College Code : SDES
SREE DATTHA INSTITUTE OF ENGINEERING & SCIENCE
(Autonomous, NAAC A+, NBA Accreditated, Affiliated to JNTUH, Approved by AICTE, New Delhi)
Nagarjuna Sagar Road, Sheriguda (V), Ibrahimpatnam (M), R.R. Dist., Greater Hyderabad, Telangana-501510.
Website: www.sreedattha.ac.in Email: info@sreedattha.ac.in

Department of Computer Science and Engineering

LAB Notes

on

Database Management System Laboratory

2nd Year 2nd Semester

by

Prof. Puneet Shetteppanavar

Assistant Professor, Computer Science and Engineering Department,
Sree Dattha Institute of Engineering and Science, Sheriguda, Hyderabad - 501510

DATABASE MANAGEMENT SYSTEMS LAB

B.Tech. II Year II Sem.

L	T	P	C
0	0	2	1

Co-requisites: "Database Management Systems"

Course Objectives:

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Course Outcomes:

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

List of Experiments:

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A. Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)
   B. Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

TEXT BOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill,3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7thEdition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C.J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

Note: {Previous Hand Written Notes will be Added Soon}

MySQL View

A view is a database object that has no values. Its contents are based on the base table. It contains rows and columns similar to the real table. In MySQL, the View is a **virtual table** created by a query by joining one or more tables. It is operated similarly to the base table but does not contain any data of its own. The View and table have one main difference that the views are definitions built on top of other tables (or views). If any changes occur in the underlying table, the same changes reflected in the View also.

MySQL allows us to create a view in mainly two ways:

1. MySQL Command line client
2. MySQL Workbench

Let us discuss both in detail.

MySQL Command Line Client

We can create a new view by using the **CREATE VIEW** and **SELECT** statement. SELECT statements are used to take data from the source table to make a VIEW.

Syntax

Following is the syntax to create a view in MySQL:

```
CREATE [OR REPLACE] VIEW view_name AS
SELECT columns
FROM tables
[WHERE conditions];
```

Parameters:

The view syntax contains the following parameters:

OR REPLACE: It is optional. It is used when a VIEW already exists. If you do not specify this clause and the VIEW already exists, the CREATE VIEW statement will return an error.

view_name: It specifies the name of the VIEW that you want to create in MySQL.

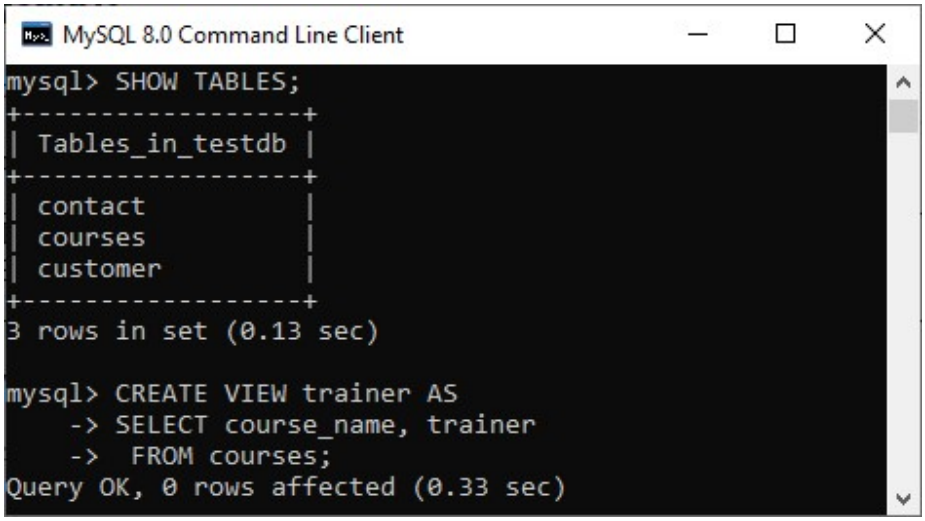
WHERE conditions: It is also optional. It specifies the conditions that must be met for the records to be included in the VIEW.

Example

Let us understand it with the help of an example. Suppose our database has a table **course**, and we are going to create a view based on this table. Thus, the below example will create a VIEW name "**trainer**" that creates a virtual table made by taking data from the table courses.

```
CREATE VIEW trainer AS
SELECT course_name, trainer
FROM courses;
```

Once the execution of the CREATE VIEW statement becomes successful, MySQL will create a view and stores it in the database.



