Assignment No .2

Title of Assignment: SQL Queries:
   a. Design and Develop SQL DDL statements which demonstrate the use of SQL objects suchas Table, View, Index, Sequence, Synonym, different constraints etc.
   b. Write at least 10 SQL queries on the suitable database application using SQL DMLstatements.
   Note: Instructor will design the queries which demonstrate the use of concepts like Insert, Select, Update, Delete with operators, functions, and set operator etc.

Course Objective:

Implement SQL queries for given requirements, using different SQL concepts

Course Outcome:

C306.3 Implement SQL queries for given requirements, using different SQL concepts

Software Required: - Mysql

Theory: -

Database: - a structured set of data held in a computer, especially one that is accessible in various ways. **Database System**: - A database management system (DBMS) is a computer software application that interactswith the user, other applications, and the database itself to capture and analyze data. A general-purpose DBMS isdesigned to allow the definition, creation, querying, update, and administration of databases.

A database management system (DBMS) is a software tool that makes it possible to organize data in a database.**A database** is a collection of information that is organized so that it can be easily accessed, managed and updated.

SQL: The Language of Database Access: -
Structured Query Language (SQL) is a standardized programming language for accessing and manipulating databases. In an RDBMS like MySQL, Sybase, Oracle, or IMB DM2, SQL writes programming that can manage data and stream data processing. SQL is like a database's own version of a server-side script and is responsible for:

  * Executing queries, which are "questions" asked of the database
  * Retrieving data
  * Editing data: inserting, updating, deleting, or creating new records
  * Creating views

- Setting permissions
- Creating new databases

SQL is a standard programming language, but has a number of variations—including some databases' own proprietary SQL extensions.

Top 7 open source relational databases: -

CUBRID, Firebird, MariaDB, MySQL, Postgre SQL, SQLite

RDBMS Terminology

Before we proceed to explain the MySQL database system, let us revise a few definitions related to the database.

- ✓ **Database** − A database is a collection of tables, with related data.
- ✓ **Table** − A table is a matrix with data. A table in a database looks like a simple spreadsheet.
- ✓ **Column** − One column (data element) contains data of one and the same kind, for example the column postcode.
- ✓ **Row** − A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
- ✓ **Redundancy** − Storing data twice, redundantly to make the system faster.
- ✓ **Primary Key** − A primary key is unique. A key value can not occur twice in one table. With a key, you can only find one row.
- ✓ **Foreign Key** − A foreign key is the linking pin between two tables.
- ✓ **Compound Key** − A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
- ✓ **Index** − An index in a database resembles an index at the back of a book.
- ✓ **Referential Integrity** − Referential Integrity makes sure that a foreign key value always points to an existing row.

MySQL Database

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

- ✓ MySQL is released under an open-source license. So you have nothing to pay to use it.
- ✓ MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- ✓ MySQL uses a standard form of the well-known SQL data language.
- ✓ MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- ✓ MySQL works very quickly and works well even with large data sets.
- ✓ MySQL is very friendly to PHP, the most appreciated language for web development.

- ✓ MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- ✓ MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

Administrative MySQL Command

Here is the list of the important MySQL commands, which you will use time to time to work with MySQL database −

- ✓ **USE Databasename** − This will be used to select a database in the MySQL workarea.
- ✓ **SHOW DATABASES** − Lists out the databases that are accessible by the MySQL DBMS.
- ✓ **SHOW TABLES** − Shows the tables in the database once a database has been selected with the use command.
- ✓ **SHOW COLUMNS FROM** *tablename:* Shows the attributes, types of attributes, key information, whether NULL is permitted, defaults, and other information for a table.
- ✓ **SHOW INDEX FROM tablename** − Presents the details of all indexes on the table, including the PRIMARY KEY.
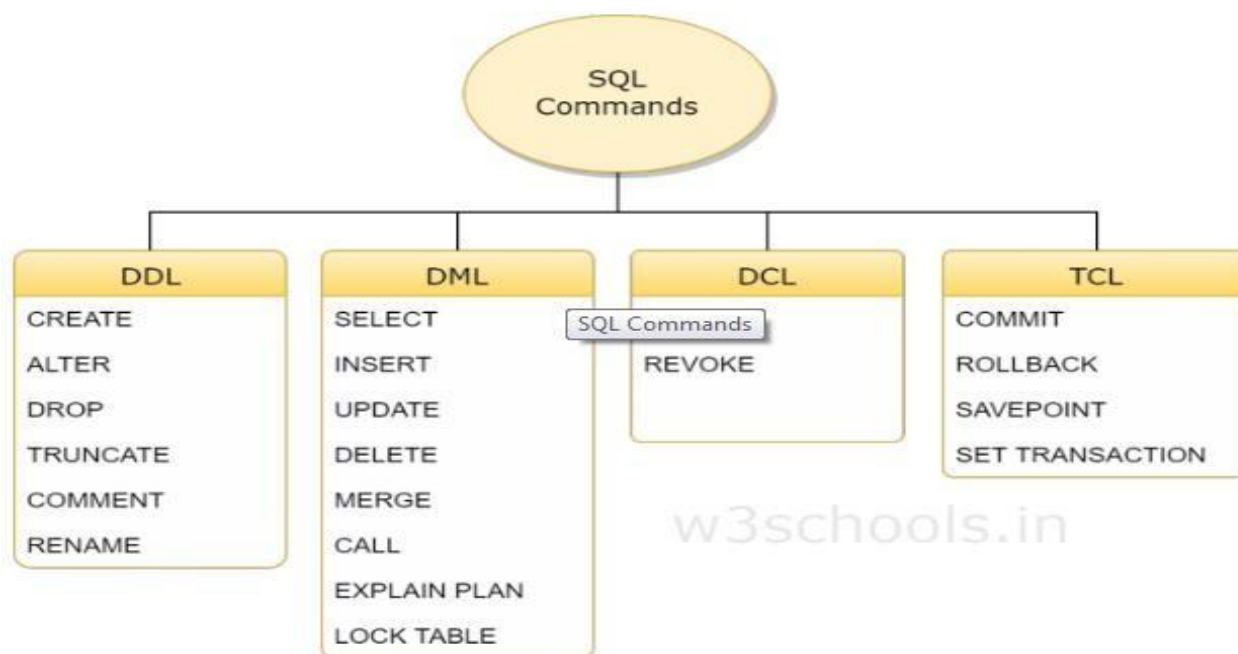
MySQL Connection Using MySQL Binary

You can establish the MySQL database using the **mysql** binary at the command prompt.

Example

Here is a simple example to connect to the MySQL server from the command prompt −

[root@host]#mysql-u root -p

Enter password:******

This will give you the mysql> command prompt where you will be able to execute any SQL command. Following is the result of above command −

The following code block shows the result of above code −

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 2854760 to server version: 5.0.9

Type 'help;' or '\h' for help.Type '\c' to clear the buffer.

In the above example, we have used **root** as a user but you can use any other user as well. Any user will be able to perform all the SQL operations, which are allowed to that user.

You can disconnect from the MySQL database any time using the **exit** command at mysql> prompt.

mysql> exit

Bye

USE DatabaseName;

Always the database name should be unique within the RDBMS.

DROP DATABASE databaseName   -- Delete the database (irrecoverable!)

DROP DATABASE IF EXISTS databaseName   -- Delete if it exists

CREATE DATABASE databaseName   -- Create a new database

CREATE DATABASE IF NOT EXISTS databaseName -- Create only if it does not exists

SHOW DATABASES   -- Show all the databases in this server

USE databaseName   -- Set the default (current) database

SELECT DATABASE()   -- Show the default database

SHOW CREATE DATABASE databaseName   -- Show the CREATE DATABASE statement

-- Table-Level

DROP TABLE [IF EXISTS] tableName, ...

CREATE TABLE [IF NOT EXISTS] tableName (

  columnName columnType columnAttribute, ...

  PRIMARY KEY(columnName),

  FOREIGN KEY (columnNmae) REFERENCES tableName (columnNmae)

)

SHOW TABLES   -- Show all the tables

in the default databaseDESCRIBE|DESC

tableName   -- Describe the

details for a table

ALTER TABLE tableName ... -- Modify a table, e.g., ADD COLUMN and

DROP COLUMNALTER TABLE tableName ADD columnDefinition

ALTER TABLE tableName DROP columnName

ALTER TABLE tableName ADD FOREIGN KEY (columnNmae) REFERENCES

tableName (columnNmae)ALTER TABLE tableName DROP FOREIGN KEY

constraintName

SHOW CREATE TABLE tableName   -- Show the CREATE TABLE statement for this
tableName

Data Manipulation Language (DML)
Allows you to modify the database instance by inserting, modifying, and deleting its data. It is responsible for performing all types of data modification in a database. There are three basic constructs which allow database program and user to enter data and information are:
Here are some important DML commands in SQL:
INSERT
UPDATE
DELETE

INSERT:
This is a statement is a SQL query. This command is used to insert data into the row of a table.
Syntax:

INSERT INTO TABLE_NAME (col1, col2, col3,.... col N)
VALUES (value1, value2, value3, ... valueN);
Or
INSERT INTO TABLE_NAME
VALUES (value1, value2, value3, ... valueN);
For example:

INSERT INTO students (RollNo, FIrstName, LastName) VALUES ('60', 'Tom', Erichsen');

UPDATE:
This command is used to update or modify the value of a column in the table.

Syntax:

UPDATE table_name SET [column_name1= value1,...column_nameN = valueN] [WHERE
CONDITION]
For example:

UPDATE students
SET FirstName = 'Jhon', LastName= 'Wick'
WHERE StudID = 3;
DELETE:
This command is used to remove one or more rows from a table.

Syntax:

DELETE FROM table_name [WHERE condition];
For example:

DELETE FROM students
WHERE FirstName = 'Jhon';

I.   Set Operators

Set operators combine the results of two component queries into a single result. Queries containing set operators are called compound queries. The lists of SQL set operators are "UNION [ALL], INTERSECT, MINUS Operators".You can combine multiple queries usingthe set operators UNION, UNION ALL, INTERSECT, and MINUS. All set operators have

equal precedence. If a SQL statement contains multiple set operators, then Oracle Databaseevaluates them from the left to right unless parentheses explicitly specify another order.Thecorresponding expressions in the select lists of the component queries of a compound querymust match in number and must be in the same datatype group (such as numeric or character).

If component queries select character data, then the datatype of the return values aredetermined as follows:

- If both queries select values of datatype CHAR of equal length, then the returned values have datatype CHAR of that length. If the queries select values of CHAR with different lengths, then the returned value is VARCHAR2 with the length of the larger CHAR value.
- If either or both of the queries select values of datatype VARCHAR2, then the returned values have datatype VARCHAR2.

If component queries select numeric data, then the datatype of the return values is determinedby numeric precedence:

- If any query selects values of type BINARY_DOUBLE, then the returned values have datatype BINARY_DOUBLE.
- If no query selects values of type BINARY_DOUBLE but any query selects values of type BINARY_FLOAT, then the returned values have datatype BINARY_FLOAT.
- If all queries select values of type NUMBER, then the returned values have datatype NUMBER.

In queries using set operators, Oracle does not perform implicit conversion across datatype groups. Therefore, if the corresponding expressions of component queries resolve to both character data and numeric data, Oracle returns an error to fetch only unique records insteadof fetching duplicate records.

