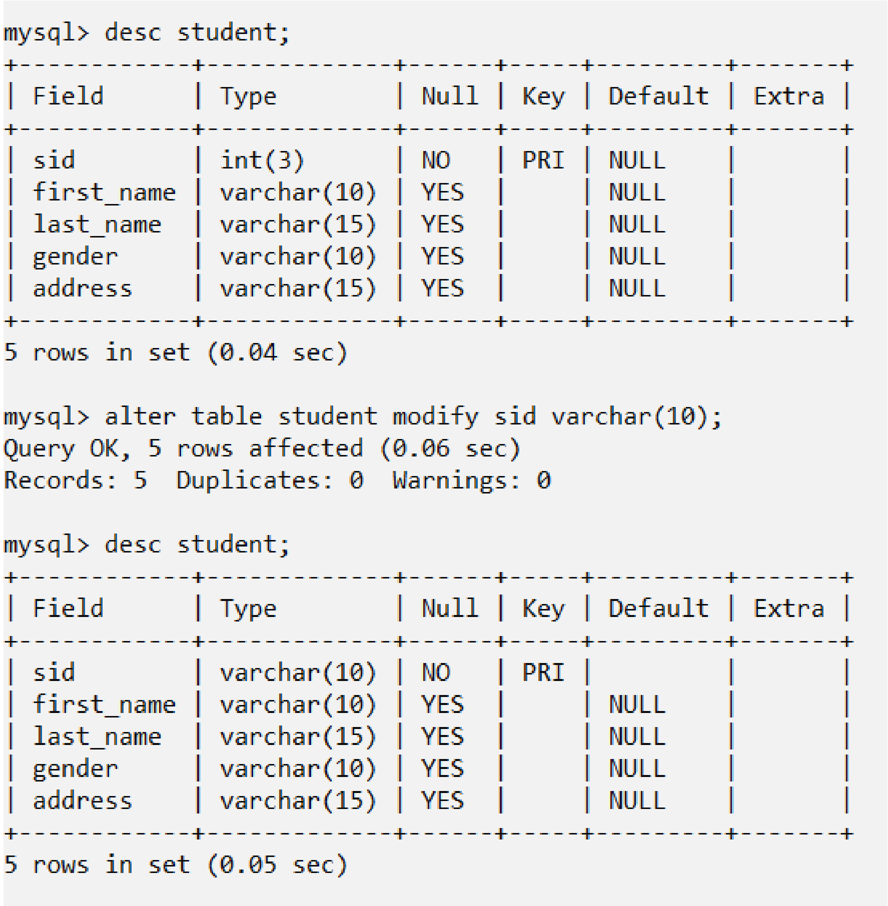
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)

mysql> alter table student add (address varchar(15) default 'pulchowk');
Query OK, 5 rows affected (0.05 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+----------+-------+
| Field      | Type        | Null | Key | Default  | Extra |
+------------+-------------+------+-----+----------+-------+
| sid        | int(3)      | NO   | PRI | NULL     |       |
| first_name | varchar(10) | YES  |     | NULL     |       |
| last_name  | varchar(15) | YES  |     | NULL     |       |
| gender     | varchar(10) | YES  |     | NULL     |       |
| address    | varchar(15) | YES  |     | pulchowk |       |
+------------+-------------+------+-----+----------+-------+
5 rows in set (0.04 sec)
```

3.3.5 Change Datatype Of A Particular Column

It Changes The Datatype Of A Particular Column.

Syntax:

```
Alter table table_name modify column_name datatype(length);
```



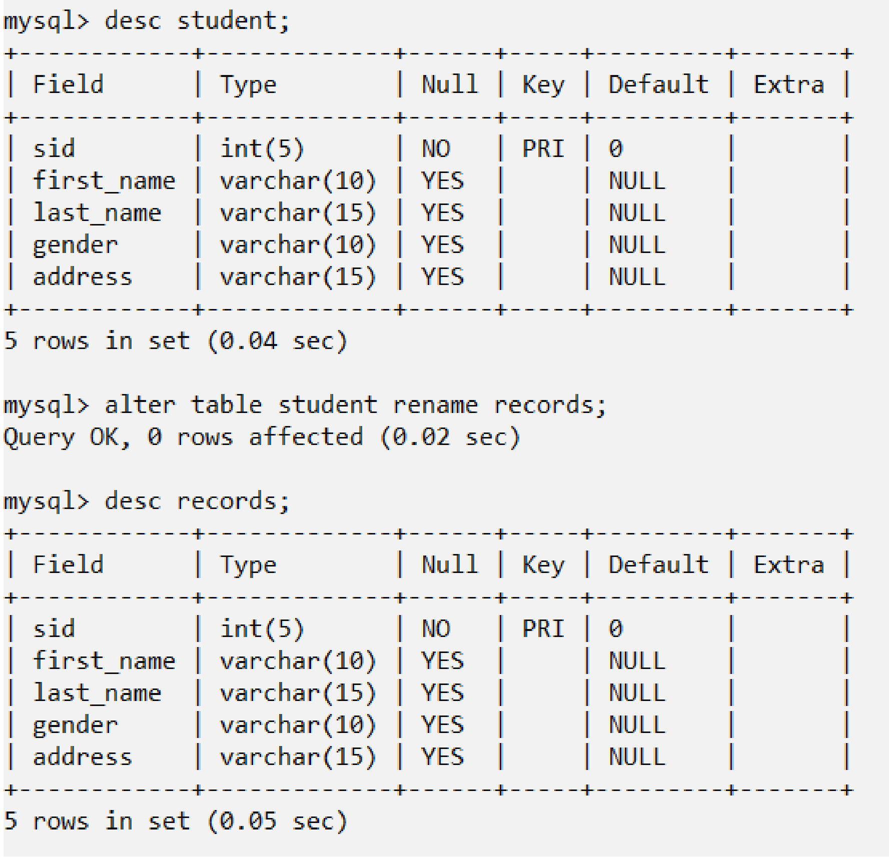
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(3)      | NO   | PRI | NULL    |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.04 sec)

mysql> alter table student modify sid varchar(10);
Query OK, 5 rows affected (0.06 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | varchar(10) | NO   | PRI |         |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.05 sec)
```

3.3.6 Rename Table

This Command Is Used To Rename Table.It Replaces Old Name With New One.

Syntax:

```
Alter table table_name rename new table_name;
```



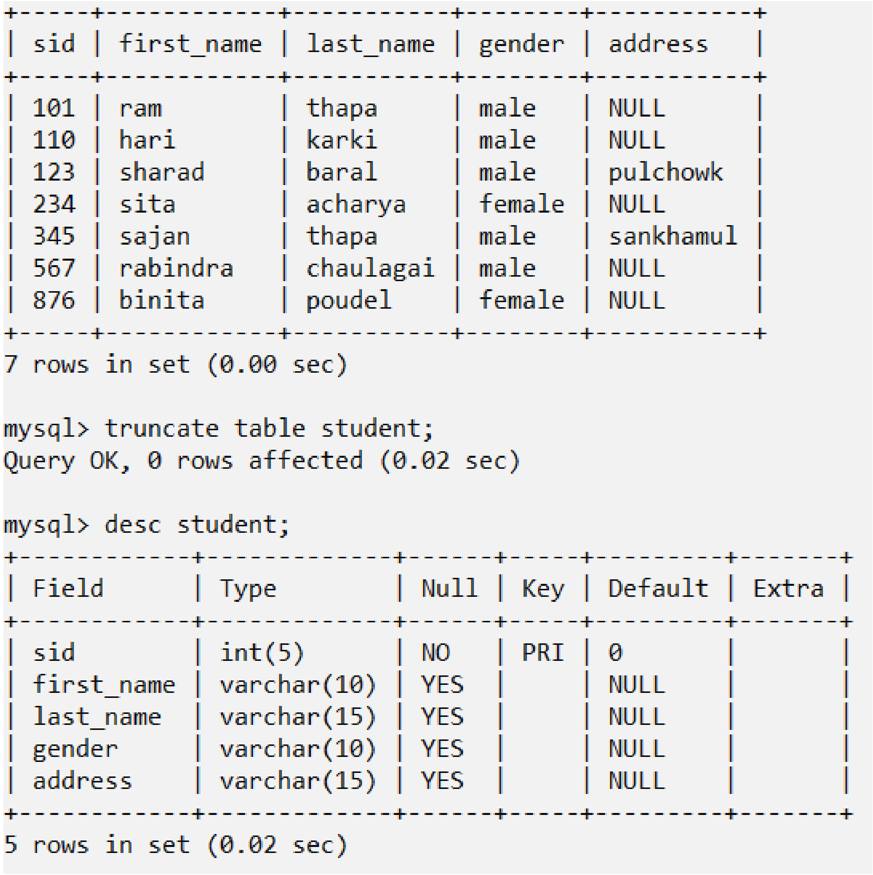
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.04 sec)

mysql> alter table student rename records;
Query OK, 0 rows affected (0.02 sec)

mysql> desc records;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.05 sec)
```

3.4 Truncate Table

This Command Is Used To Remove All Rows From A Table But The Table Structure And Its Column, Constraints, Index, Remain.

Syntax:

```
Truncate table table_name:
```



```
+-----+------------+-----------+--------+-----------+
| sid | first_name | last_name | gender | address   |
+-----+------------+-----------+--------+-----------+
| 101 | ram        | thapa     | male   | NULL      |
| 110 | hari       | karki     | male   | NULL      |
| 123 | sharad     | baral     | male   | pulchowk  |
| 234 | sita       | acharya   | female | NULL      |
| 345 | sajan      | thapa     | male   | sankhamul |
| 567 | rabindra   | chaulagai | male   | NULL      |
| 876 | binita     | poudel    | female | NULL      |
+-----+------------+-----------+--------+-----------+
7 rows in set (0.00 sec)

mysql> truncate table student;
Query OK, 0 rows affected (0.02 sec)

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.02 sec)
```

3.6 Difference Between Truncate And Drop Command.

Drop Command	Truncate Command
The Drop Command Is Used To Remove Table Definition And Its Contents.	The Truncate Command Is Used To Delete All The Rows From The Table.
View Of Table Does Not Exists.	View Of Table Exists.
Integrity, Constraints Will Be Removed.	Integrity, Constraints Will Not Be Removed.

CHAPER-4: SQL CONSTRAINTS

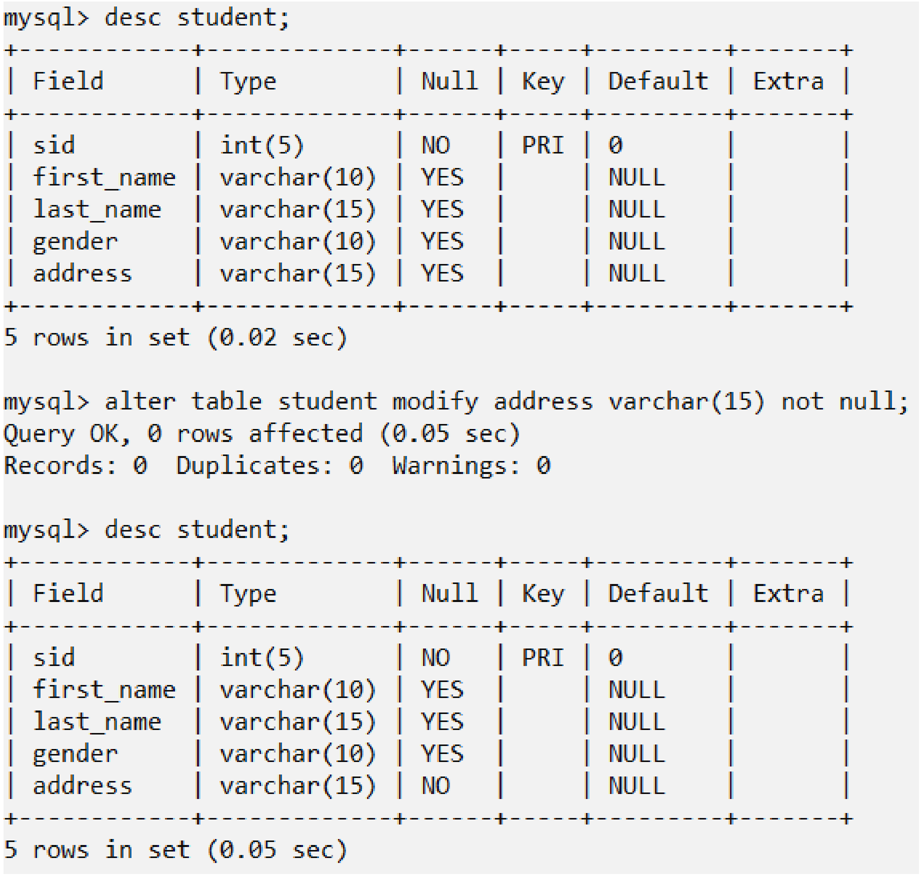
SQL Constraints Are Used To Specify Rules For The Data In A Table. Constraints Are Used To Limit The Type Of Data That Can Go Into A Table. This Ensures The Accuracy And Reliability Of The Data In The Table. If There Are Any Violations Between The Constraints And The Data Action, The Action Is Aborted.

4.1 Not Null

It Ensures That A Column Have A Null Value.

Syntax:

```
Alter table table_name modify column_name datatype(length)
not null;
```



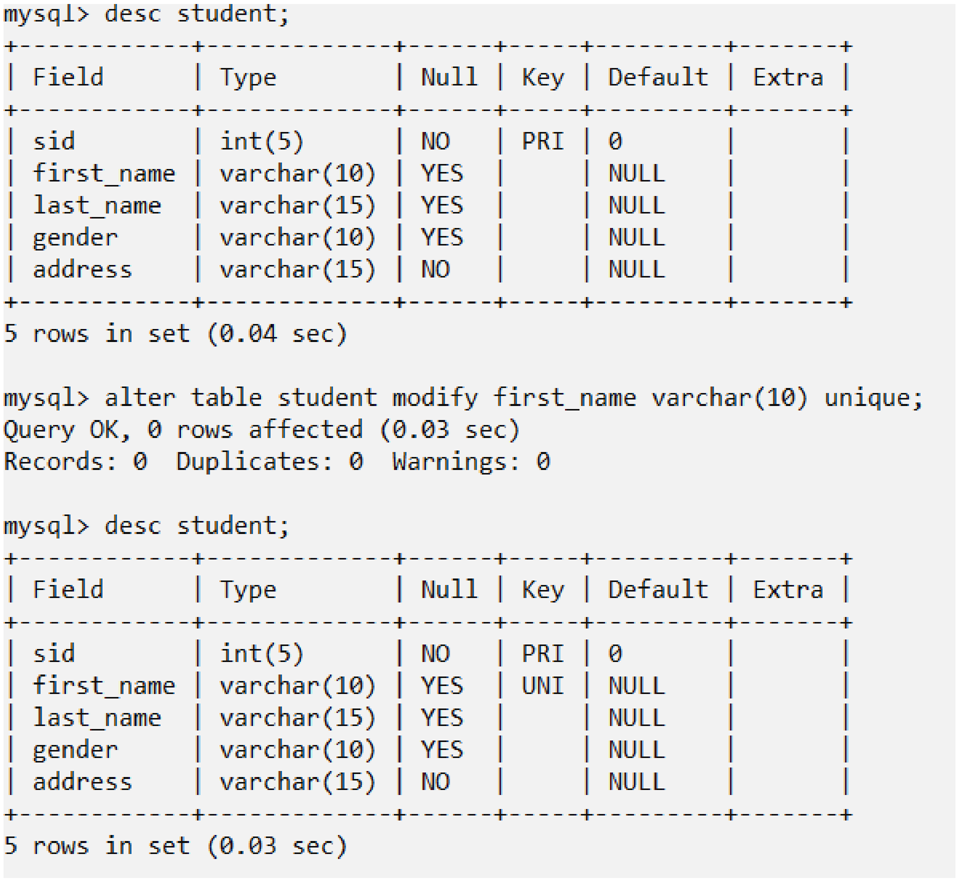
```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.02 sec)

mysql> alter table student modify address varchar(15) not null;
Query OK, 0 rows affected (0.05 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | NO   |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.05 sec)
```

4.2 Unique

It Ensures Constraints That All Values In A Column Are Different. Both The Unique And Primary Key Constraints Provide A Guarantee Of Uniqueness For A Column Or Set Of A Column Or .A Primary Key Constraints Automatically Has Unique Constraints.

Syntax:

```
Alter table table_name modify column_name datatype(length)
unique;
```



```
mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  |     | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | NO   |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.04 sec)

mysql> alter table student modify first_name varchar(10) unique;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| sid        | int(5)      | NO   | PRI | 0       |       |
| first_name | varchar(10) | YES  | UNI | NULL    |       |
| last_name  | varchar(15) | YES  |     | NULL    |       |
| gender     | varchar(10) | YES  |     | NULL    |       |
| address    | varchar(15) | NO   |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.03 sec)
```

4.3 Default

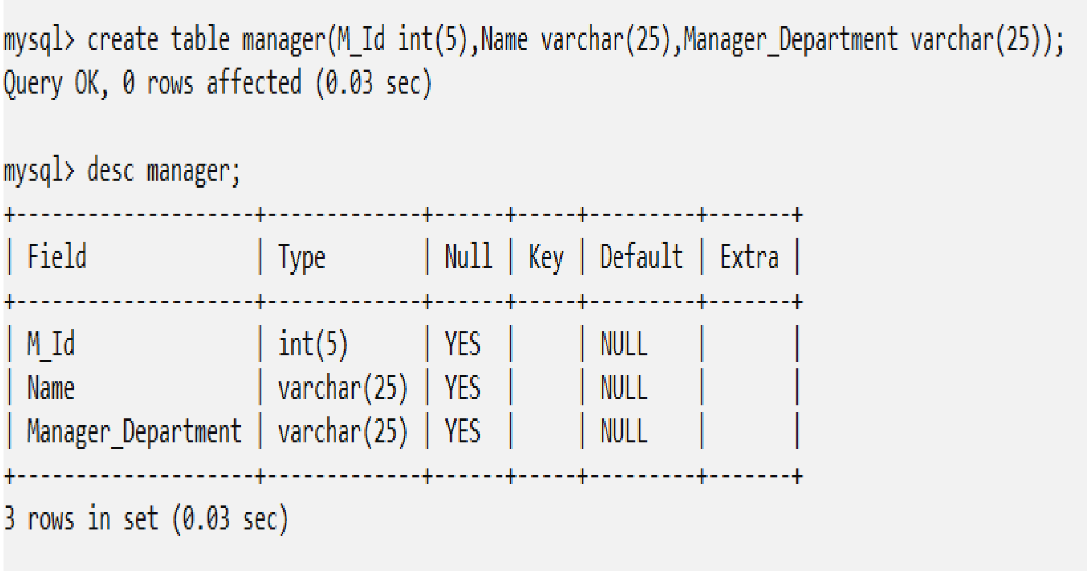
This Is Used To Provide A Default Value For A Column In The Table. The Default Value Will Be Added To All New Records If No Other Value Is Specified.

4.3.1 Default In DDL

Default Values Exists In The Beginning When The Table Exists Or The Column Is Added.

Syntax:

```
Create table table_name(column_name datatype(length )
default default_value;
```



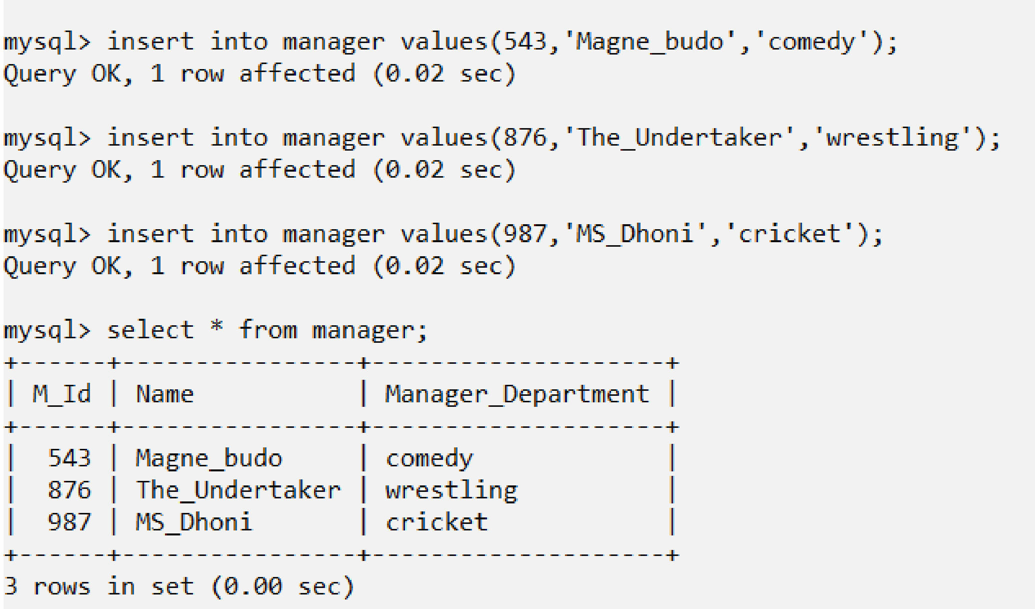
```
mysql> create table manager(M_Id int(5),Name varchar(25),Manager_Department varchar(25));
Query OK, 0 rows affected (0.03 sec)

mysql> desc manager;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| M_Id               | int(5)      | YES  |     | NULL    |       |
| Name               | varchar(25) | YES  |     | NULL    |       |
| Manager_Department | varchar(25) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
3 rows in set (0.03 sec)
```

4.3.2 Default In DML

Default Values Can Be Added If Necessary By Using Default Keyword.

Syntax:

```
Insert into table_name values (data1,data2,default);
```



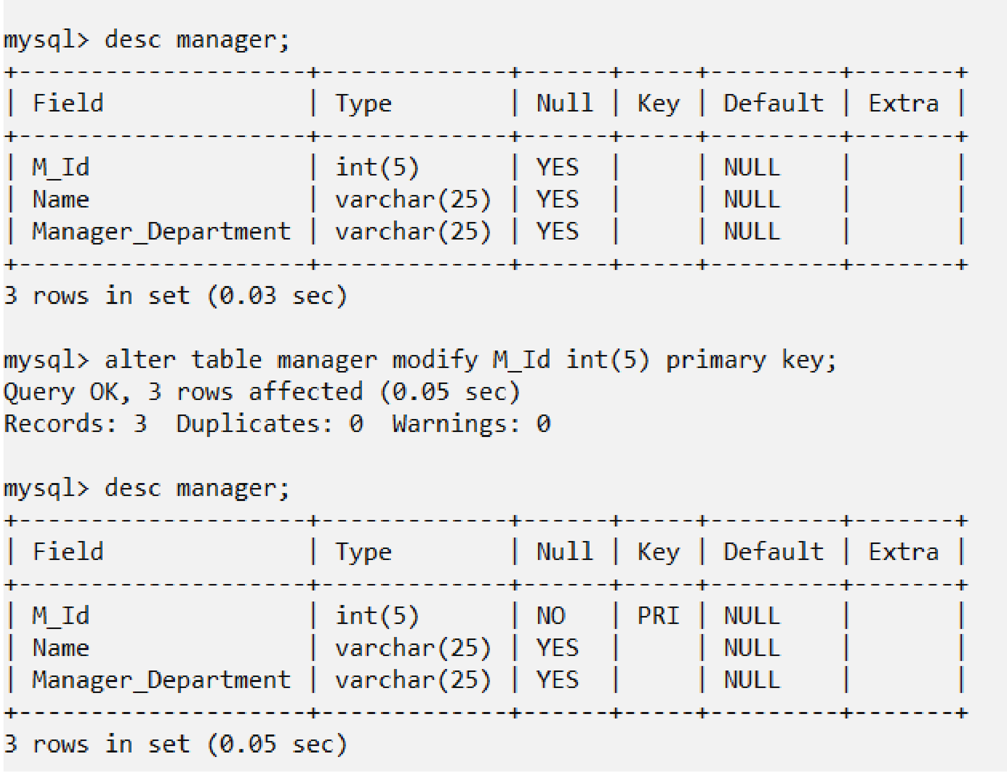
```
mysql> insert into manager values(543,'Magne_budo','comedy');
Query OK, 1 row affected (0.02 sec)

mysql> insert into manager values(876,'The_Undertaker','wrestling');
Query OK, 1 row affected (0.02 sec)

mysql> insert into manager values(987,'MS_Dhoni','cricket');
Query OK, 1 row affected (0.02 sec)

mysql> select * from manager;
+------+----------------+--------------------+
| M_Id | Name           | Manager_Department |
+------+----------------+--------------------+
|  543 | Magne_budo     | comedy             |
|  876 | The_Undertaker | wrestling          |
|  987 | MS_Dhoni       | cricket            |
+------+----------------+--------------------+
3 rows in set (0.00 sec)
```

4.4 Primary Key

It Uniquely Identifies Each Record In Table Primary Keys Must Contain Unique Values And Cannot Contain Null Values In A Table Can Have Only Values A Table Can Have Only One Primary Key And In The Table, This Primary Key Consists Of Single Or Multiple Columns.

Syntax:

```
Alter table table_name modify column_name datatype(length)
primary key;
```



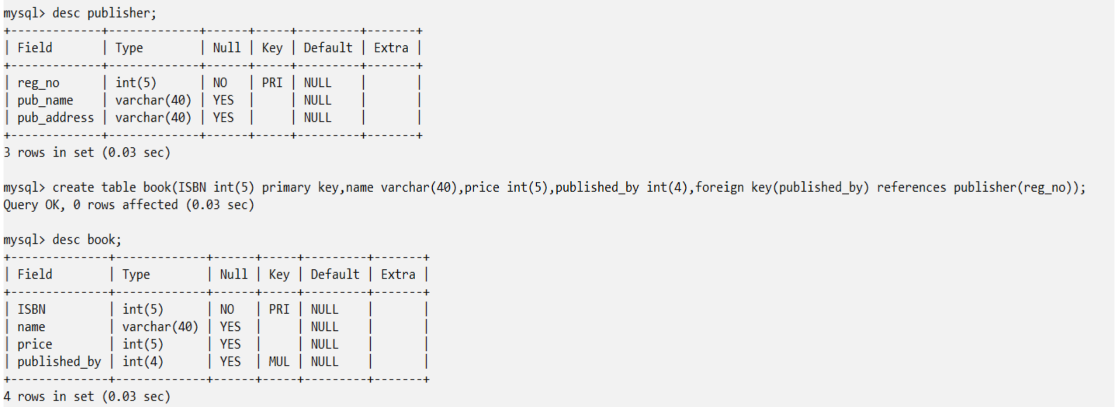
```
mysql> desc manager;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| M_Id               | int(5)      | YES  |     | NULL    |       |
| Name               | varchar(25) | YES  |     | NULL    |       |
| Manager_Department | varchar(25) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
3 rows in set (0.03 sec)

mysql> alter table manager modify M_Id int(5) primary key;
Query OK, 3 rows affected (0.05 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> desc manager;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| M_Id               | int(5)      | NO   | PRI | NULL    |       |
| Name               | varchar(25) | YES  |     | NULL    |       |
| Manager_Department | varchar(25) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
3 rows in set (0.05 sec)
```

4.5 Foreign Key

This Is A Key Which Is Used To Link Two Tables Together.
[Publisher=Parent_Table Book=Child_Table]

Syntax:

```
Alter table table_name add foreign key (column_name)
references parent_table_name(parent_table_attribute);
```



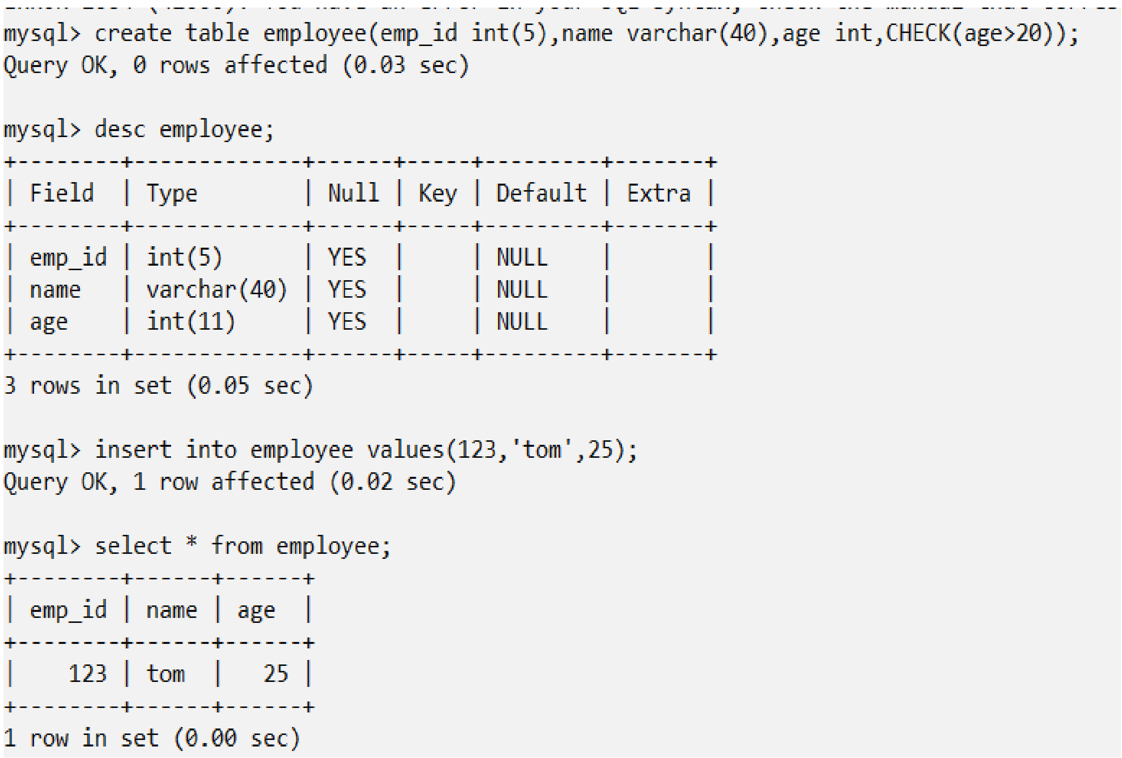
```
mysql> desc publisher;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| reg_no      | int(5)      | NO   | PRI | NULL    |       |
| pub_name    | varchar(40) | YES  |     | NULL    |       |
| pub_address | varchar(40) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.03 sec)

mysql> create table book(ISBN int(5) primary key,name varchar(40),price int(5),published_by int(4),foreign key(published_by) references publisher(reg_no));
Query OK, 0 rows affected (0.03 sec)

mysql> desc book;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| ISBN         | int(5)      | NO   | PRI | NULL    |       |
| name         | varchar(40) | YES  |     | NULL    |       |
| price        | int(5)      | YES  |     | NULL    |       |
| published_by | int(4)      | YES  | MUL | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.03 sec)
```

4.6 Check Constraints

This Command Is Used To Limit The Value Range That Can Be Placed In Column.

Syntax:

```
Create table table_name(column 1 datatype(length) check
(condition);
```



```
mysql> create table employee(emp_id int(5),name varchar(40),age int,CHECK(age>20));
Query OK, 0 rows affected (0.03 sec)

mysql> desc employee;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| emp_id | int(5)      | YES  |     | NULL    |       |
| name   | varchar(40) | YES  |     | NULL    |       |
| age    | int(11)     | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
3 rows in set (0.05 sec)

mysql> insert into employee values(123,'tom',25);
Query OK, 1 row affected (0.02 sec)

mysql> select * from employee;
+--------+------+------+
| emp_id | name | age  |
+--------+------+------+
|    123 | tom  |   25 |
+--------+------+------+
1 row in set (0.00 sec)
```

CHAPTER-5: DATA MANIPULATION LANGUAGE (DML)

Data manipulation language (DML) is a subset of SQL statements that modify the data stored in table's .the list of DML statements are listed below:

a) Insert statement

b) Delete statement

c) Update statement

d) Select statement

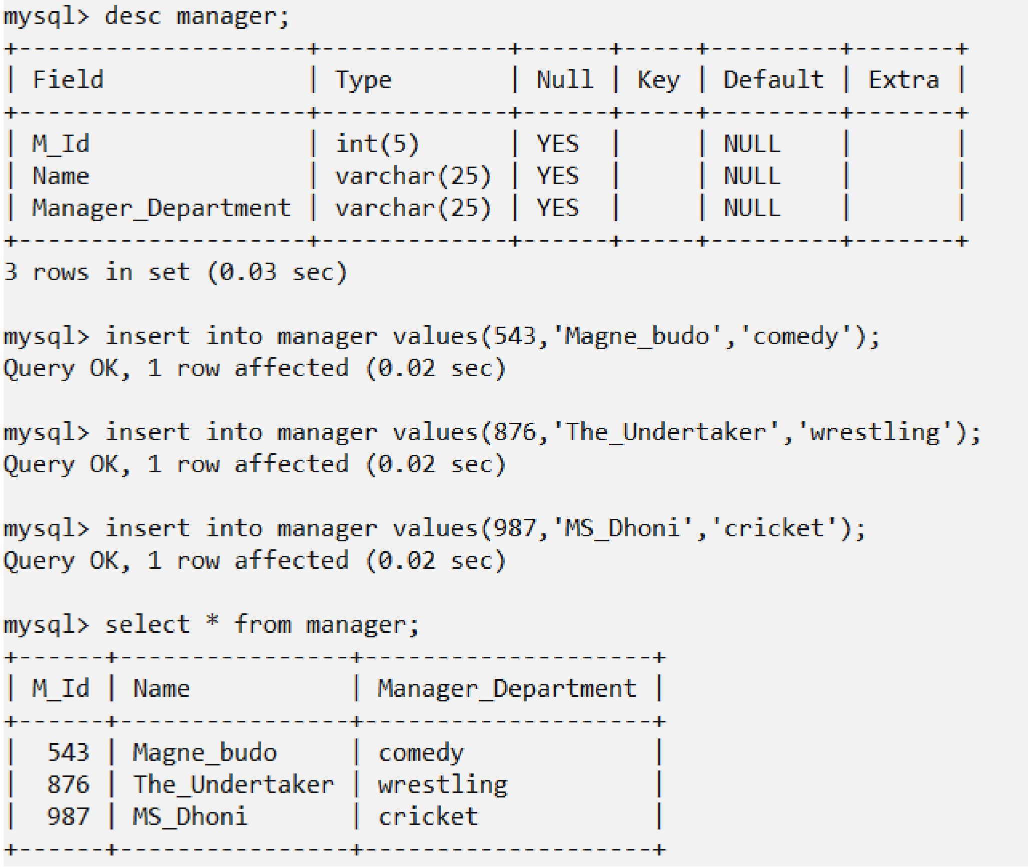
CHAPTER-6: DML COMMANDS

6.1 Insert A Record In A Table.

It Helps In Inserting A Record In The Table .The Orders In Which The Data Is Entered Must Match The Orders Of The Column In The Schema.

Syntax:

```
Insert into table_name values (data1,data2…………………datan);
```



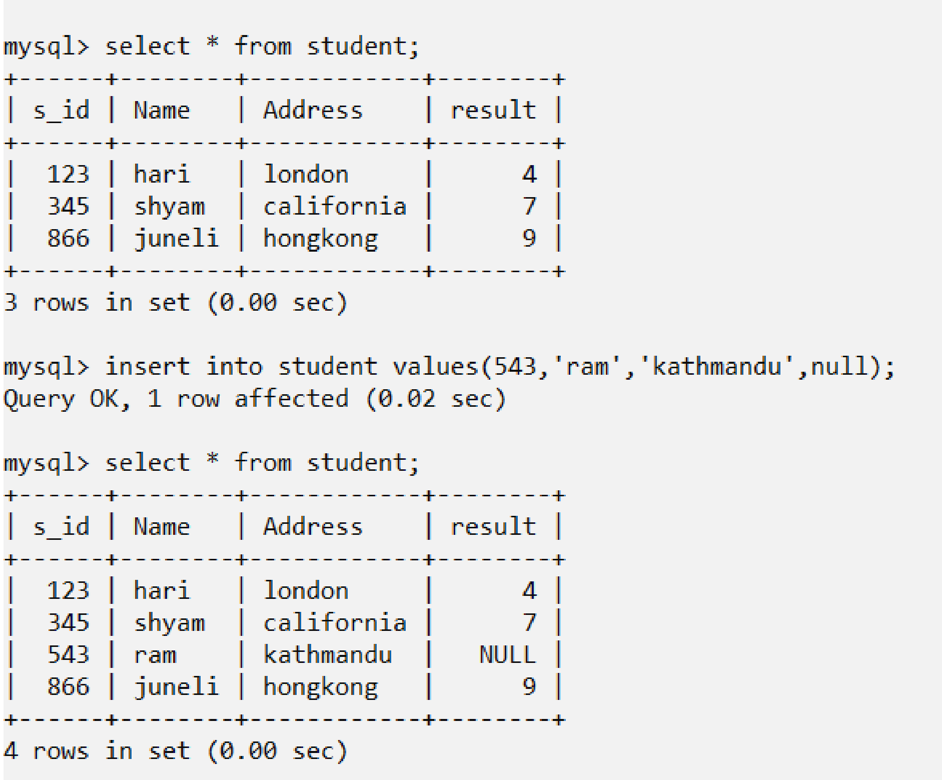
```
mysql> desc manager;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| M_Id               | int(5)      | YES  |     | NULL    |       |
| Name               | varchar(25) | YES  |     | NULL    |       |
| Manager_Department | varchar(25) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
3 rows in set (0.03 sec)

mysql> insert into manager values(543,'Magne_budo','comedy');
Query OK, 1 row affected (0.02 sec)

mysql> insert into manager values(876,'The_Undertaker','wrestling');
Query OK, 1 row affected (0.02 sec)

mysql> insert into manager values(987,'MS_Dhoni','cricket');
Query OK, 1 row affected (0.02 sec)

mysql> select * from manager;
+------+----------------+--------------------+
| M_Id | Name           | Manager_Department |
+------+----------------+--------------------+
|  543 | Magne_budo     | comedy             |
|  876 | The_Undertaker | wrestling          |
|  987 | MS_Dhoni       | cricket            |
+------+----------------+--------------------+
```

6.2 Inserting A Null Value In Table Column.

It Refers To Adding Null Value In A Column Of A Table.

Syntax:

```
Insert into table_name values
(data1,data2,null,data4……,null);
```



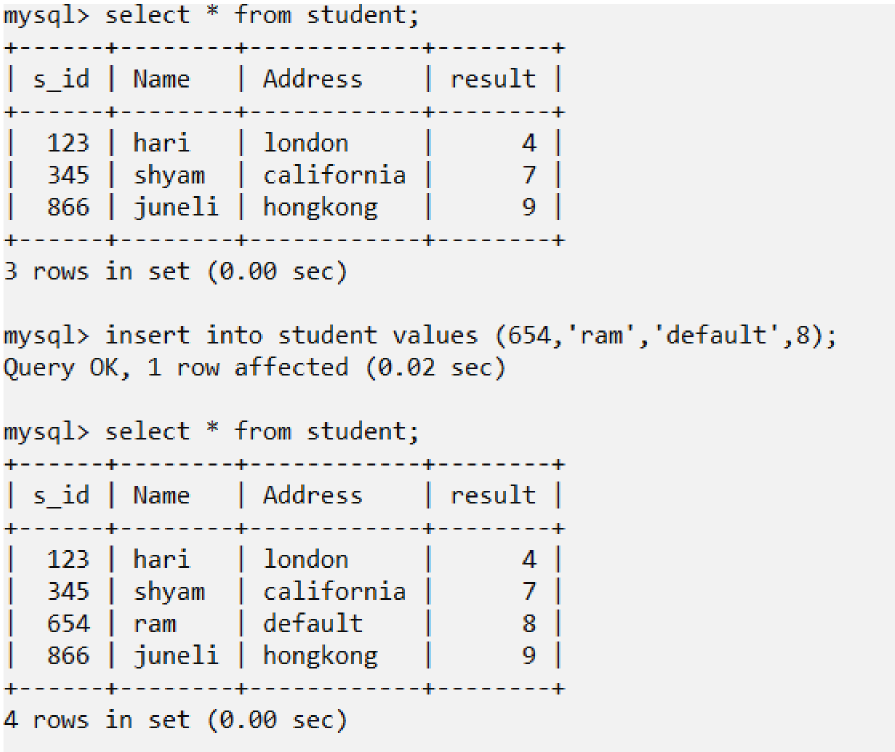
```
mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
3 rows in set (0.00 sec)

mysql> insert into student values(543,'ram','kathmandu',null);
Query OK, 1 row affected (0.02 sec)

mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  543 | ram    | kathmandu  |   NULL |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
4 rows in set (0.00 sec)
```

6.3 Inserting Default Value In A Table.

It Enables Default Value In A Column.

Syntax:

```
Insert into tables_name values(data1,data2,data3,default);
```



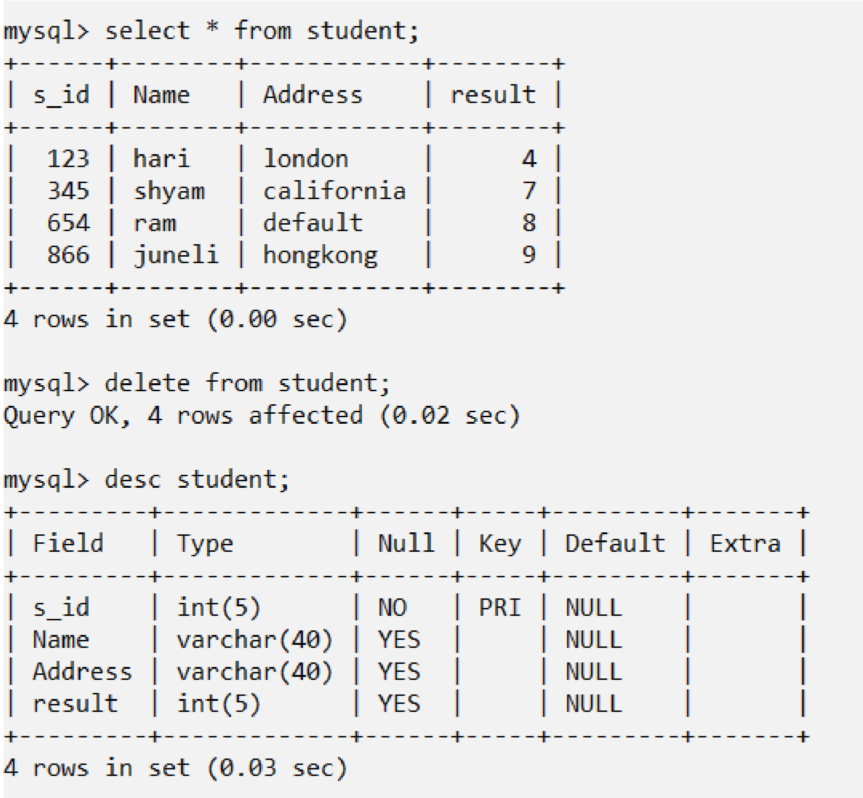
```
mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
3 rows in set (0.00 sec)

mysql> insert into student values (654,'ram','default',8);
Query OK, 1 row affected (0.02 sec)

mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  654 | ram    | default    |      8 |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
4 rows in set (0.00 sec)
```

6.4 Delete All Records Of A Table.

It Enables To Delete All Records Of The Table.

Syntax:

```
Delete from table_name;
```



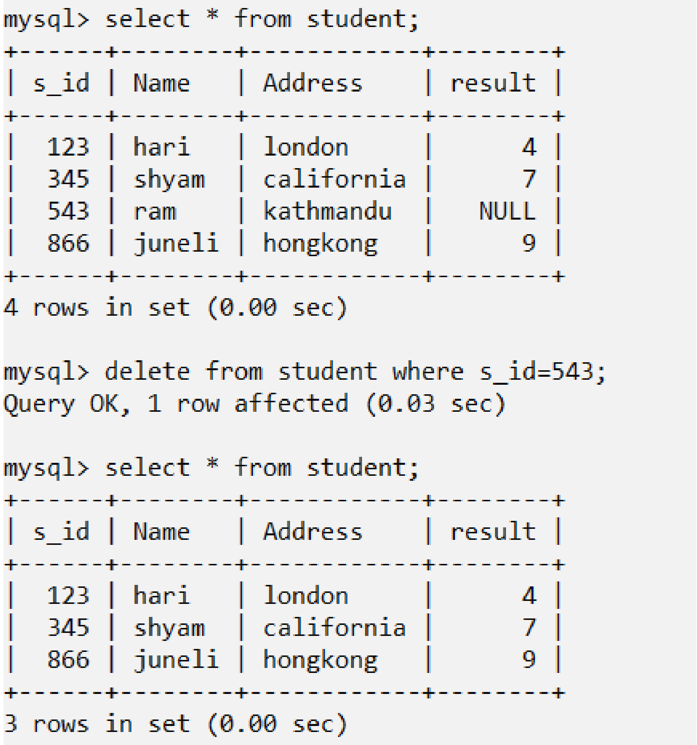
```
mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  654 | ram    | default    |      8 |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
4 rows in set (0.00 sec)

mysql> delete from student;
Query OK, 4 rows affected (0.02 sec)

mysql> desc student;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| s_id    | int(5)      | NO   | PRI | NULL    |       |
| Name    | varchar(40) | YES  |     | NULL    |       |
| Address | varchar(40) | YES  |     | NULL    |       |
| result  | int(5)      | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
4 rows in set (0.03 sec)
```

6.5 Delete A Particular Record Using Where Clause.

It Deletes A Particular Record From The Table.

Syntax:

```
Delete from table_name where column_name=column value;
```



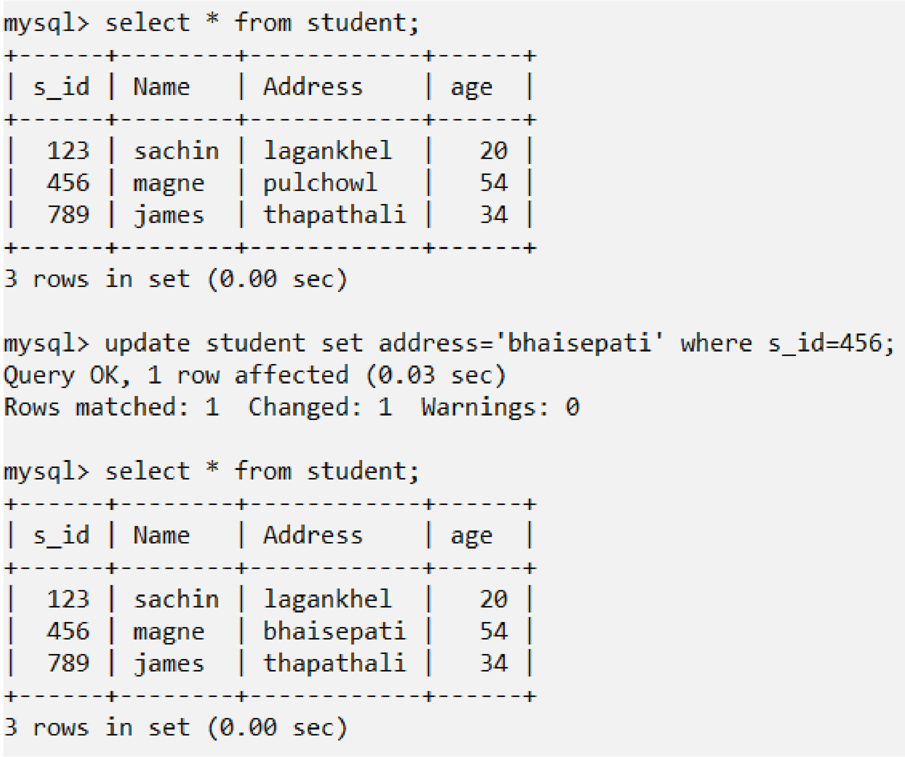
```
mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  543 | ram    | kathmandu  |   NULL |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
4 rows in set (0.00 sec)

mysql> delete from student where s_id=543;
Query OK, 1 row affected (0.03 sec)

mysql> select * from student;
+------+--------+------------+--------+
| s_id | Name   | Address    | result |
+------+--------+------------+--------+
|  123 | hari   | london     |      4 |
|  345 | shyam  | california |      7 |
|  866 | juneli | hongkong   |      9 |
+------+--------+------------+--------+
3 rows in set (0.00 sec)
```

6.6 Update A Single Column Value Of A Record.

It Is Used To Modify Single Column Value Of A Record.

Syntax:

```
Update table_name set attribute =values primary key=value;
```



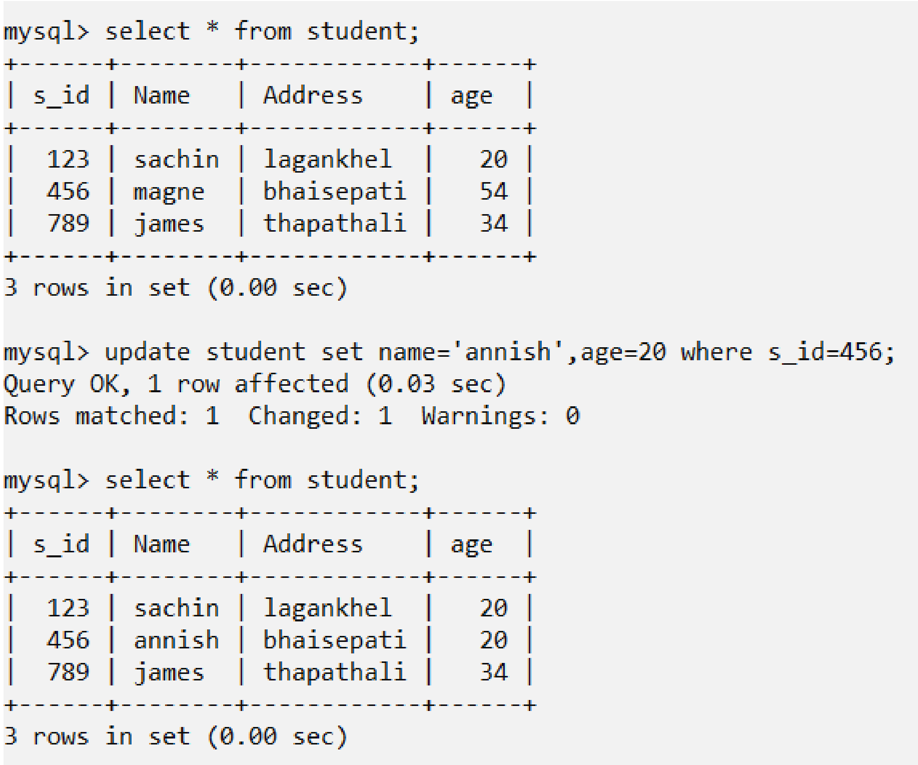
```
mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | magne  | pulchowl   |   54 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)

mysql> update student set address='bhaisepati' where s_id=456;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | magne  | bhaisepati |   54 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)
```

6.7 Update Multiple Column Value Of A Record.

It Is Used To Modify Multiple Column Value Of A Record.

Syntax:

```
Update table_name set
column1_name=value,column2_name=value……..where
‘condition’;.
```



```
mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | magne  | bhaisepati |   54 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)

mysql> update student set name='annish',age=20 where s_id=456;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | annish | bhaisepati |   20 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)
```

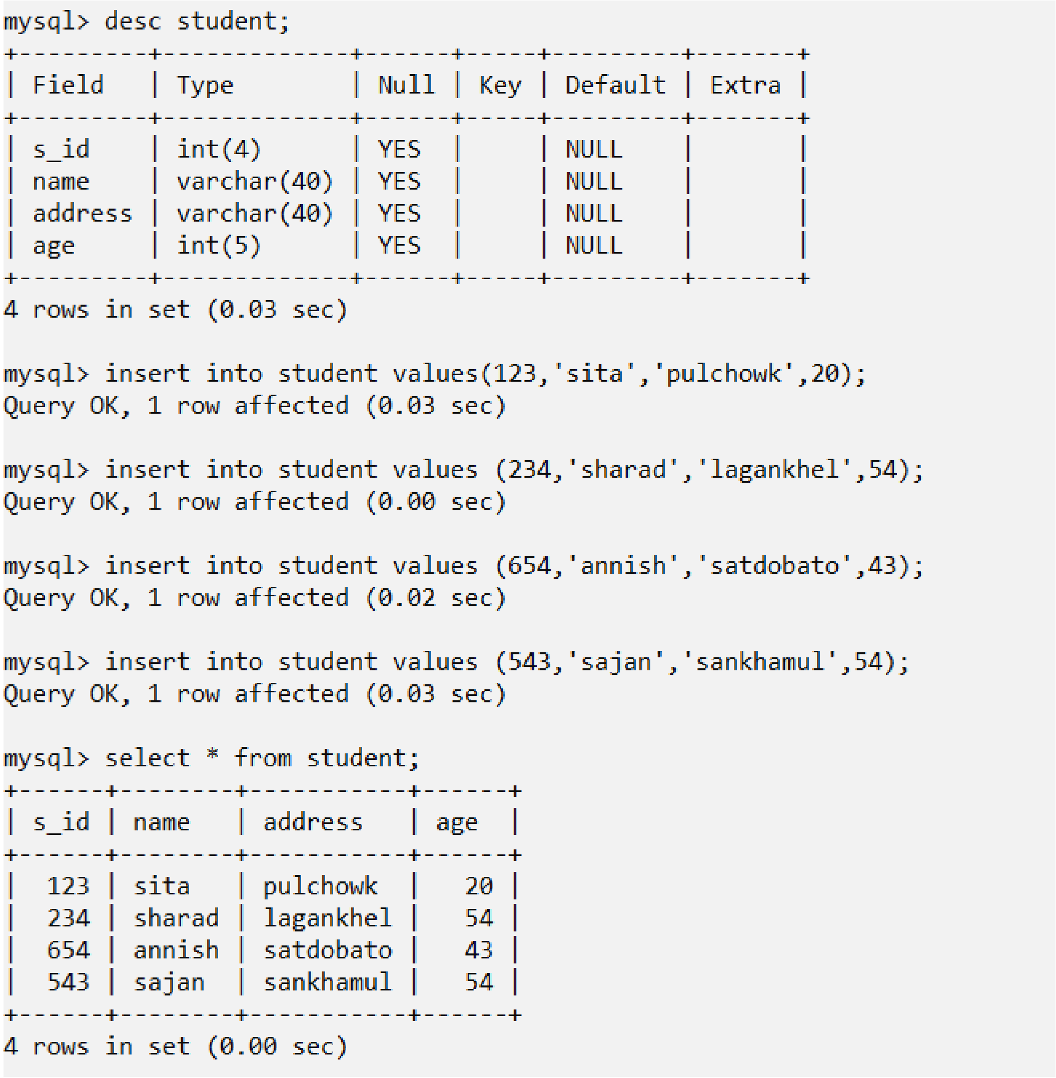
CHAPTER-7: SELECT STATEMENT

7.1 Selecting All Records

It Is Used To Retrieve All Records Of The Table.

Syntax:

```
Select * from table_name;
```



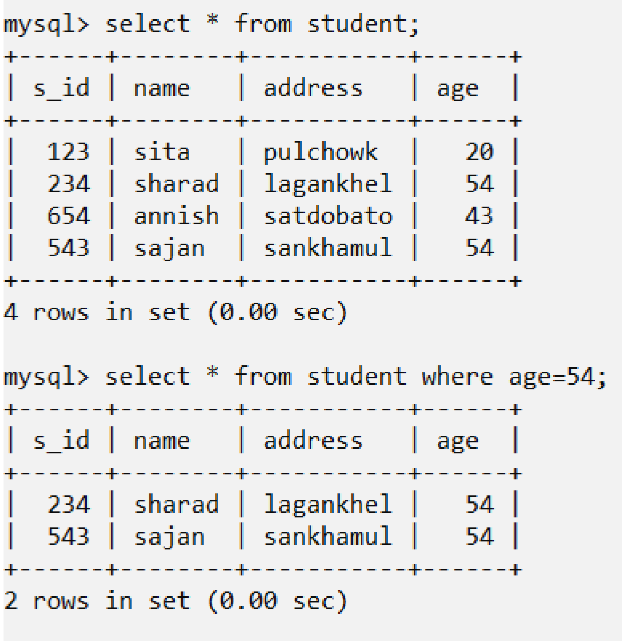
```
mysql> desc student;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| s_id    | int(4)      | YES  |     | NULL    |       |
| name    | varchar(40) | YES  |     | NULL    |       |
| address | varchar(40) | YES  |     | NULL    |       |
| age     | int(5)      | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
4 rows in set (0.03 sec)

mysql> insert into student values(123,'sita','pulchowk',20);
Query OK, 1 row affected (0.03 sec)

mysql> insert into student values (234,'sharad','lagankhel',54);
Query OK, 1 row affected (0.00 sec)

mysql> insert into student values (654,'annish','satdobato',43);
Query OK, 1 row affected (0.02 sec)

mysql> insert into student values (543,'sajan','sankhamul',54);
Query OK, 1 row affected (0.03 sec)

mysql> select * from student;
+------+--------+-----------+------+
| s_id | name   | address   | age  |
+------+--------+-----------+------+
|  123 | sita   | pulchowk  |   20 |
|  234 | sharad | lagankhel |   54 |
|  654 | annish | satdobato |   43 |
|  543 | sajan  | sankhamul |   54 |
+------+--------+-----------+------+
4 rows in set (0.00 sec)
```

7.2 Selecting Records Based On Condition

It Is Used To Extract Only Those Records That Fulfill A Specified Condition.

Syntax:

```
Select column_name from table_name where column_name=value;
```



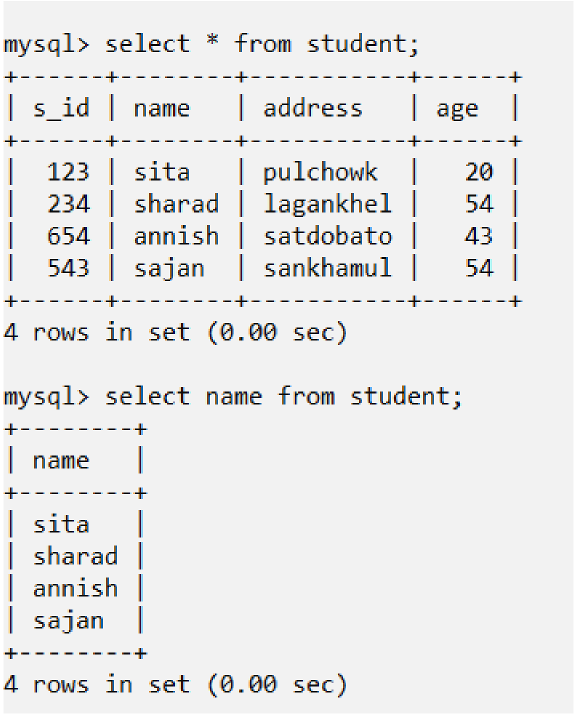
```
mysql> select * from student;
+------+--------+----------+------+
| s_id | name   | address  | age  |
+------+--------+----------+------+
|  123 | sita   | pulchowk |   20 |
|  234 | sharad | lagankhel |   54 |
|  654 | annish | satdobato |   43 |
|  543 | sajan  | sankhamul |   54 |
+------+--------+----------+------+
4 rows in set (0.00 sec)

mysql> select * from student where age=54;
+------+--------+----------+------+
| s_id | name   | address  | age  |
+------+--------+----------+------+
|  234 | sharad | lagankhel |   54 |
|  543 | sajan  | sankhamul |   54 |
+------+--------+----------+------+
2 rows in set (0.00 sec)
```

7.3 Selecting Specific Column From A Record

It Is Used To Select Specific Column From Table Name.

Syntax:

```
Select column_name from table_name;
```



```
mysql> select * from student;
+------+--------+-----------+------+
| s_id | name   | address   | age  |
+------+--------+-----------+------+
|  123 | sita   | pulchowk  |   20 |
|  234 | sharad | lagankhel |   54 |
|  654 | annish | satdobato |   43 |
|  543 | sajan  | sankhamul |   54 |
+------+--------+-----------+------+
4 rows in set (0.00 sec)

mysql> select name from student;
+--------+
| name   |
+--------+
| sita   |
| sharad |
| annish |
| sajan  |
+--------+
4 rows in set (0.00 sec)
```

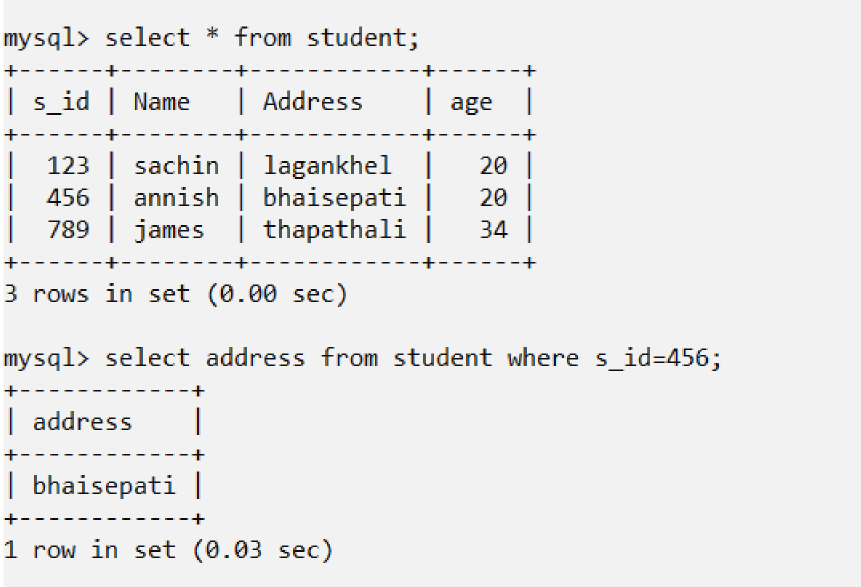
CHAPTER-8: SQL CLAUSES

8.1 Where Clause

It Is Used Extract Only Those Records Which Fulfill The Required Condition.

Syntax:

```
Select column_name from table_name where column_name=value;
```



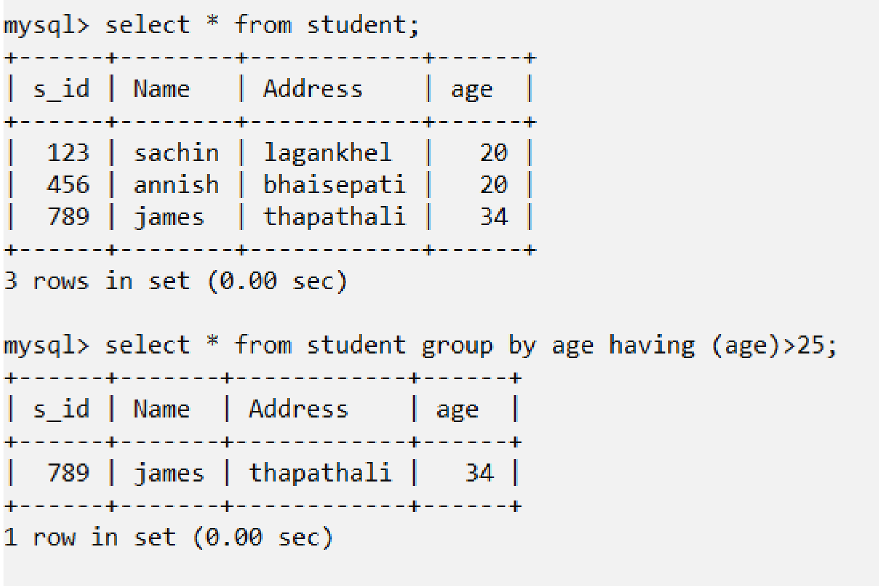
```
mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | annish | bhaisepati |   20 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)

mysql> select address from student where s_id=456;
+------------+
| address    |
+------------+
| bhaisepati |
+------------+
1 row in set (0.03 sec)
```

8.2 Having Clause

This Specifies That An Sql Select Statement Must Only Return Rows Where Aggregate Values Meet The Specified Condition.

Syntax:

```
Select column from table_name group by column_name having
function(column_name)‘condition’;
```



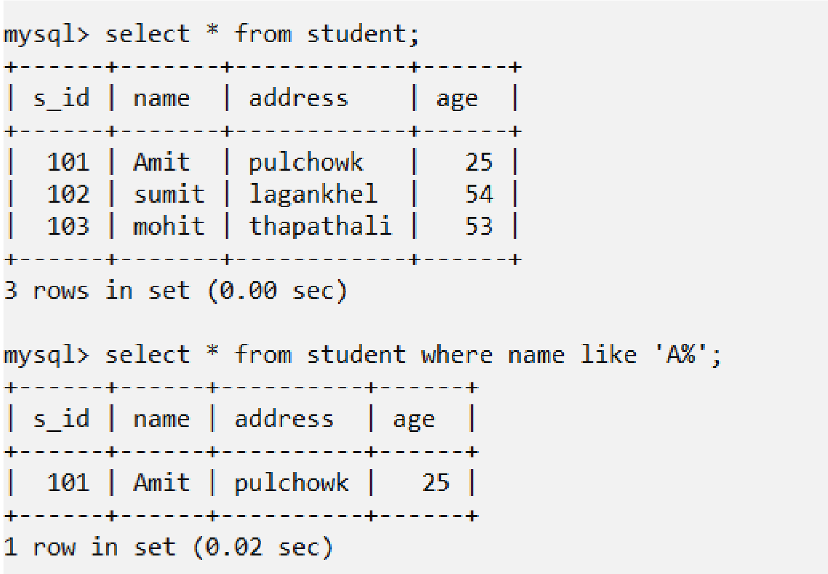
```
mysql> select * from student;
+------+--------+------------+------+
| s_id | Name   | Address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | annish | bhaisepati |   20 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
3 rows in set (0.00 sec)

mysql> select * from student group by age having (age)>25;
+------+-------+------------+------+
| s_id | Name  | Address    | age  |
+------+-------+------------+------+
|  789 | james | thapathali |   34 |
+------+-------+------------+------+
1 row in set (0.00 sec)
```

8.3 Like Clause

This Is Used In Where Clause To Search For A Specified Pattern In A Column.

Syntax:

```
Select * from table_name where column_name like”A%’;
```



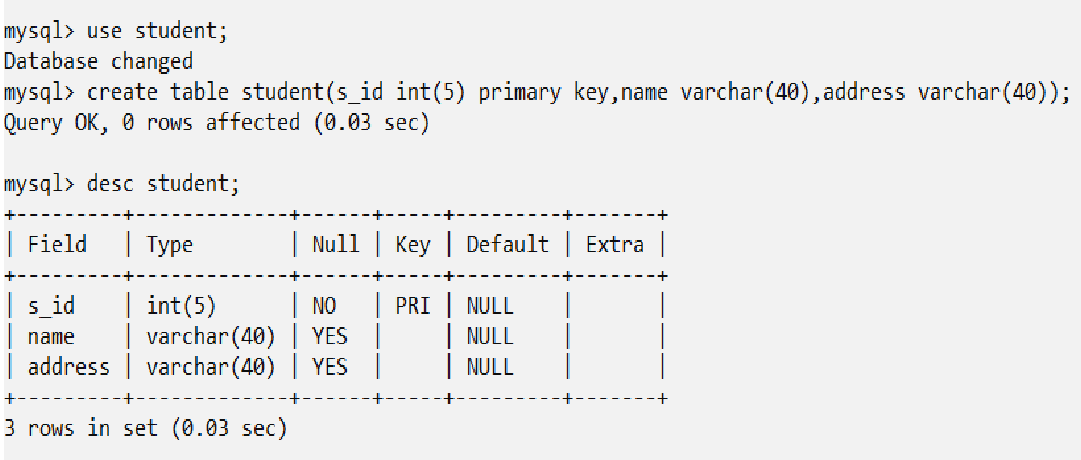
```
mysql> select * from student;
+------+-------+------------+------+
| s_id | name  | address    | age  |
+------+-------+------------+------+
|  101 | Amit  | pulchowk   |   25 |
|  102 | sumit | lagankhel  |   54 |
|  103 | mohit | thapathali |   53 |
+------+-------+------------+------+
3 rows in set (0.00 sec)

mysql> select * from student where name like 'A%';
+------+------+----------+------+
| s_id | name | address  | age  |
+------+------+----------+------+
|  101 | Amit | pulchowk |   25 |
+------+------+----------+------+
1 row in set (0.02 sec)
```

8.4 Describe Clause

It Is Used To Display Information About A Table Like Column Names And Constraints On Column Name.

Syntax:

```
Desc table_name;
```



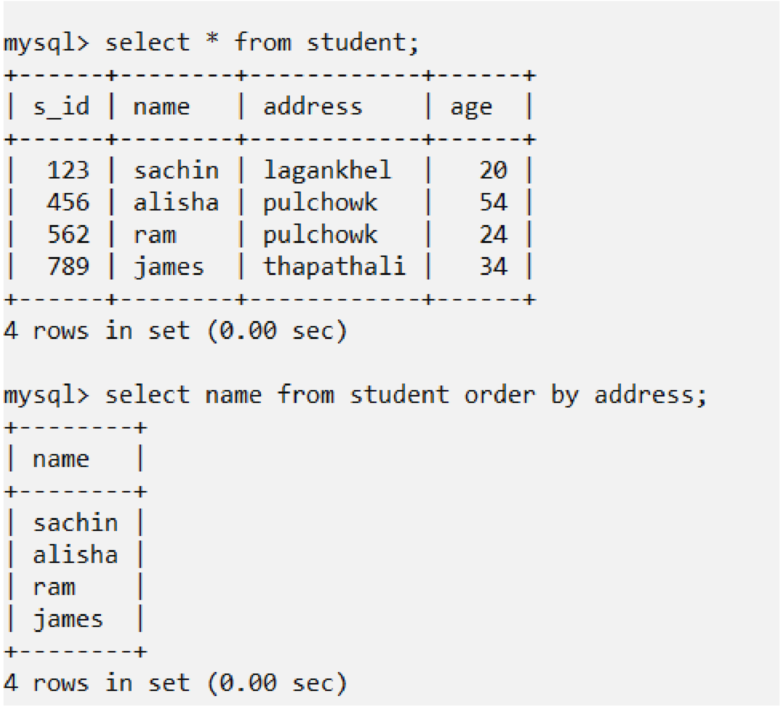
```
mysql> use student;
Database changed
mysql> create table student(s_id int(5) primary key,name varchar(40),address varchar(40));
Query OK, 0 rows affected (0.03 sec)

mysql> desc student;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| s_id    | int(5)      | NO   | PRI | NULL    |       |
| name    | varchar(40) | YES  |     | NULL    |       |
| address | varchar(40) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
3 rows in set (0.03 sec)
```

8.4 Order By Clause

It Is Used To Sort The Query Result Sets In Either Ascending Or Descending Order. [For Sorting In Ascending Order Or Default]

Syntax:

```
Select column_name from table order by column_name;
```



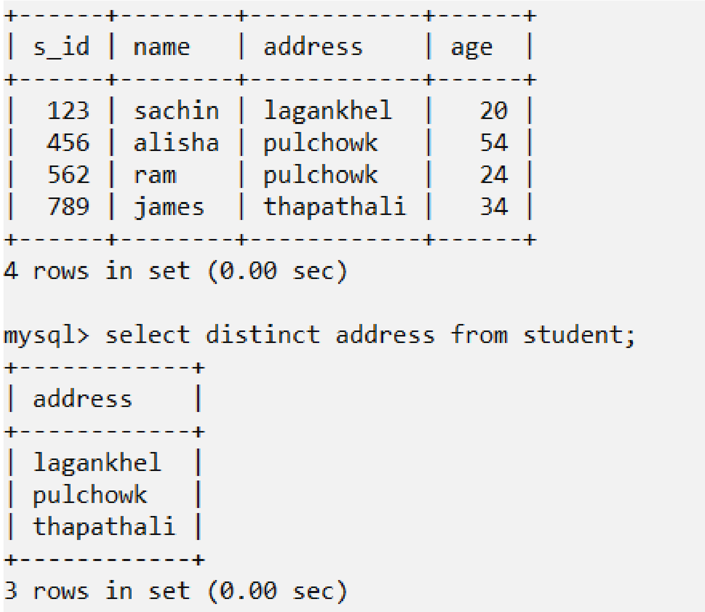
```
mysql> select * from student;
+------+--------+-----------+------+
| s_id | name   | address   | age  |
+------+--------+-----------+------+
|  123 | sachin | lagankhel |   20 |
|  456 | alisha | pulchowk  |   54 |
|  562 | ram    | pulchowk  |   24 |
|  789 | james  | thapathali |   34 |
+------+--------+-----------+------+
4 rows in set (0.00 sec)

mysql> select name from student order by address;
+--------+
| name   |
+--------+
| sachin |
| alisha |
| ram    |
| james  |
+--------+
4 rows in set (0.00 sec)
```

8.7 Distinct Keyword

This Is Used To Eliminate Duplicate Rows And Displays A Unique List Of Values.

Syntax:

```
Select distinct column_name from table_name;
```



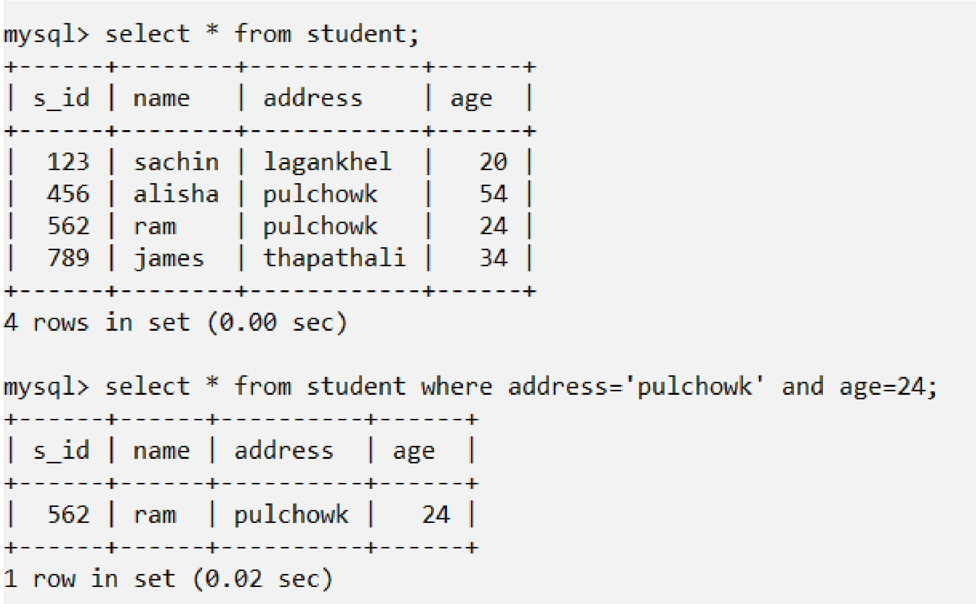
```
+------+--------+------------+------+
| s_id | name   | address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | alisha | pulchowk   |   54 |
|  562 | ram    | pulchowk   |   24 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
4 rows in set (0.00 sec)

mysql> select distinct address from student;
+------------+
| address    |
+------------+
| lagankhel  |
| pulchowk   |
| thapathali |
+------------+
3 rows in set (0.00 sec)
```

8.8 AND Operator For Where Clause

This Operator Combines More Than One Conditions In The Select Statement.

Syntax:

```
Select * from table_name where attribute_name and
attribute_name;
```



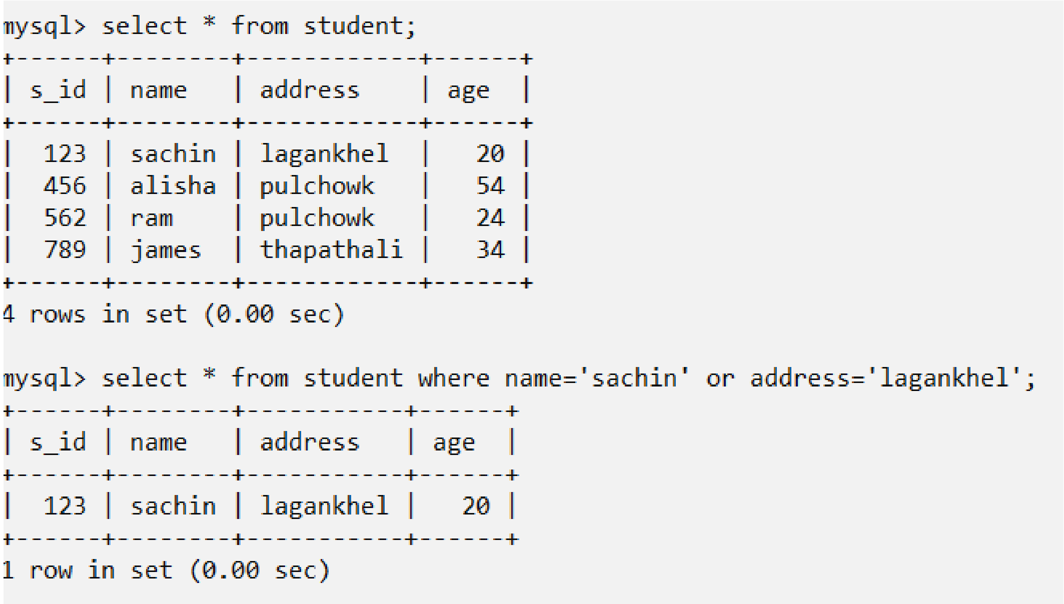
```
mysql> select * from student;
+------+--------+------------+------+
| s_id | name   | address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | alisha | pulchowk   |   54 |
|  562 | ram    | pulchowk   |   24 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
4 rows in set (0.00 sec)

mysql> select * from student where address='pulchowk' and age=24;
+------+------+----------+------+
| s_id | name | address  | age  |
+------+------+----------+------+
|  562 | ram  | pulchowk |   24 |
+------+------+----------+------+
1 row in set (0.02 sec)
```

8.9 OR Operator For Where Clause

This Operator Combines More Than One Condition In Select Statement, But Unlike AND Condition. All Mentioned Conditions Show Not Meet At One I.E. At Least One Of The Continued Should Hold Time.

Syntax:

```
Select * from table_name where “attribute1_name or
attribute2_name”;
```



```
mysql> select * from student;
+------+--------+------------+------+
| s_id | name   | address    | age  |
+------+--------+------------+------+
|  123 | sachin | lagankhel  |   20 |
|  456 | alisha | pulchowk   |   54 |
|  562 | ram    | pulchowk   |   24 |
|  789 | james  | thapathali |   34 |
+------+--------+------------+------+
4 rows in set (0.00 sec)

mysql> select * from student where name='sachin' or address='lagankhel';
+------+--------+-----------+------+
| s_id | name   | address   | age  |
+------+--------+-----------+------+
|  123 | sachin | lagankhel |   20 |
+------+--------+-----------+------+
1 row in set (0.00 sec)
```

CHAPTER-9: SQL VIEW

9.1 Difference Between View And Table.

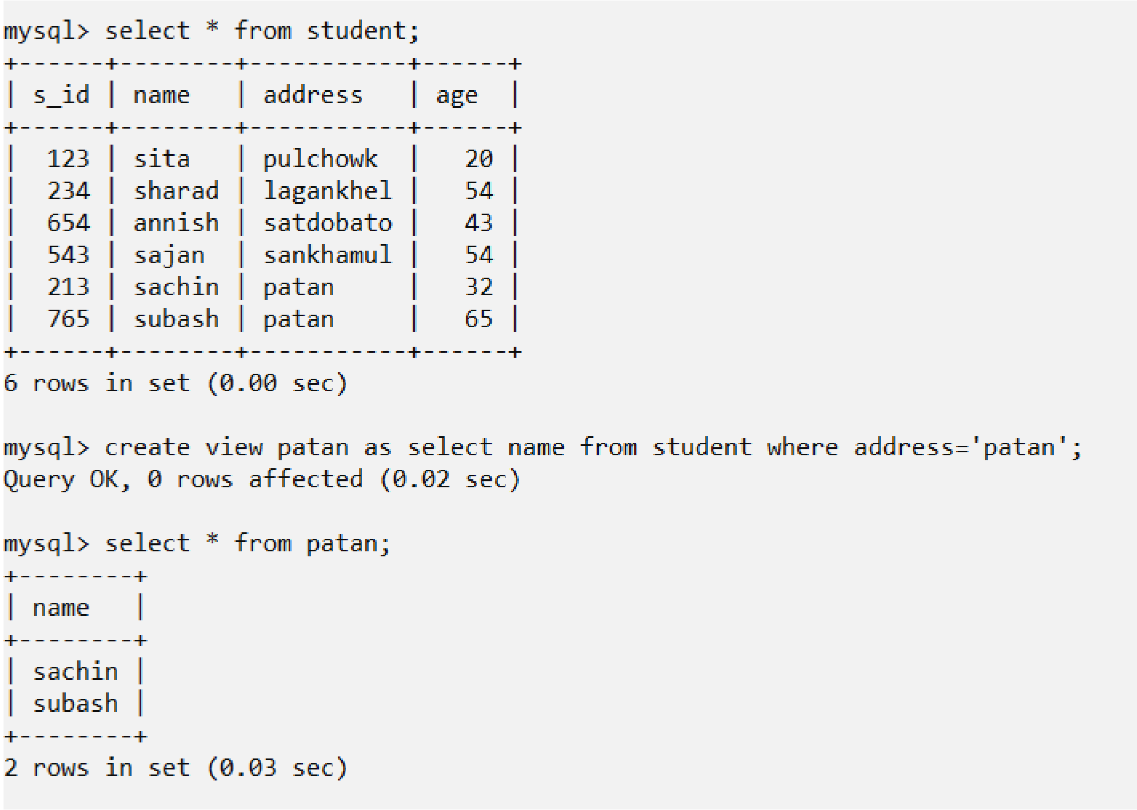
View	Table
View Is A Virtual Table Based On Result Set Of A SQL Statement.	Table Is A Database Object That Consists Of Rows And Columns That Store Data Of Database
A Virtual Table	An Actual Table
View Depends On Table	Table Is Independent Data Object
It Is A Database Object That Allows Generating A Logical Subset Of Data From One Or More Values	A Database Object Or An Entity That Stores That Data Of A Database

9.2 Creating View

It Creates A Separate Mini-Result Set That Stays In Database Like A Normal Table, It Is Actually A Sub Table.

Syntax:

```
Create view view_name as select
column_name1,column_name=value;
```



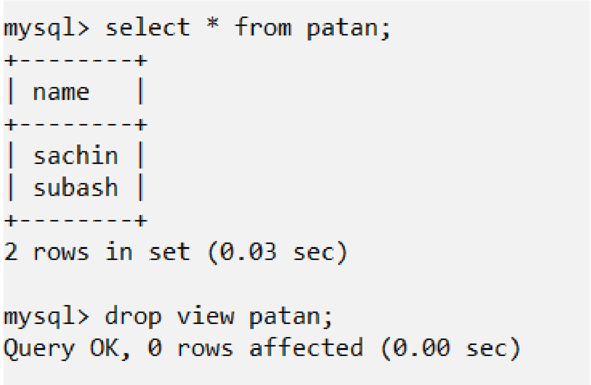
```
mysql> select * from student;
+------+--------+----------+------+
| s_id | name   | address  | age  |
+------+--------+----------+------+
|  123 | sita   | pulchowk |   20 |
|  234 | sharad | lagankhel|   54 |
|  654 | annish | satdobato|   43 |
|  543 | sajan  | sankhamul|   54 |
|  213 | sachin | patan    |   32 |
|  765 | subash | patan    |   65 |
+------+--------+----------+------+
6 rows in set (0.00 sec)

mysql> create view patan as select name from student where address='patan';
Query OK, 0 rows affected (0.02 sec)

mysql> select * from patan;
+--------+
| name   |
+--------+
| sachin |
| subash |
+--------+
2 rows in set (0.03 sec)
```

9.3 Deleting View

It Is Used To Remove One Or More Views.

Syntax:

```
Drop view view_name;
```



```
mysql> select * from patan;
+--------+
| name   |
+--------+
| sachin |
| subash |
+--------+
2 rows in set (0.03 sec)

mysql> drop view patan;
Query OK, 0 rows affected (0.00 sec)
```

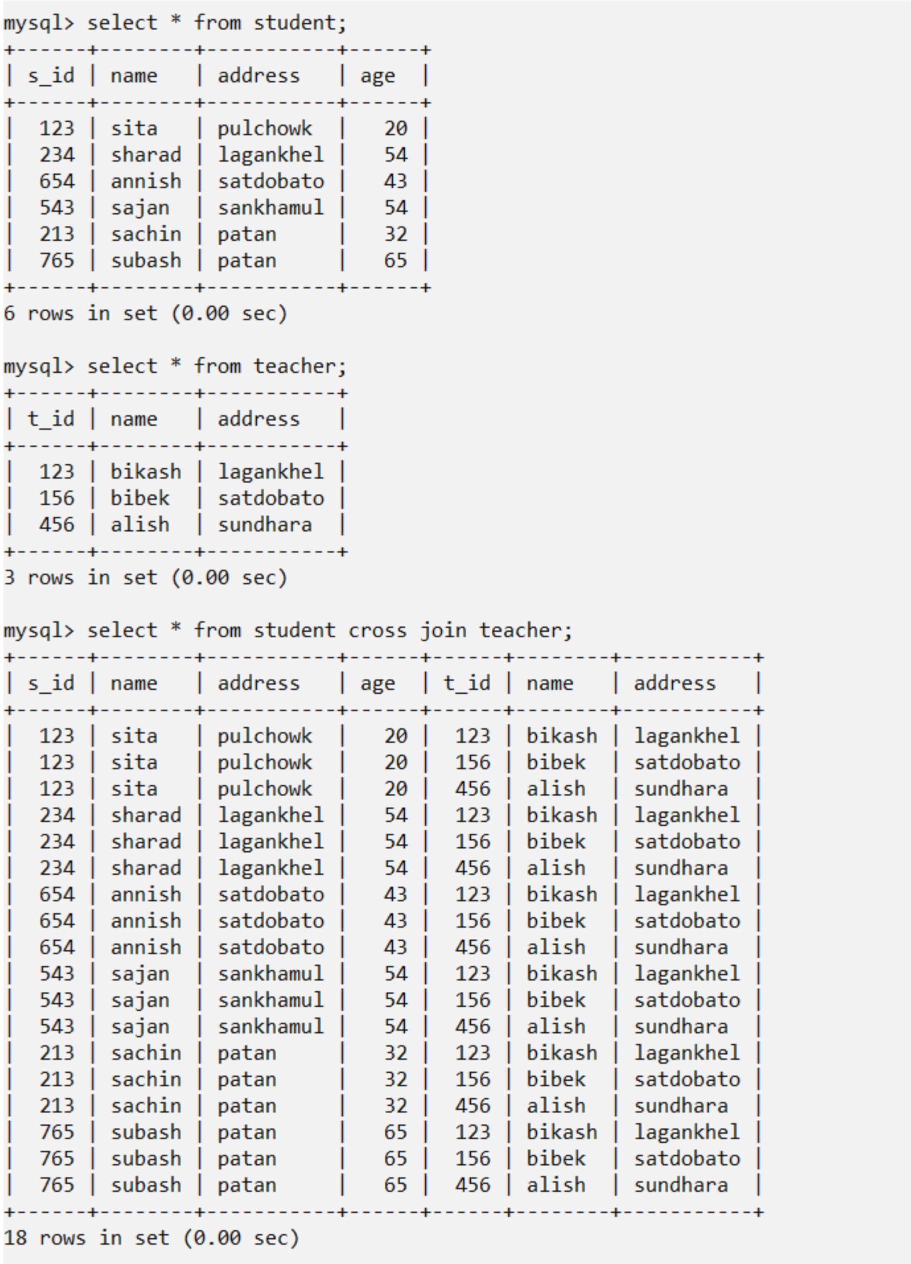
CHAPTER-10: SQL JOIN

10.1 Cross Join

It Returns Cartesian Product Of The Two Tables.

Syntax:

```
Select column_list from table_name1 cross join table_name2;
```



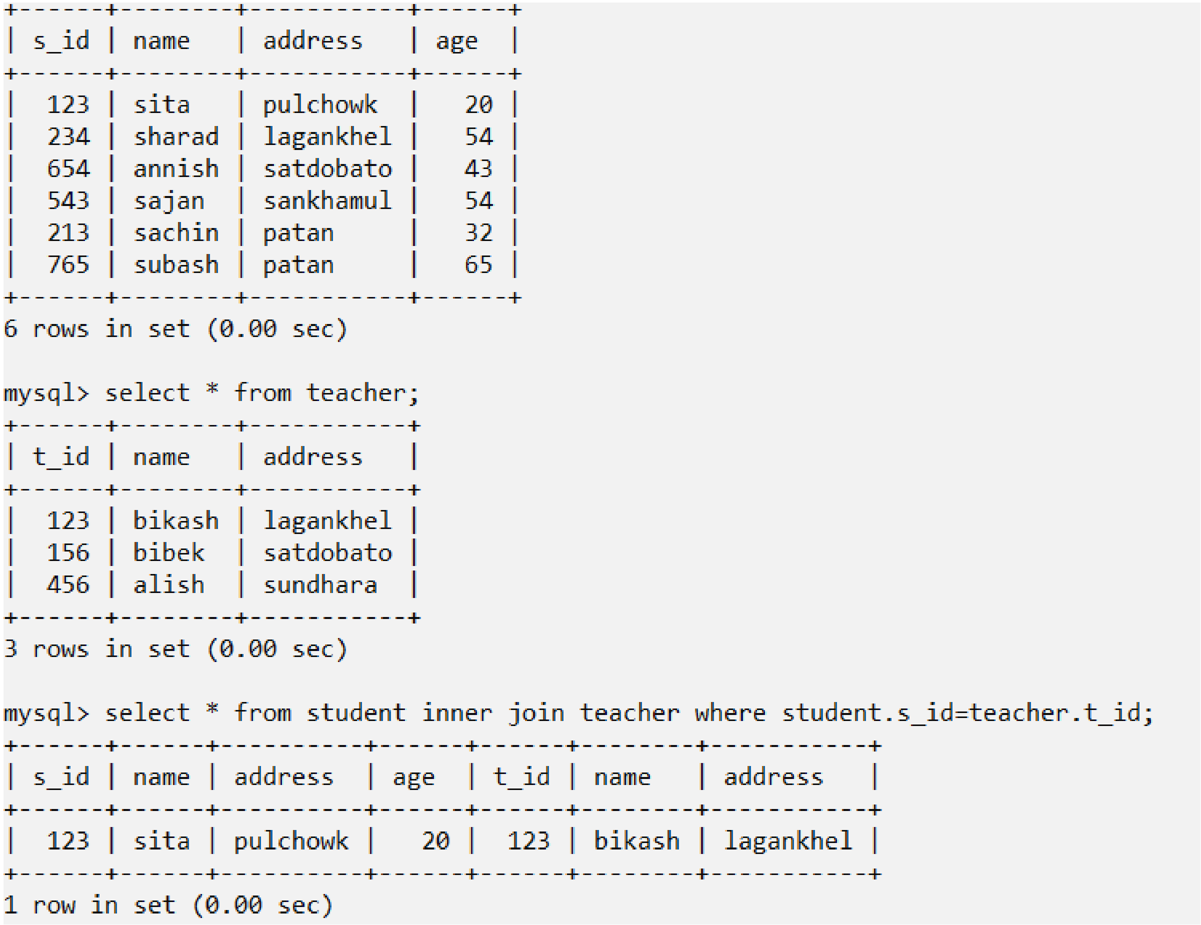
```
mysql> select * from student;
+------+--------+----------+------+
| s_id | name   | address  | age  |
+------+--------+----------+------+
|  123 | sita   | pulchowk |   20 |
|  234 | sharad | lagankhel|   54 |
|  654 | annish | satdobato|   43 |
|  543 | sajan  | sankhamul|   54 |
|  213 | sachin | patan    |   32 |
|  765 | subash | patan    |   65 |
+------+--------+----------+------+
6 rows in set (0.00 sec)

mysql> select * from teacher;
+------+--------+-----------+
| t_id | name   | address   |
+------+--------+-----------+
|  123 | bikash | lagankhel |
|  156 | bibek  | satdobato |
|  456 | alish  | sundhara  |
+------+--------+-----------+
3 rows in set (0.00 sec)

mysql> select * from student cross join teacher;
+------+--------+-----------+------+------+--------+-----------+
| s_id | name   | address   | age  | t_id | name   | address   |
+------+--------+-----------+------+------+--------+-----------+
|  123 | sita   | pulchowk  |   20 |  123 | bikash | lagankhel |
|  123 | sita   | pulchowk  |   20 |  156 | bibek  | satdobato |
|  123 | sita   | pulchowk  |   20 |  456 | alish  | sundhara  |
|  234 | sharad | lagankhel |   54 |  123 | bikash | lagankhel |
|  234 | sharad | lagankhel |   54 |  156 | bibek  | satdobato |
|  234 | sharad | lagankhel |   54 |  456 | alish  | sundhara  |
|  654 | annish | satdobato |   43 |  123 | bikash | lagankhel |
|  654 | annish | satdobato |   43 |  156 | bibek  | satdobato |
|  654 | annish | satdobato |   43 |  456 | alish  | sundhara  |
|  543 | sajan  | sankhamul |   54 |  123 | bikash | lagankhel |
|  543 | sajan  | sankhamul |   54 |  156 | bibek  | satdobato |
|  543 | sajan  | sankhamul |   54 |  456 | alish  | sundhara  |
|  213 | sachin | patan     |   32 |  123 | bikash | lagankhel |
|  213 | sachin | patan     |   32 |  156 | bibek  | satdobato |
|  213 | sachin | patan     |   32 |  456 | alish  | sundhara  |
|  765 | subash | patan     |   65 |  123 | bikash | lagankhel |
|  765 | subash | patan     |   65 |  156 | bibek  | satdobato |
|  765 | subash | patan     |   65 |  456 | alish  | sundhara  |
+------+--------+-----------+------+------+--------+-----------+
18 rows in set (0.00 sec)
```

10.2 Inner Join

It Returns On The Basis Of Matched Values Of The Attributes Chosen As Basis For The Join.

Syntax:

```
Select column_name_list from table_name1 inner join
table_name2 where table_name1 column_name table=table_name2
column_name;
```



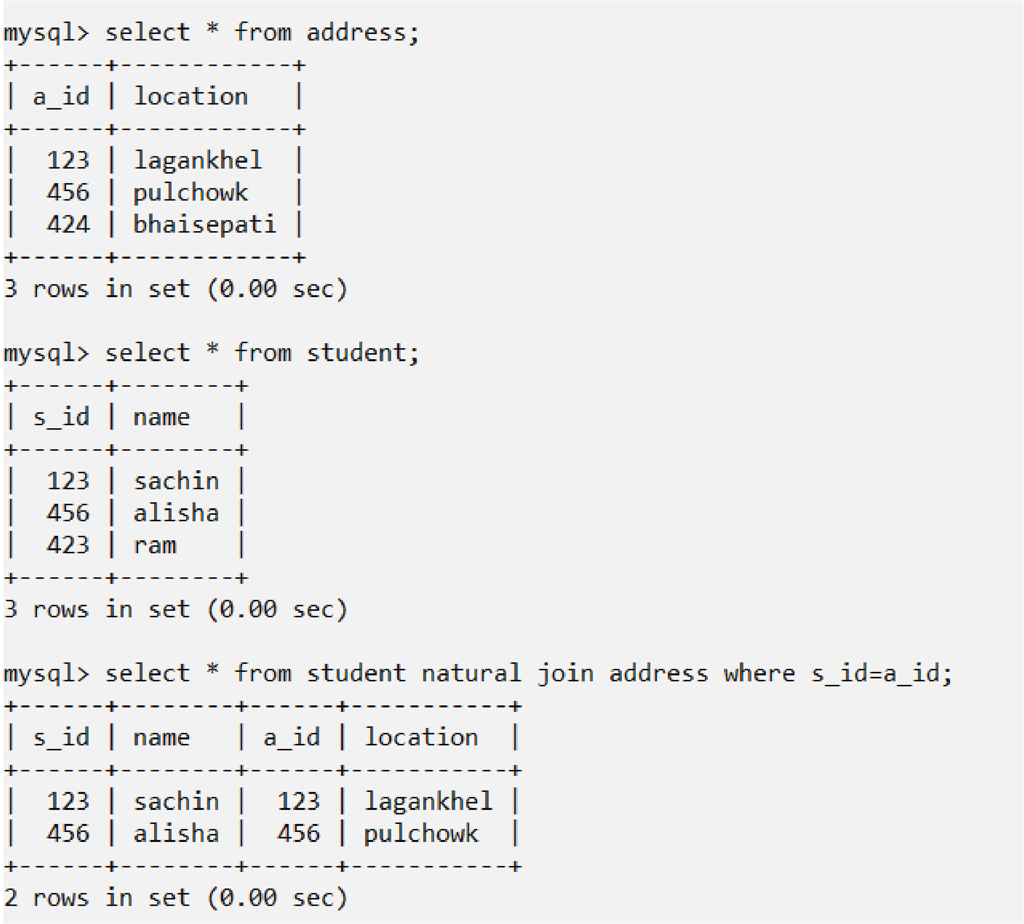
```
+------+--------+-----------+------+
| s_id | name   | address   | age  |
+------+--------+-----------+------+
|  123 | sita   | pulchowk  |   20 |
|  234 | sharad | lagankhel |   54 |
|  654 | annish | satdobato |   43 |
|  543 | sajan  | sankhamul |   54 |
|  213 | sachin | patan     |   32 |
|  765 | subash | patan     |   65 |
+------+--------+-----------+------+
6 rows in set (0.00 sec)

mysql> select * from teacher;
+------+--------+-----------+
| t_id | name   | address   |
+------+--------+-----------+
|  123 | bikash | lagankhel |
|  156 | bibek  | satdobato |
|  456 | alish  | sundhara  |
+------+--------+-----------+
3 rows in set (0.00 sec)