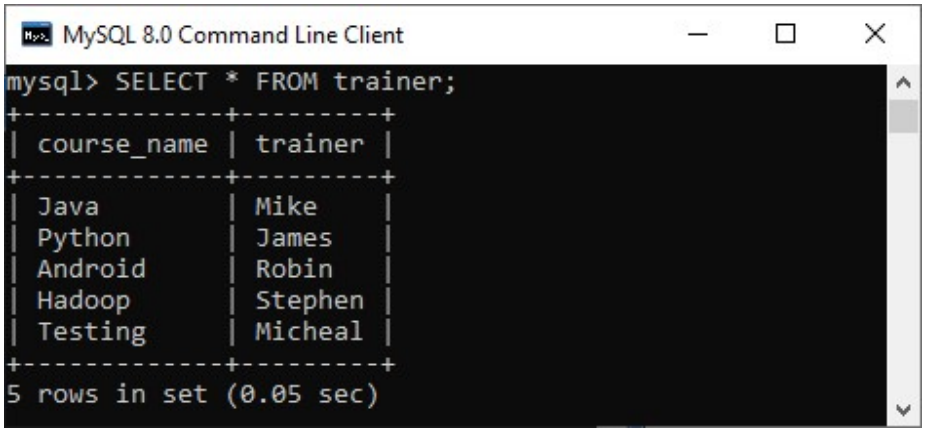
To see the created VIEW

We can see the created view by using the following syntax:

1. `SELECT * FROM view_name;`

Let's see how it looks the created VIEW:

```
SELECT * FROM trainer;
```



NOTE: It is essential to know that a view does not store the data physically. When we execute the SELECT statement for the view, MySQL uses the query specified in the view's definition and produces the output. Due to this feature, it is sometimes referred to as a virtual table.

MySQL Update VIEW

In MYSQL, the ALTER VIEW statement is used to modify or update the already created VIEW without dropping it.

Syntax:

Following is the syntax used to update the existing view in MySQL:

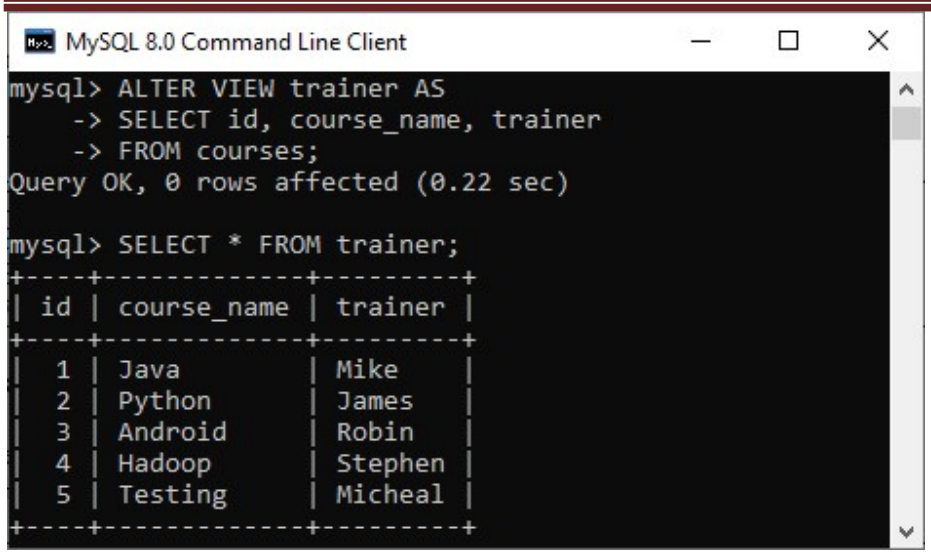
```
ALTER VIEW view_name AS
SELECT columns
FROM table
WHERE conditions;
```

Example:

The following example will alter the already created VIEW name "trainer" by adding a new column.

```
ALTER VIEW trainer AS
SELECT id, course_name, trainer
FROM courses;
```

Once the execution of the **ALTER VIEW** statement becomes successful, MySQL will update a view and stores it in the database. We can see the altered view using the SELECT statement, as shown in the output:



MySQL Drop VIEW

We can drop the existing VIEW by using the **DROP VIEW** statement.

Syntax:

The following is the syntax used to delete the view:

DROP VIEW [IF EXISTS] view_name;

Parameters:

view_name: It specifies the name of the VIEW that we want to drop.

IF EXISTS: It is optional. If we do not specify this clause and the VIEW doesn't exist, the DROP VIEW statement will return an error.

Example:

Suppose we want to delete the view "**trainer**" that we have created above. Execute the below statement:

DROP VIEW trainer;

After successful execution, it is required to verify the view is available or not as below: