	Bansilal Ramnath Agarwal Charitable Trust's Vishwakarma Institute of Information Technology **Department of Artificial Intelligence and Data Science**

Student Name: **Siddhesh Dilip Khairnar**		
Class: **SY**	Division: **B**	Roll No:**272028**
Semester: **III**	Academic Year:**2022-23**	
Subject Name & Code: **DBMS, ADUA21204**		
Title of Assignment: **Design at least 10 SQL queries for suitable database application using SQL DML statements: All types of Joins, Sub-Query and View.**		
Date of Performance: **26-09-2022**	Date of Submission: **30-09-2022**	

Aim: Design at least 10 SQL queries for suitable database application using SQL DML statements: all types of Joins, Sub-Query and View.

Problem Statement:

1. Create table Teaching_Faculty_information containing attributes such as (Faculty_id, Faculty_name, Dept_name) and Subject_information containing attributes such as (Subject_id,Subject_name,Faculty_id) perform below mentioned operations of joins
Inner join

left join
Right join
Full join
Cross Join

2. Create Student table (sid,sname,marks) and execute Sub-Query to calculate second highest marks

3. Create view on Student table (sid,sname,marks) for marks > 60

4. Perform the operation to create Index, Sequence, Synonym by taking suitable example

Background Information: A join clause is used to combine rows from two or more tables, based on a related column between them. A MySQL sub-query is a query nested within another query such as Select, Insert, Update, Delete, etc.

Software Requirements: MYSQL Shell

The Handwritten Write-Up:

PAGE NO.:
DATE: / /

Assignment no. 3

Name: Siddhesh Dilip Khairnar
Division: B Roll no: 272028
PRN no: 22110398
Subject: DBMS

Aim: Design atleast 10 SQL queries for suitable database application using SQL DML statement: All types of joins, sub-Query and view.

Q1 Brief about join and types of joins

Ans SQL join

A join clause is used to combine rows from two or more table, based on a related column between them. My SQL joins are used with select statement. It is used to retrieve data from multiple tables. It is performed whenever you needs to fetch records from two or more tables.

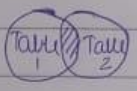
Different types of SQL JOINS

① Inner JOIN

The inner join keyword select record that having matching values in both tables. The INNER JOIN keyword select all rows from both tables as long as there is a match between the columns. It is the most common types of join. In other words, Return record that have matching value in both tables.

Inner Join syntax:

SELECT column_name FROM table-1 INNER JOIN tables 2
ON table 1. column-name = table 2 column_name



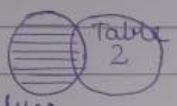
PAGE NO :
DATE : / /

② LEFT JOIN

The left join keyword return all record from left table (table1), and the matching record from right table (table2). The result is record from the right side, if there is no match. The left join keyword return all records from the left table, even if there are no matches in right table. In some databases left join is called left outer join.

Left join syntax

SELECT column name FROM table1 LEFT JOIN table2 ON table1.column name = table2.column_name

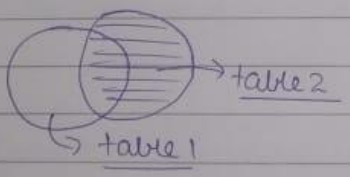


③ RIGHT JOIN

The RIGHT JOIN keyword return all record from the right table & the matching record from the left table. The result is zero record from the left side, if there is no match. In some databases & right join is called right outer join.

Right join syntax:

SELECT column_name FROM table1 RIGHT JOIN table2 ON table1.column_name = table2 column_name.



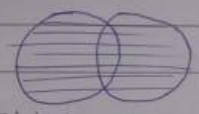
PAGE NO.:
DATE: / /

④ FULL JOIN

The full outer join keyword return all record when there is a match in left table or right table. The full ~~OR~~ outer join and full join are the same. Full outer join can potentially return very large result sets!

FULL JOIN SYNTAX

SELECT column_name FROM table1 FULL outer join table2
ON table1.column_name = table2.column_name



Q2 Brief about Sub-Query

Ans A MySQL sub query is a ~~very~~ query nested within another query such as SELECT, INSERT, UPDATES or DELETE. Also, a subquery can be nested within another subquery. A MySQL subquery is called an inner query while the query that contain the subquery is called an outer query. A subquery can be used anywhere that expression is used and must be closed in parenthesis.

Ex: select Max(Marks) from student where Marks not in (select max(marks) from student);

The above query is to calculate the second highest marks.

Q3 Brief about concept of view, Index, sequence, Synonym.

Ans ① VIEW - In SQL, a view is a virtual table based on the result-set of an SQL statement. A view contain row and column, just like a real

PAGE NO.:
DATE: / /

table. The field in a view are field from one or more real table in the database. You can add SQL statement and function to a view & present the data as if the data were coming from one single table. A view is created with the CREATE view statement.

Syntax: CREATE VIEW view_name as
SELECT column1, column2, ---
FROM table_name.
WHERE condition;

② Index: The CREATE INDEX statement is used to creat indexes in tables. Indexes are used to retrieve data from the ~~to~~ database more quickly ~~than~~ than otherwise. The users cannot see the indexs, they are just used to speed up searches/ queries.

CREATE INDEX SYNTAX:
CREATE INDEX index_name ON table_name (column1, column2...)

CREATE UNIQUE INDEX SYNTAX:
Create UNIQUE index name ON table_name (column1, ...).

③ SEQUENCE: To generate unique numbers for the record in table sequence can be a proper solution, it generate integer value in the ascending order.
creating MYSQL sequence:
Creating sequence is easy in MYSQL, at the time of table creation set the auto_increament for column, most probably the column with primary key.

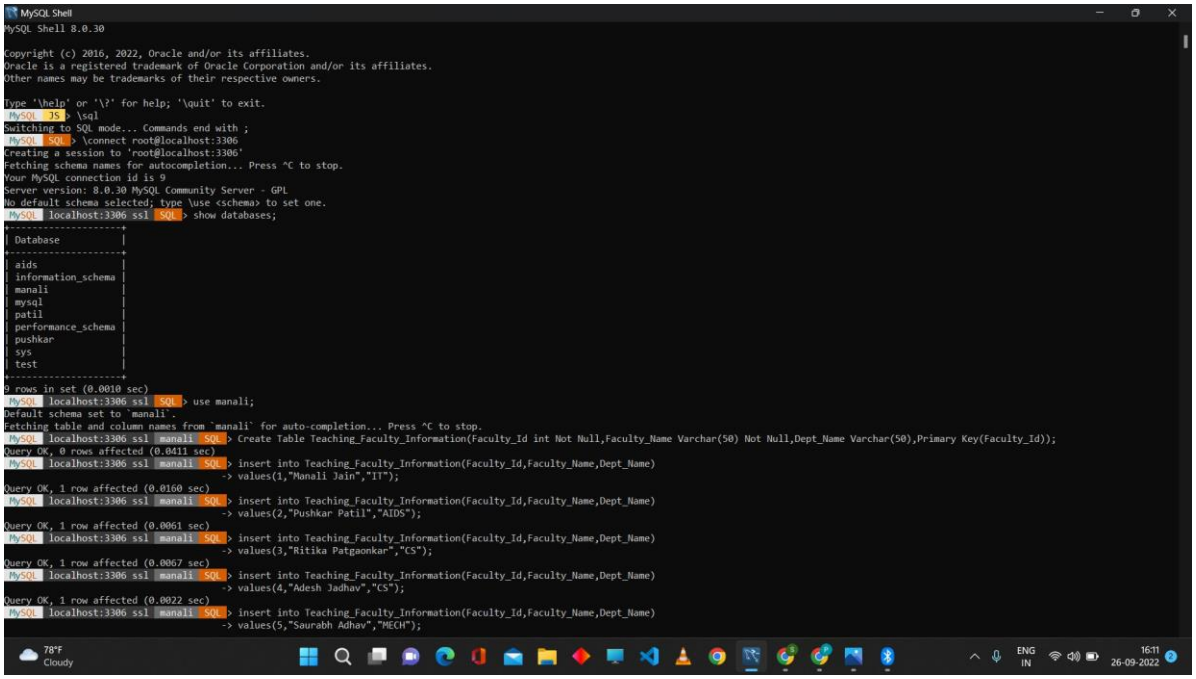
PAGE NO.
DATE: / /

Ex: CREATE TABLE TABLE_NAME(column_name int not null
~~last~~ Auto_Increment,...);

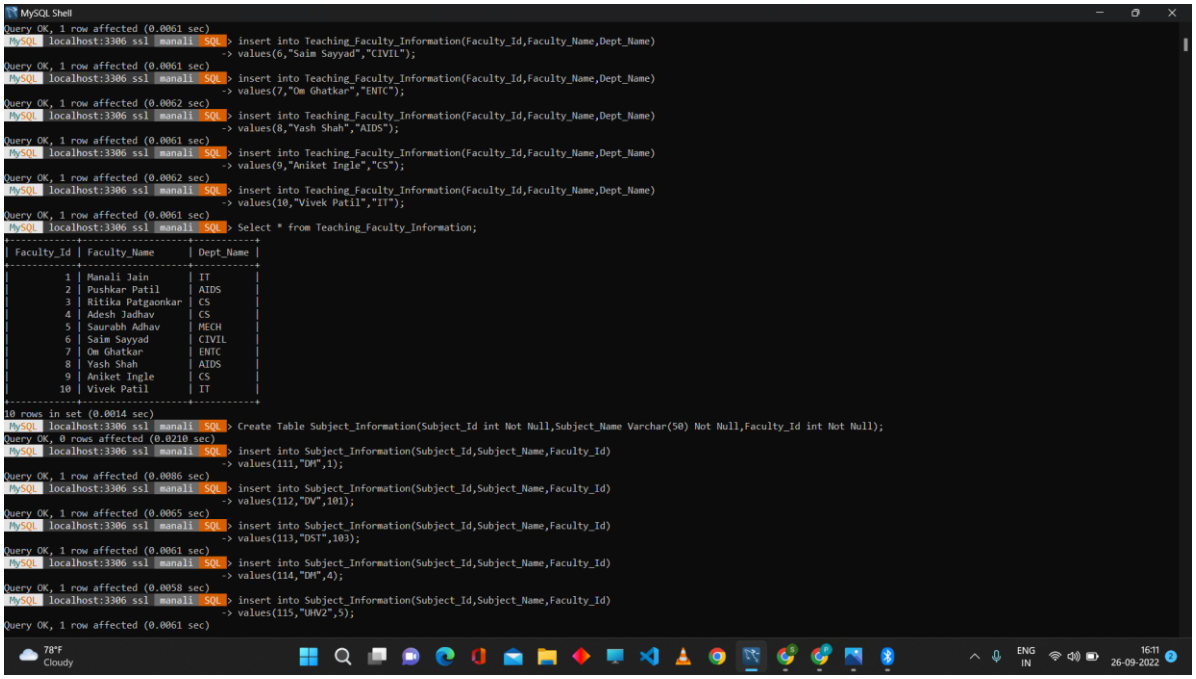
④ SYNONYM: CREATE-SYNONYM_db() Procedure is used to creat Synonym schema, which contain views that refer to all the tables and view in the original schemas.

Ex: CALL sys.create_synonym_db('Demo' 'View_Demo');

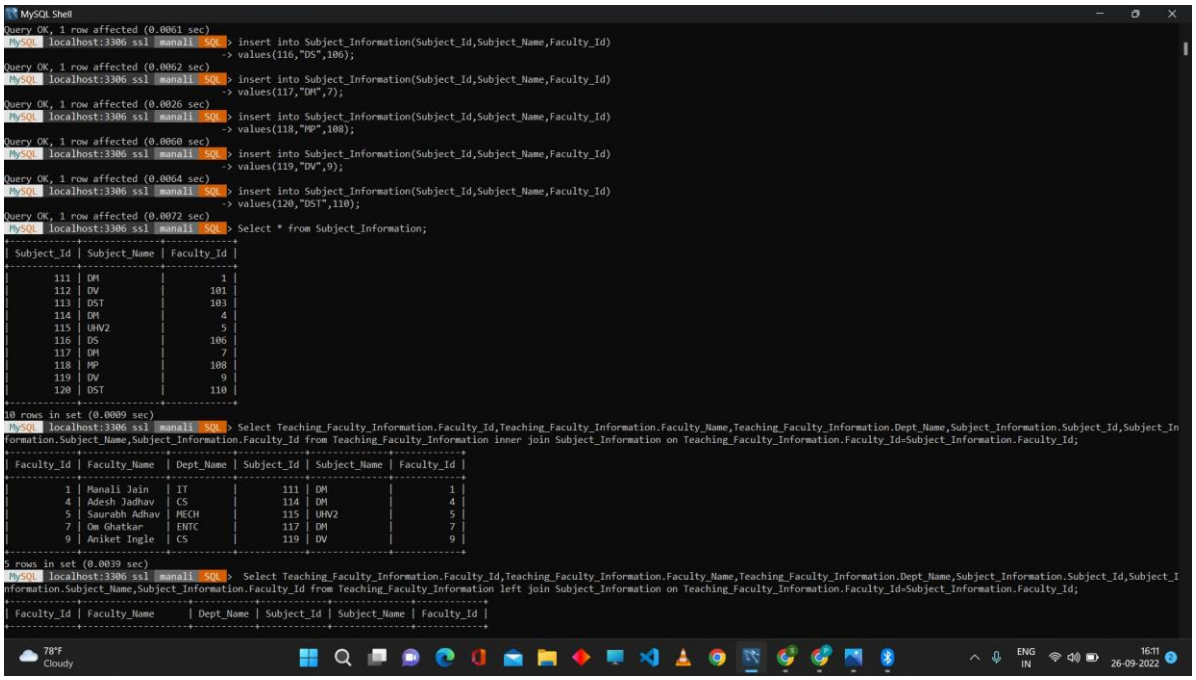
Results or Experimentation:

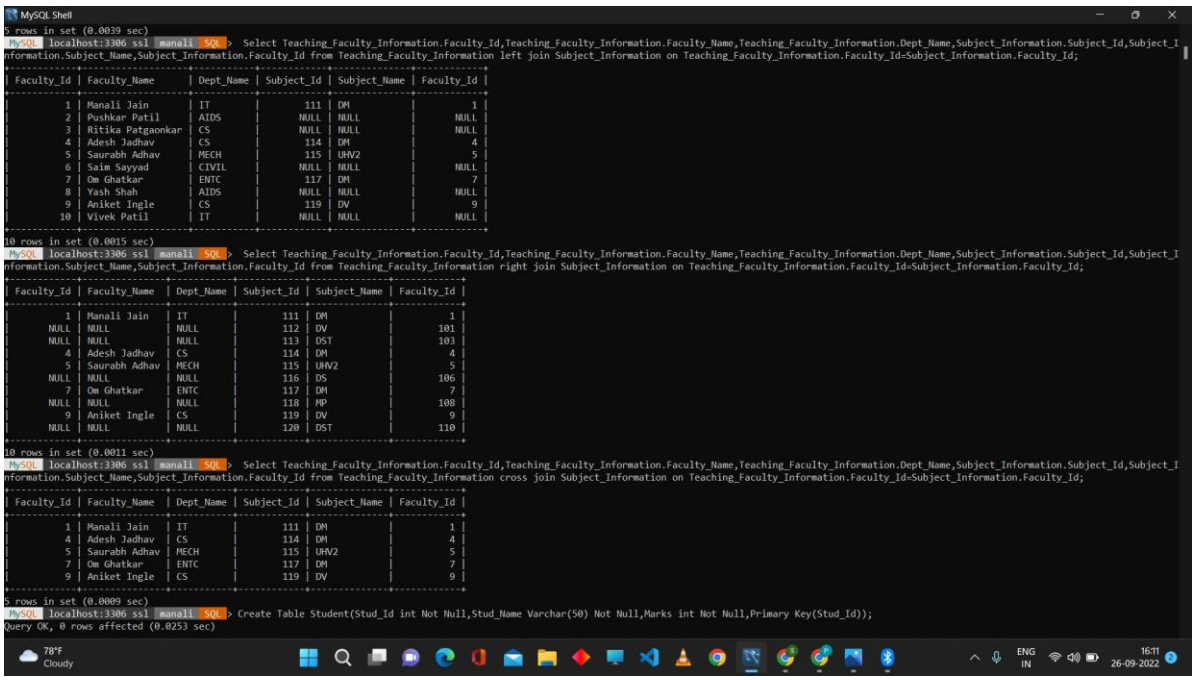


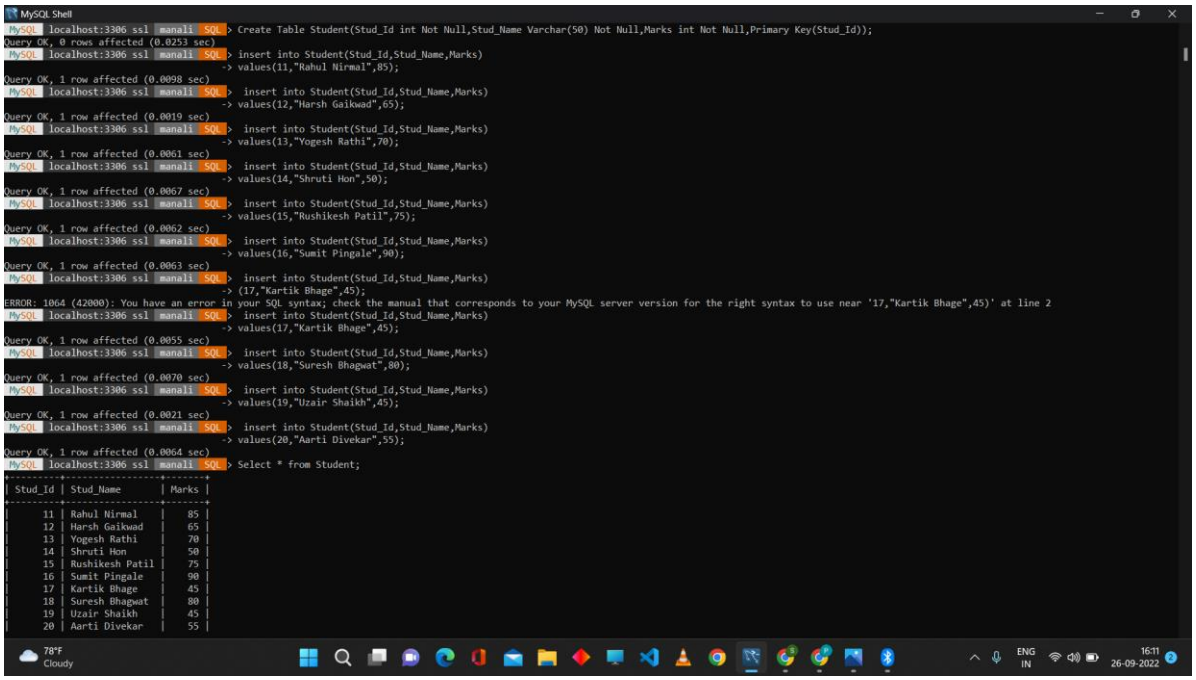
```
MySQL Shell
MySQL Shell 8.0.30

Copyright (c) 2016, 2022, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.

Type '\help' or '\?' for help; '\quit' to exit.
 MySQL  JS > \sql
Switching to SQL mode... Commands end with ;
 MySQL  SQL > \connect root@localhost:3306
Creating a session to 'root@localhost:3306'
Fetching schema names for autocompletion... Press ^C to stop.
Your MySQL connection id is 9
Server version: 8.0.30 MySQL Community Server - GPL
No default schema selected; type \use <schema> to set one.
 MySQL  localhost:3306 ssl  SQL > show databases;
+--------------------+
| Database           |
+--------------------+
| aids               |
| information_schema |
| manali             |
| mysql              |
| patil              |
| performance_schema |
| pushkar            |
| sys                |
| test               |
+--------------------+
9 rows in set (0.0010 sec)
 MySQL  localhost:3306 ssl  SQL > use manali;
Default schema set to `manali`.
Fetching table and column names from `manali` for auto-completion... Press ^C to stop.
 MySQL  localhost:3306 ssl  manali  SQL > Create Table Teaching_Faculty_Information(Faculty_Id int Not Null,Faculty_Name Varchar(50) Not Null,Dept_Name Varchar(50),Primary Key(Faculty_Id));
Query OK, 0 rows affected (0.0411 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(1,"Manali Jain","IT");
Query OK, 1 row affected (0.0160 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(2,"Pushkar Patil","AIDS");
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(3,"Ritika Patgaonkar","CS");
Query OK, 1 row affected (0.0067 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(4,"Adesh Jadhav","CS");
Query OK, 1 row affected (0.0022 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(5,"Saurabh Adhav","MECH");
```

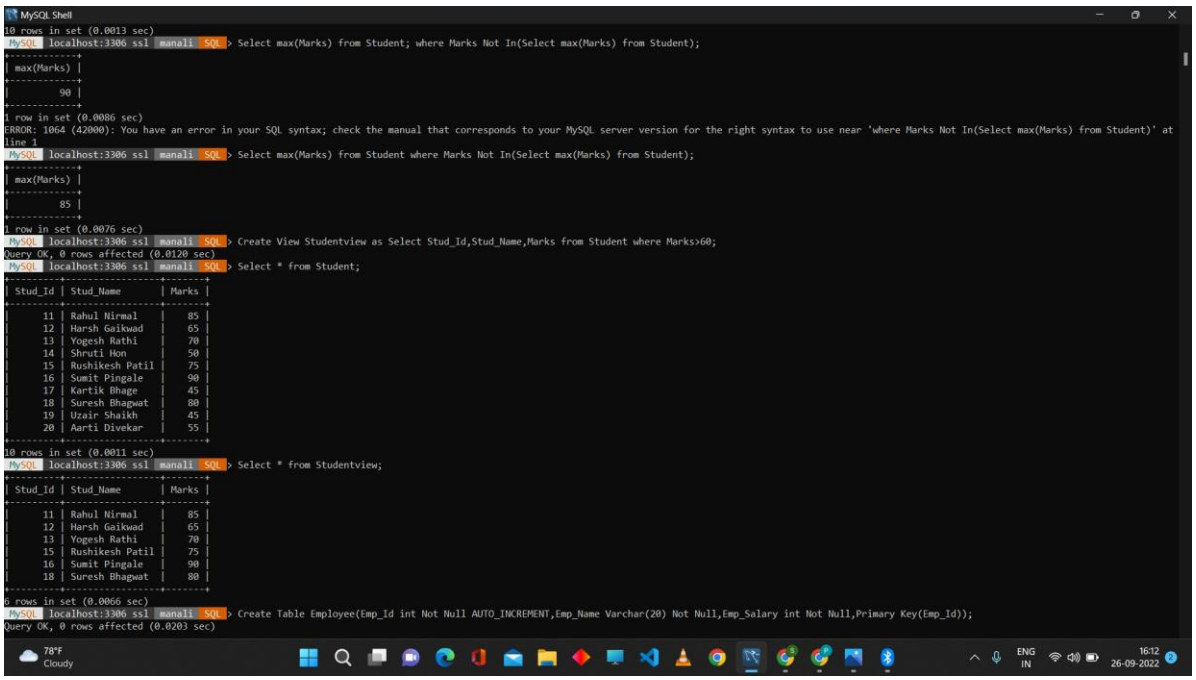


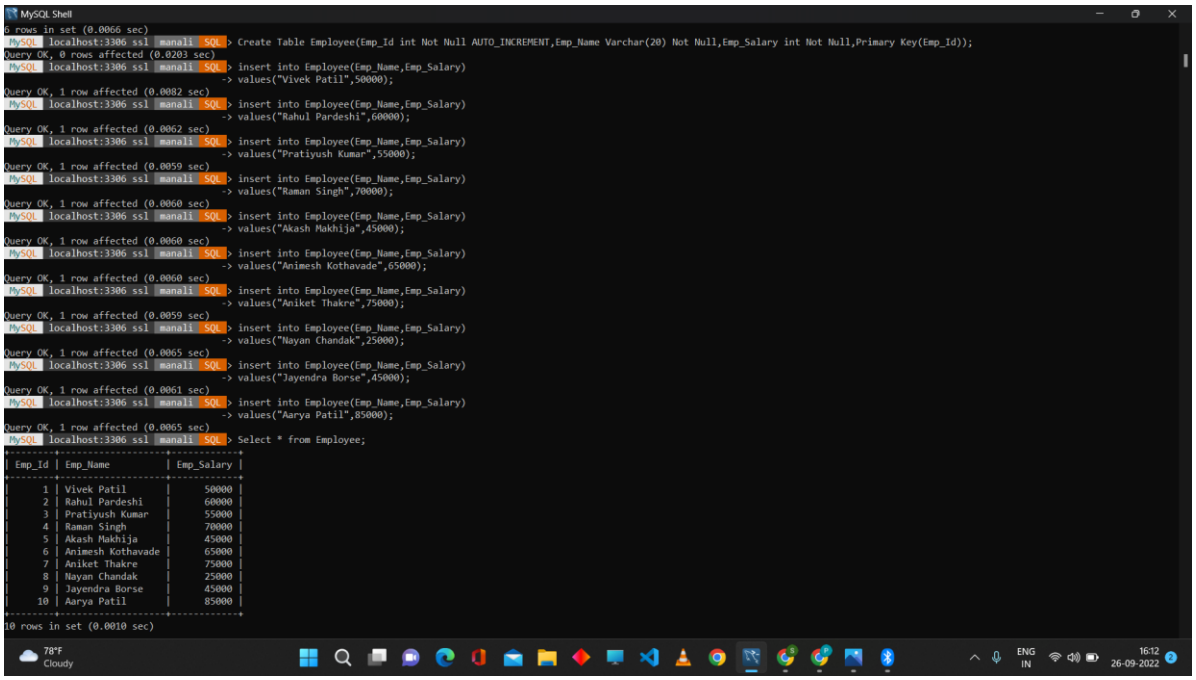
```
MySQL Shell
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(6,"Saim Sayyad","CIVIL");
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(7,"Om Ghatkar","ENTC");
Query OK, 1 row affected (0.0062 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(8,"Yash Shah","AIDS");
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(9,"Aniket Ingle","CS");
Query OK, 1 row affected (0.0062 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Teaching_Faculty_Information(Faculty_Id,Faculty_Name,Dept_Name)
                                    -> values(10,"Vivek Patil","IT");
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > Select * from Teaching_Faculty_Information;
+------------+------------------+-----------+
| Faculty_Id | Faculty_Name     | Dept_Name |
+------------+------------------+-----------+
|          1 | Manali Jain      | IT        |
|          2 | Pushkar Patil    | AIDS      |
|          3 | Ritika Patgaonkar | CS        |
|          4 | Adesh Jadhav     | CS        |
|          5 | Saurabh Adhav    | MECH      |
|          6 | Saim Sayyad      | CIVIL     |
|          7 | Om Ghatkar       | ENTC      |
|          8 | Yash Shah        | AIDS      |
|          9 | Aniket Ingle     | CS        |
|         10 | Vivek Patil      | IT        |
+------------+------------------+-----------+
10 rows in set (0.0014 sec)
 MySQL  localhost:3306 ssl  manali  SQL > Create Table Subject_Information(Subject_Id int Not Null,Subject_Name Varchar(50) Not Null,Faculty_Id int Not Null);
Query OK, 0 rows affected (0.0210 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
                                    -> values(111,"DM",1);
Query OK, 1 row affected (0.0086 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
                                    -> values(112,"DV",101);
Query OK, 1 row affected (0.0065 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
                                    -> values(113,"DST",103);
Query OK, 1 row affected (0.0061 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
                                    -> values(114,"DM",4);
Query OK, 1 row affected (0.0058 sec)
 MySQL  localhost:3306 ssl  manali  SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
                                    -> values(115,"UHV2",5);
Query OK, 1 row affected (0.0061 sec)
```

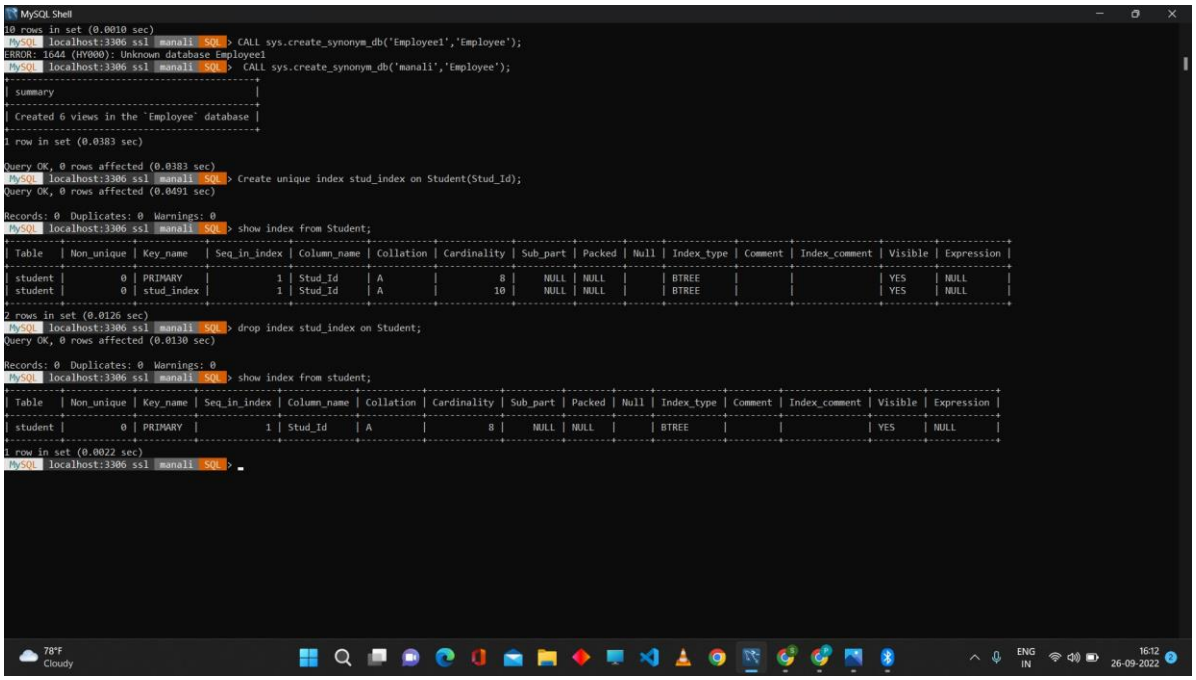

MySQL Shell
Query OK, 1 row affected (0.0061 sec)
MySQL localhost:3306 ssl manali SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
-> values(116,"DS",106);
Query OK, 1 row affected (0.0062 sec)
MySQL localhost:3306 ssl manali SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
-> values(117,"DM",7);
Query OK, 1 row affected (0.0026 sec)
MySQL localhost:3306 ssl manali SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
-> values(118,"MP",108);
Query OK, 1 row affected (0.0060 sec)
MySQL localhost:3306 ssl manali SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
-> values(119,"DV",9);
Query OK, 1 row affected (0.0064 sec)
MySQL localhost:3306 ssl manali SQL > insert into Subject_Information(Subject_Id,Subject_Name,Faculty_Id)
-> values(120,"DST",110);
Query OK, 1 row affected (0.0072 sec)
MySQL localhost:3306 ssl manali SQL > Select * from Subject_Information;
Subject_Id	Subject_Name	Faculty_Id
111	DM	1
112	DV	101
113	DST	103
114	DM	4
115	UHV2	5
116	DS	106
117	DM	7
118	MP	108
119	DV	9
120	DST	110
10 rows in set (0.0009 sec)
MySQL localhost:3306 ssl manali SQL > Select Teaching_Faculty_Information.Faculty_Id,Teaching_Faculty_Information.Faculty_Name,Teaching_Faculty_Information.Dept_Name,Subject_Information.Subject_Id,Subject_Information.Subject_Name,Subject_Information.Faculty_Id from Teaching_Faculty_Information inner join Subject_Information on Teaching_Faculty_Information.Faculty_Id=Subject_Information.Faculty_Id;
Faculty_Id	Faculty_Name	Dept_Name	Subject_Id	Subject_Name	Faculty_Id
1	Manali Jain	IT	111	DM	1
4	Adesh Jadhav	CS	114	DM	4
5	Saurabh Adhav	MECH	115	UHV2	5
7	Om Ghatkar	ENTC	117	DM	7
9	Aniket Ingle	CS	119	DV	9
5 rows in set (0.0039 sec)
MySQL localhost:3306 ssl manali SQL > Select Teaching_Faculty_Information.Faculty_Id,Teaching_Faculty_Information.Faculty_Name,Teaching_Faculty_Information.Dept_Name,Subject_Information.Subject_Id,Subject_Information.Subject_Name,Subject_Information.Faculty_Id from Teaching_Faculty_Information left join Subject_Information on Teaching_Faculty_Information.Faculty_Id=Subject_Information.Faculty_Id;
Faculty_Id	Faculty_Name	Dept_Name	Subject_Id	Subject_Name	Faculty_Id
78°F
Cloudy
ENG
IN
16:11
26-09-2022


MySQL Shell
5 rows in set (0.0039 sec)
MySQL localhost:3306 ssl manali SQL > Select Teaching_Faculty_Information.Faculty_Id,Teaching_Faculty_Information.Faculty_Name,Teaching_Faculty_Information.Dept_Name,Subject_Information.Subject_Id,Subject_Information.Subject_Name,Subject_Information.Faculty_Id from Teaching_Faculty_Information left join Subject_Information on Teaching_Faculty_Information.Faculty_Id=Subject_Information.Faculty_Id;
Faculty_Id	Faculty_Name	Dept_Name	Subject_Id	Subject_Name	Faculty_Id
1	Manali Jain	IT	111	DM	1
2	Pushkar Patil	AIDS	NULL	NULL	NULL
3	Ritika Patgaonkar	CS	NULL	NULL	NULL
4	Adesh Jadhav	CS	114	DM	4
5	Saurabh Adhav	MECH	115	UHV2	5
6	Saim Sayyad	CIVIL	NULL	NULL	NULL
7	Om Ghatkar	ENTC	117	DM	7
8	Yash Shah	AIDS	NULL	NULL	NULL
9	Aniket Ingle	CS	119	DV	9
10	Vivek Patil	IT	NULL	NULL	NULL
10 rows in set (0.0015 sec)
MySQL localhost:3306 ssl manali SQL > Select Teaching_Faculty_Information.Faculty_Id,Teaching_Faculty_Information.Faculty_Name,Teaching_Faculty_Information.Dept_Name,Subject_Information.Subject_Id,Subject_Information.Subject_Name,Subject_Information.Faculty_Id from Teaching_Faculty_Information right join Subject_Information on Teaching_Faculty_Information.Faculty_Id=Subject_Information.Faculty_Id;
Faculty_Id	Faculty_Name	Dept_Name	Subject_Id	Subject_Name	Faculty_Id
1	Manali Jain	IT	111	DM	1
NULL	NULL	NULL	112	DV	101
NULL	NULL	NULL	113	DST	103
4	Adesh Jadhav	CS	114	DM	4
5	Saurabh Adhav	MECH	115	UHV2	5
NULL	NULL	NULL	116	DS	106
7	Om Ghatkar	ENTC	117	DM	7
NULL	NULL	NULL	118	MP	108
9	Aniket Ingle	CS	119	DV	9
NULL	NULL	NULL	120	DST	110
10 rows in set (0.0011 sec)
MySQL localhost:3306 ssl manali SQL > Select Teaching_Faculty_Information.Faculty_Id,Teaching_Faculty_Information.Faculty_Name,Teaching_Faculty_Information.Dept_Name,Subject_Information.Subject_Id,Subject_Information.Subject_Name,Subject_Information.Faculty_Id from Teaching_Faculty_Information cross join Subject_Information on Teaching_Faculty_Information.Faculty_Id=Subject_Information.Faculty_Id;
Faculty_Id	Faculty_Name	Dept_Name	Subject_Id	Subject_Name	Faculty_Id
1	Manali Jain	IT	111	DM	1
4	Adesh Jadhav	CS	114	DM	4
5	Saurabh Adhav	MECH	115	UHV2	5
7	Om Ghatkar	ENTC	117	DM	7
9	Aniket Ingle	CS	119	DV	9
5 rows in set (0.0009 sec)
MySQL localhost:3306 ssl manali SQL > Create Table Student(Stud_Id int Not Null,Stud_Name Varchar(50) Not Null,Marks int Not Null,Primary Key(Stud_Id));
Query OK, 0 rows affected (0.0253 sec)
78°F
Cloudy
ENG
IN
16:11
26-09-2022


MySQL Shell
MySQL localhost:3306 ssl manali SQL > Create Table Student(Stud_Id int Not Null,Stud_Name Varchar(50) Not Null,Marks int Not Null,Primary Key(Stud_Id));
Query OK, 0 rows affected (0.0253 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(11,"Rahul Nirmal",85);
Query OK, 1 row affected (0.0098 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(12,"Harsh Gaikwad",65);
Query OK, 1 row affected (0.0019 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(13,"Yogesh Rathi",70);
Query OK, 1 row affected (0.0061 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(14,"Shruti Hon",50);
Query OK, 1 row affected (0.0067 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(15,"Rushikesh Patil",75);
Query OK, 1 row affected (0.0062 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(16,"Sumit Pingale",90);
Query OK, 1 row affected (0.0063 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> (17,"Kartik Bhage",45);
ERROR: 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '17,"Kartik Bhage",45)' at line 2
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(17,"Kartik Bhage",45);
Query OK, 1 row affected (0.0055 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(18,"Suresh Bhagwat",80);
Query OK, 1 row affected (0.0070 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(19,"Uzair Shaikh",45);
Query OK, 1 row affected (0.0021 sec)
MySQL localhost:3306 ssl manali SQL > insert into Student(Stud_Id,Stud_Name,Marks)
-> values(20,"Aarti Divekar",55);
Query OK, 1 row affected (0.0064 sec)
MySQL localhost:3306 ssl manali SQL > Select * from Student;
Stud_Id	Stud_Name	Marks
11	Rahul Nirmal	85
12	Harsh Gaikwad	65
13	Yogesh Rathi	70
14	Shruti Hon	50
15	Rushikesh Patil	75
16	Sumit Pingale	90
17	Kartik Bhage	45
18	Suresh Bhagwat	80
19	Uzair Shaikh	45
20	Aarti Divekar	55
78°F
Cloudy
ENG
IN
16:11
26-09-2022


MySQL Shell
10 rows in set (0.0013 sec)
MySQL localhost:3306 ssl manali SQL > Select max(Marks) from Student; where Marks Not In(Select max(Marks) from Student);
| max(Marks) |
| 90 |
1 row in set (0.0006 sec)
ERROR: 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'where Marks Not In(Select max(Marks) from Student)' at line 1
MySQL localhost:3306 ssl manali SQL > Select max(Marks) from Student where Marks Not In(Select max(Marks) from Student);
| max(Marks) |
| 85 |
1 row in set (0.0076 sec)
MySQL localhost:3306 ssl manali SQL > Create View Studentview as Select Stud_Id,Stud_Name,Marks from Student where Marks>60;
Query OK, 0 rows affected (0.0120 sec)
MySQL localhost:3306 ssl manali SQL > Select * from Student;
Stud_Id	Stud_Name	Marks
11	Rahul Nirmal	85
12	Harsh Gaikwad	65
13	Yogesh Rathi	70
14	Shruti Hon	50
15	Rushikesh Patil	75
16	Sumit Pingale	90
17	Kartik Bhage	45
18	Suresh Bhagwat	80
19	Uzair Shaikh	45
20	Aarti Divekar	55
10 rows in set (0.0011 sec)
MySQL localhost:3306 ssl manali SQL > Select * from Studentview;
Stud_Id	Stud_Name	Marks
11	Rahul Nirmal	85
12	Harsh Gaikwad	65
13	Yogesh Rathi	70
15	Rushikesh Patil	75
16	Sumit Pingale	90
18	Suresh Bhagwat	80
6 rows in set (0.0066 sec)
MySQL localhost:3306 ssl manali SQL > Create Table Employee(Emp_Id int Not Null AUTO_INCREMENT,Emp_Name Varchar(20) Not Null,Emp_Salary int Not Null,Primary Key(Emp_Id));
Query OK, 0 rows affected (0.0203 sec)
78°F
Cloudy
ENG
IN
16:12
26-09-2022


MySQL Shell
6 rows in set (0.0066 sec)
MySQL localhost:3306 ssl manali SQL > Create Table Employee(Emp_Id int Not Null AUTO_INCREMENT,Emp_Name Varchar(20) Not Null,Emp_Salary int Not Null,Primary Key(Emp_Id));
Query OK, 0 rows affected (0.0203 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Vivek Patil",50000);
Query OK, 1 row affected (0.0082 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Rahul Pardeshi",60000);
Query OK, 1 row affected (0.0062 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Pratiyush Kumar",55000);
Query OK, 1 row affected (0.0059 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Raman Singh",70000);
Query OK, 1 row affected (0.0060 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Akash Makhija",45000);
Query OK, 1 row affected (0.0060 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Animesh Kothavade",65000);
Query OK, 1 row affected (0.0060 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Aniket Thakre",75000);
Query OK, 1 row affected (0.0059 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Nayan Chandak",25000);
Query OK, 1 row affected (0.0065 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Jayendra Borse",45000);
Query OK, 1 row affected (0.0061 sec)
MySQL localhost:3306 ssl manali SQL > insert into Employee(Emp_Name,Emp_Salary)
-> values("Aarya Patil",85000);
Query OK, 1 row affected (0.0065 sec)
MySQL localhost:3306 ssl manali SQL > Select * from Employee;
+--------+----------------+------------+
| Emp_Id | Emp_Name | Emp_Salary |
+--------+----------------+------------+
1	Vivek Patil	50000
2	Rahul Pardeshi	60000
3	Pratiyush Kumar	55000
4	Raman Singh	70000
5	Akash Makhija	45000
6	Animesh Kothavade	65000
7	Aniket Thakre	75000
8	Nayan Chandak	25000
9	Jayendra Borse	45000
10	Aarya Patil	85000
+--------+----------------+------------+
10 rows in set (0.0010 sec)
78°F
Cloudy
ENG
IN
16:12
26-09-2022


MySQL Shell
10 rows in set (0.0010 sec)
MySQL localhost:3306 ssl manali SQL > CALL sys.create_synonym_db('Employee1','Employee');
ERROR: 1644 (HY000): Unknown database Employee1
MySQL localhost:3306 ssl manali SQL > CALL sys.create_synonym_db('manali','Employee');
+--+
| summary |
+--+
| Created 6 views in the `Employee` database |
+--+
1 row in set (0.0383 sec)
Query OK, 0 rows affected (0.0383 sec)
MySQL localhost:3306 ssl manali SQL > Create unique index stud_index on Student(Stud_Id);
Query OK, 0 rows affected (0.0491 sec)
Records: 0 Duplicates: 0 Warnings: 0
MySQL localhost:3306 ssl manali SQL > show index from Student;
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| student | 0 | PRIMARY | 1 | Stud_Id | A | 8 | NULL | NULL | | BTREE | | | YES | NULL |
| student | 0 | stud_index | 1 | Stud_Id | A | 10 | NULL | NULL | | BTREE | | | YES | NULL |
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
2 rows in set (0.0126 sec)
MySQL localhost:3306 ssl manali SQL > drop index stud_index on Student;
Query OK, 0 rows affected (0.0130 sec)
Records: 0 Duplicates: 0 Warnings: 0
MySQL localhost:3306 ssl manali SQL > show index from student;
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| student | 0 | PRIMARY | 1 | Stud_Id | A | 8 | NULL | NULL | | BTREE | | | YES | NULL |
+---------+------------+------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
1 row in set (0.0022 sec)
MySQL localhost:3306 ssl manali SQL >
78°F
Cloudy
ENG
IN
16:12
26-09-2022

Conclusion: Thus, we have successfully written and executed join operations such as Inner Join, Left Join, Right Join, Full Join etc. Sub-query for second highest Marks and also created queries for VIEW, INDEX, SEQUENCE, SYNONYM.