II.   SQL Functions

SQL functions are built into Oracle Database and are available for use in various appropriate SQL statements. Do not confuse SQL functions with user-defined functions written in PL/SQL.If you call a SQL function with an argument of a datatype other than the datatype expected by the SQL function, then Oracle attempts to convert the argument to the expected datatype before performing the SQL function. If you call a SQL function with a null argument, then the SQL function automatically returns null.

1. Single-Row Functions

Single-row functions return a single result row for every row of a queried table or view. These   functions   can   appear   in   select lists, WHERE clauses, STARTWITH and CONNECT BY clauses, and HAVING clauses.

2. Character Functions Returning Character Value

Character functions that return character values return values of the following datatypesunless otherwise documented:

- If the input argument is CHAR or VARCHAR2, then the value returnedis VARCHAR2.
- If the input argument is NCHAR or NVARCHAR2, then the value returnedis NVARCHAR2.

The length of the value returned by the function is limited by the maximum length of thedatatype returned.

- For functions that return CHAR or VARCHAR2, if the length of the return value exceeds the limit, then Oracle Database truncates it and returns the result without an error message.
- For functions that return CLOB values, if the length of the return values exceeds the limit, then Oracle raises an error and returns no data.

3. Datetime Functions

Datetime functions operate on date (DATE), timestamp(TIMESTAMP, TIMESTAMP WITH TIME ZONE,and TIMESTAMP WITH L   OCAL TIM E ZONE), and interval (INTERVAL DAY TO SECOND, INTERVAL YEAR   TO MONT values. Some of the datetime functions were designed for the Oracle DATE datatype (ADD_MONTHS, CURRENT_DATE, LAST_DAY, NEW_TIME, and NEXT_DAY). If you

provide a timestamp value as their argument, Oracle Database internally converts the input type toa DATE value and returns a DATE value. The exceptions are the MONTHS_BETWEEN function,which returns a number, and the ROUND and TRUNC functions, which do not accept timestamp or interval values at all. The remaining datetime functions were designed to accept any of the three types of data (date, timestamp, and interval) and to return a value of one of these types.

Conclusion:

Students are able to implement SQL queries for given requirements, using different SQL concepts

Activity to be Submitted by Students

1. Write a SQL statement to create a table job_history including columns employee_id, start_date, end_date, job_id and department_id and make sure that, the employee_id column does not contain any duplicate value at the time of insertion and the foreign key column job_id contain only those values which are exists in the jobs table.

Here is the structure of the table jobs;

```
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| JOB_ID     | varchar(10) | NO   | PRI |         |       |
| JOB_TITLE  | varchar(35) | NO   |     | NULL    |       |
| MIN_SALARY | decimal(6,0)| YES  |     | NULL    |       |
| MAX_SALARY | decimal(6,0)| YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
```

2. Write a SQL statement to create a table employees including columns employee_id, first_name, last_name, job_id, salary and make sure that, the employee_id column does not contain any duplicate value at the time of insertion, and the foreign key column job_id, referenced by the column job_id of jobs table, can contain only those values which are exists in the jobs table. The InnoDB Engine have been used to create the tables. The specialty of the statement is that, The ON DELETE NO ACTION and the ON UPDATE NO ACTION actions will reject the deletion and any updates.

3. Consider the following schema for a LibraryDatabase: BOOK (Book_id, Title, Publisher_Name, Pub_Year) BOOK_AUTHORS (Book_id, Author_Name) PUBLISHER (Name, Address, Phone) BOOK_COPIES (Book_id, Branch_id, No-of_Copies) BOOK_LENDING (Book_id, Branch_id, Card_No, Date_Out, Due_Date) LIBRARY_BRANCH (Branch_id, Branch_Name, Address) Write SQL queries to

1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch,etc.

2. Get the particulars of borrowers who have borrowed more than

3 books, but from Jan 2017 to Jun2017

3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulationoperation.

4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.

5. Create a view of all books and its number of copies that are currently available in the Library.