Example:

```
SELECT column_name(s) FROM table_name
[WHERE condition] ORDER BY column_name(s)
LIMIT number;
```

Example:—

```
SELECT * FROM employees ORDER BY salary
DESC LIMIT 10;
```

* **DELETE**

In MySQL, the Delete command is used to delete one or more records from a table. The basic syntax for the DELETE command is as follows.

```
DELETE FROM table_name WHERE condition;
```

Here, 'table_name' is the name of the table from which you want to delete records, and 'condition' specifies the criteria for selecting the records to be deleted. If you omit 'WHERE' clause, all records in the table will be deleted.

To delete only specific records based on a condition, we would use a command like this.

```
DELETE FROM mytable; WHERE id = 1;
```

This would be delete only the record(s) where the 'id' column has a value of 1.

* **UPDATE :—**

The UPDATE command is used to modifying existing records in a table.

```
UPDATE table_name SET column1 = value1, column2 = value2,
..... WHERE condition;
```

For example:- To change the 'name' and 'age' columns of a record where 'id' equals 1 in a table named 'mytable', you would use the following command.

```
UPDATE mytable SET name = 'John', age = 30
WHERE id = 1;
```

⇒ It is important to use a 'WHERE' clause when using the update command, otherwise all records in the table will be updated.

* **ALTER :—**

The ALTER command is used to modify the structure of a table, such as adding or removing columns, changing columns' datatypes, or renaming the table itself.

```
ALTER TABEE table_name action;
```

- Some examples of ALTER commands :—

```
ALTER TABLE mytable ADD COLUMN new_column_name
data_type;
```

This would add a new column with the specified name and data type to the table name 'mytable'.

- Changing the data type of an existing column.
  ALTER TABLE mytable MODIFY COLUMN columnname new_data_type.;

- Renaming a table:
  ALTER TABLE old_table_name RENAME TO new_table_name;

⇒ Note that when using the ALTER command to modify a table, there is a possibility that data loss or corruption can occur. Therefore it is important to backup your data before making any significant changes to a table's structure.

→ D/F b/w ALTER and Update command.
The main d/f b/w ALTER and UPDATE commands in MYSQL that 'ALTER' is used to modify the structure of a table, while 'UPDATE' is used to modify the data within a table.

* ADD COLUMN ("AFTER" or "FIRST").
When adding a new column to an existing table using the ALTER command, you can specify the position of the new column within the table using the "ADD COLUMN" statement with the "AFTER" or "FIRST" keywords.
Example:-
ALTER TABLE mytable ADD COLUMN newColumn_Name data_type AFTER existing_column_name.

ALTER TABLE mytable ADD COLUMN new_column_name data_type FIRST;

* **DROP** :—
The 'DROP' command is used to delete a database, a table, or an index. The basic syntax for the DROP command as follows:—

- To drop a database:—
  DROP DATABASE database_name;

- To drop a table :—
  DROP TABLE table_name;

* DROP command with ALTER command :—

ALTER TABLE table_name DROP COLUMN column_name;

* SELECT command example

→ SELECT DoB As "Date of Birth" FROM table_name;

→ SELECT CONCAT(First_name_column, ' ', Second_name_column) AS Name FROM table_name;
(' ' for space b/w two columns)

→ SELECT DISTINCT Score FROM table_name;
'Distinct' keyword is used in a SELECT command to eliminate duplicate rows from the Result set. It is used in conjuction with the SELECT keyword to retrieve unique values from one or more columns of a table.

→ SELECT COUNT(*) FROM table_name;

The 'COUNT' function is used to count the number of rows that match a specified condition in a table. It can be used with the SELECT statement to retrieve the number of rows in a table or the number of rows that match a specific condition.

Example:—

```
SELECT COUNT(*) FROM customers WHERE
last_name = 'Smith';
```

→ SELECT SUM(column_name) FROM table_name;

The 'SUM' function is used to calculate the sum of values in a specifies column of a table. It is often used in combination with the 'SELECT' statement to retrieve the total value of a specific column.

→ SELECT AVG(column_name) FROM table_name;

The 'AVG' function is used to calculate the average value of a specific column in a table.

→ other examples:—

- MIN()
- MAX()

* Group By:—

'Group By' clause is used in a SELECT statement to group the result set based on one or more columns. It is often used in combination with aggregate functions such as SUM, COUNT, AVG, MIN and MAX to perform calculations on group of data.

→ SELECT column_name, aggregate_fun(column_name) FROM table_name GROUP BY column_name;

Example:—

• SELECT product_name, SUM(sales_amount) FROM sales GROUP BY product_name;

* **HAVING:—**
The 'HAVING' clause is used in a SELECT statement to filter the result set based on aggregate functions values. It is often used in combination with the GROUP BY clause to filter groups of data based on specific criteria.

→ SELECT column_name, aggregate_fun(column_name) FROM table_name GROUP BY column_name HAVING condition

Example:—

SELECT product_name, SUM(sales_amount) FROM sales GROUP BY productname HAVING SUM(sales_amount) > 10000;

→ **Note:** 'HAVING' clause is applied after the 'GROUP BY' clause, so it can only be used with aggregate functions such as SUM, COUNT, AVG, MIN and MAX.