mysql> select * from student inner join teacher where student.s_id=teacher.t_id;
+------+------+----------+------+------+--------+-----------+
| s_id | name | address  | age  | t_id | name   | address   |
+------+------+----------+------+------+--------+-----------+
|  123 | sita | pulchowk |   20 |  123 | bikash | lagankhel |
+------+------+----------+------+------+--------+-----------+
1 row in set (0.00 sec)
```

10.3 Natural Join

The Natural Join Is A Type Of Join Operations That Creates An Implicit Join By Combining Tables Based On Columns With Same Name And Datatype.

Syntax:

Select * from table1 natural join table2 where table1 attribute=table2 attribute;



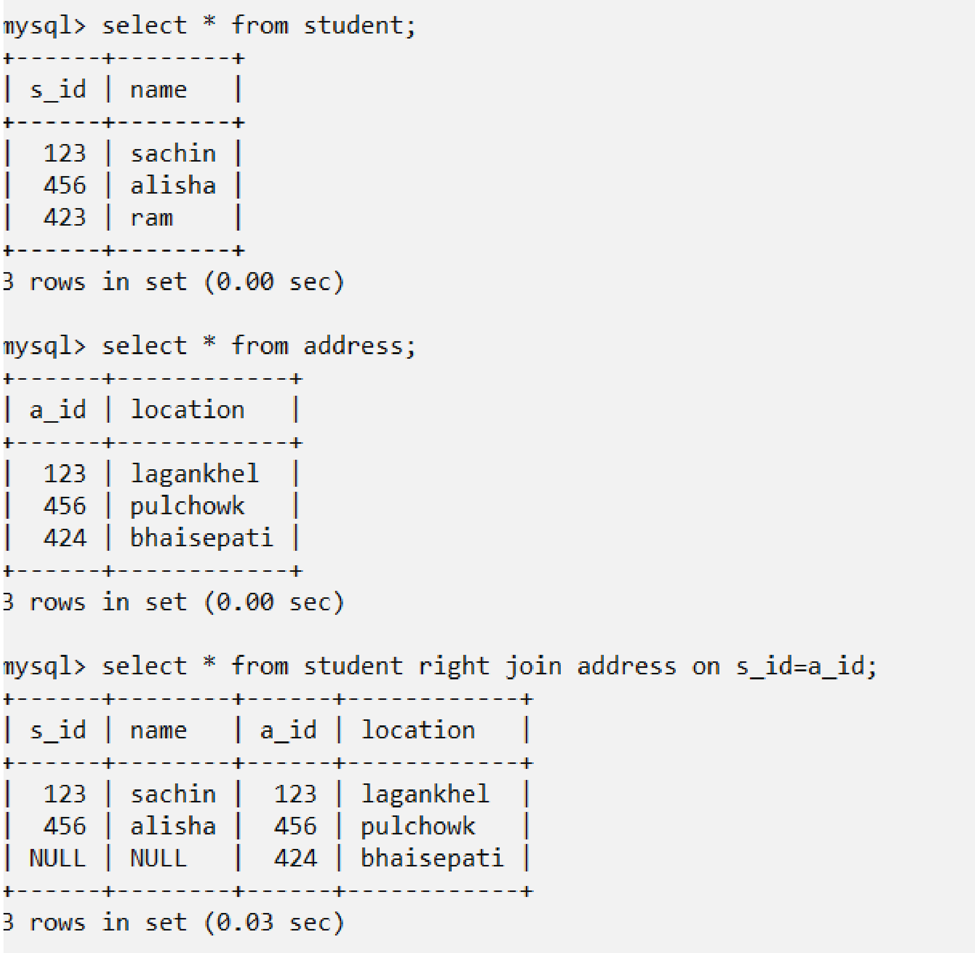
```
mysql> select * from address;
+------+------------+
| a_id | location   |
+------+------------+
|  123 | lagankhel  |
|  456 | pulchowk   |
|  424 | bhaisepati |
+------+------------+
3 rows in set (0.00 sec)

mysql> select * from student;
+------+--------+
| s_id | name   |
+------+--------+
|  123 | sachin |
|  456 | alisha |
|  423 | ram    |
+------+--------+
3 rows in set (0.00 sec)

mysql> select * from student natural join address where s_id=a_id;
+------+--------+------+-----------+
| s_id | name   | a_id | location  |
+------+--------+------+-----------+
|  123 | sachin |  123 | lagankhel |
|  456 | alisha |  456 | pulchowk  |
+------+--------+------+-----------+
2 rows in set (0.00 sec)
```

10.4 Right Outer Join

The Right Join Is Used To Join Two Or More Tables And Return All Rows From The Right-Hand Table.

Syntax:

```
Select * from table_name1 right join table_name2;
```



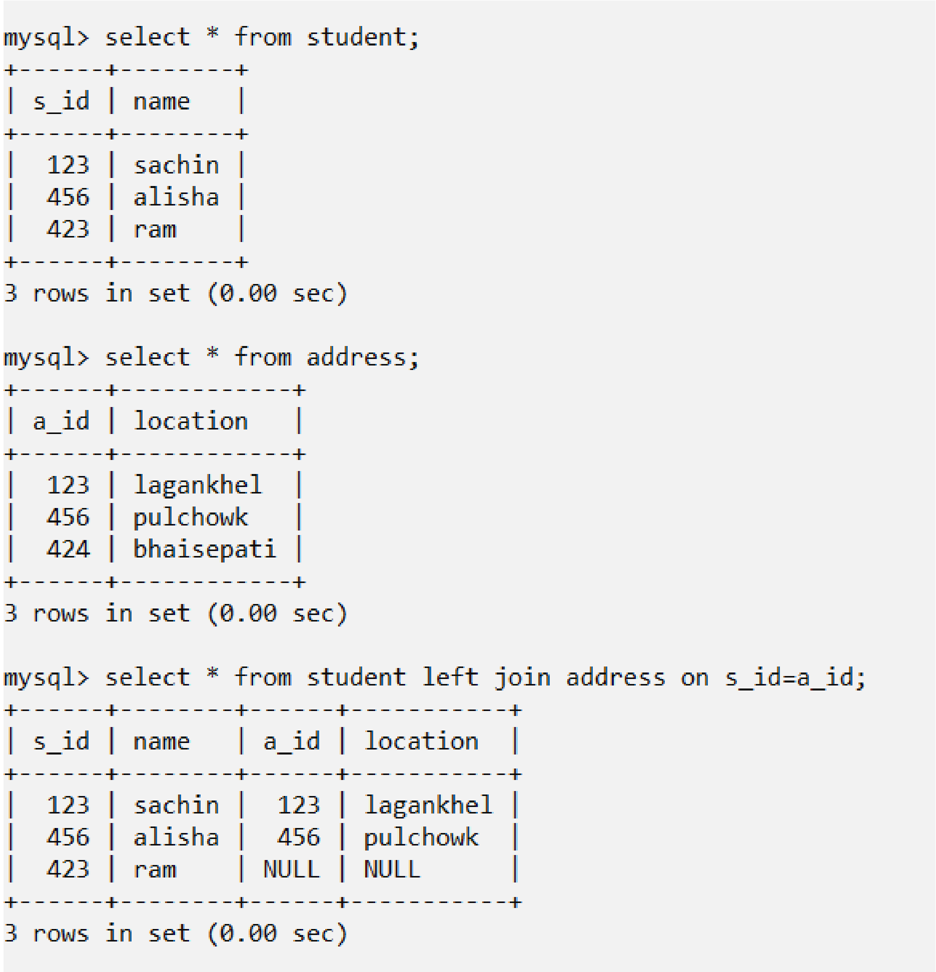
```
mysql> select * from student;
+------+--------+
| s_id | name   |
+------+--------+
|  123 | sachin |
|  456 | alisha |
|  423 | ram    |
+------+--------+
3 rows in set (0.00 sec)

mysql> select * from address;
+------+------------+
| a_id | location   |
+------+------------+
|  123 | lagankhel  |
|  456 | pulchowk   |
|  424 | bhaisepati |
+------+------------+
3 rows in set (0.00 sec)

mysql> select * from student right join address on s_id=a_id;
+------+--------+------+------------+
| s_id | name   | a_id | location   |
+------+--------+------+------------+
|  123 | sachin |  123 | lagankhel  |
|  456 | alisha |  456 | pulchowk   |
| NULL | NULL   |  424 | bhaisepati |
+------+--------+------+------------+
3 rows in set (0.03 sec)
```

10.5 left outer join

The left outer join returns all rows from left-hand.

Syntax:

```
Select * from table_name1 left join table_name2;
```



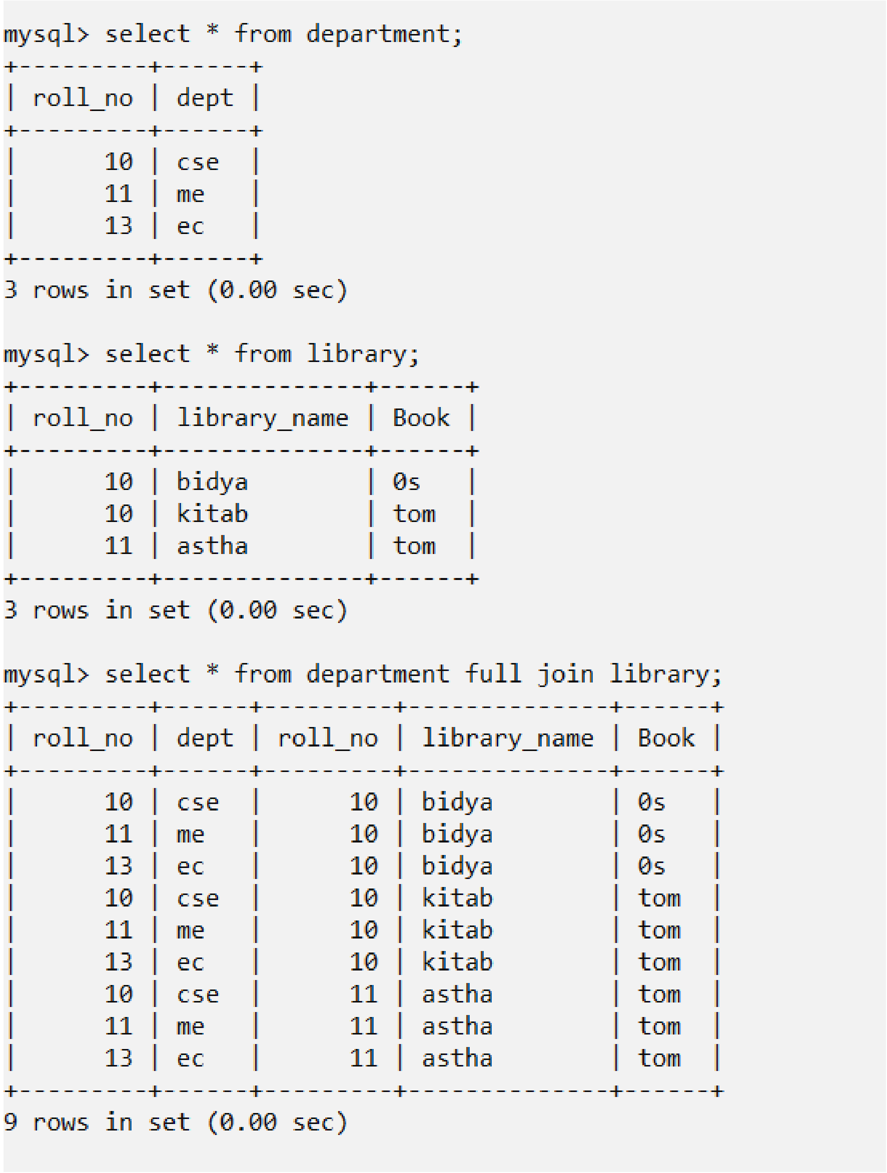
```
mysql> select * from student;
+------+--------+
| s_id | name   |
+------+--------+
|  123 | sachin |
|  456 | alisha |
|  423 | ram    |
+------+--------+
3 rows in set (0.00 sec)

mysql> select * from address;
+------+------------+
| a_id | location   |
+------+------------+
|  123 | lagankhel  |
|  456 | pulchowk   |
|  424 | bhaisepati |
+------+------------+
3 rows in set (0.00 sec)

mysql> select * from student left join address on s_id=a_id;
+------+--------+------+-----------+
| s_id | name   | a_id | location  |
+------+--------+------+-----------+
|  123 | sachin |  123 | lagankhel |
|  456 | alisha |  456 | pulchowk  |
|  423 | ram    | NULL | NULL      |
+------+--------+------+-----------+
3 rows in set (0.00 sec)
```

10.6 Full Outer Join

Full Outer Join Combines The Result Of Both Left And Right Outer Joins And Returns All (Matched Or Unmatched) Rows From Tables On Both Sides Of Join Clause.

Syntax:

```
Select * form table_name 1 full join table_name 2;
```



```
mysql> select * from department;
+---------+------+
| roll_no | dept |
+---------+------+
|      10 | cse  |
|      11 | me   |
|      13 | ec   |
+---------+------+
3 rows in set (0.00 sec)

mysql> select * from library;
+---------+--------------+------+
| roll_no | library_name | Book |
+---------+--------------+------+
|      10 | bidya        | 0s   |
|      10 | kitab        | tom  |
|      11 | astha        | tom  |
+---------+--------------+------+
3 rows in set (0.00 sec)

mysql> select * from department full join library;
+---------+------+---------+--------------+------+
| roll_no | dept | roll_no | library_name | Book |
+---------+------+---------+--------------+------+
|      10 | cse  |      10 | bidya        | 0s   |
|      11 | me   |      10 | bidya        | 0s   |
|      13 | ec   |      10 | bidya        | 0s   |
|      10 | cse  |      10 | kitab        | tom  |
|      11 | me   |      10 | kitab        | tom  |
|      13 | ec   |      10 | kitab        | tom  |
|      10 | cse  |      11 | astha        | tom  |
|      11 | me   |      11 | astha        | tom  |
|      13 | ec   |      11 | astha        | tom  |
+---------+------+---------+--------------+------+
9 rows in set (0.00 sec)
```