Date Functions:

Date_add(date, interval)

Ex. To add 1 week to current date.

Select date_add(curdate(), interval 1 week) as date;

```
mysql> Select date_add(curdate(), interval 1 week) as date;
+------------+
| date       |
+------------+
| 2025-08-04 |
+------------+
1 row in set (0.00 sec)

mysql> Select date_add(curdate(), interval 1 year) as date;
+------------+
| date       |
+------------+
| 2026-07-28 |
```

```
+------------+
1 row in set (0.00 sec)

mysql> Select date_add(curdate(), interval 4 day)
as date;
+------------+
| date       |
+------------+
| 2025-08-01 |
+------------+
1 row in set (0.00 sec)
```

Ex. To find employees who joined in February.

Month() -> this is used to extract a month of a date.

```
Select ename, hiredate
From emp
Where month(hiredate) = 2;

mysql> Select ename, hiredate
    -> From emp
```

```
    -> Where month(hiredate) = 2;
+-------+------------+
| ename | hiredate   |
+-------+------------+
| ALLEN | 1981-02-20 |
| WARD  | 1981-02-22 |
+-------+------------+
2 rows in set (0.02 sec)
```

TASK:

Table: Medicine

Columns: MID, Mname, price, exp_date

Requirement: Find medicines expiring in 3 months.

DML(data manipulation language)

Insert:

Update: Modify existing data in the table

Delete: remove rows from table

- Update
  Syntax:
  Update table_name
  Set column1 = value1, column2 = value2,……
  Where condition

Ex. To increase the salary of employees in department no 30 by 15%.

```
Update emp
Set sal = sal*1.15
Where deptno=30;
```

```
mysql> Update emp
    -> Set sal = sal*1.15
    -> Where deptno=30;
Query OK, 6 rows affected (0.05 sec)
Rows matched: 6  Changed: 6  Warnings: 0
```

```
mysql> select * from emp;
+------+--------+-----------+------+-----------
+---------+---------+---------+
| EMPNO | ENAME  | JOB       | MGR  | HIREDATE   |
sal     | COMM    | DEPTNO |
```

```
+------+--------+-----------+------+------------
+---------+---------+--------+

|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |
800.00 |    NULL |     20 |

|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 |
1840.00 |  300.00 |     30 |

|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 |
1437.50 |  500.00 |     30 |

|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 |
2975.00 |    NULL |     20 |

|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 |
1437.50 | 1400.00 |     30 |

|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 |
3277.50 |    NULL |     30 |

|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 |
2450.00 |    NULL |     10 |

|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 |
3000.00 |    NULL |     20 |

|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 |
5000.00 |    NULL |     10 |

|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 |
1725.00 |    0.00 |     30 |

|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 |
1100.00 |    NULL |     20 |
```

```
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |
1092.50 |    NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 |
3000.00 |    NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 |
1300.00 |    NULL |     10 |
+------+--------+-----------+------+-----------
+---------+---------+--------+
```

* To change the job of smith(7369) from clerk to senior clerk.

```
Update emp
Set job =  ‘Senior Clerk’
Where empno = 7369;
```

```
mysql> Update emp
    -> Set job = ’Senior Clerk’
    -> Where empno = 7369;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> select * from emp;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	sal	COMM	DEPTNO
7369	SMITH	Senior Clerk	7902	1980-12-17	800.00	NULL	20
7499	ALLEN	SALESMAN	7698	1981-02-20	1840.00	300.00	30
7521	WARD	SALESMAN	7698	1981-02-22	1437.50	500.00	30
7566	JONES	MANAGER	7839	1981-04-02	2975.00	NULL	20
7654	MARTIN	SALESMAN	7698	1981-09-28	1437.50	1400.00	30
7698	BLAKE	MANAGER	7839	1981-05-01	3277.50	NULL	30
7782	CLARK	MANAGER	7839	1981-06-09	2450.00	NULL	10
7788	SCOTT	ANALYST	7566	1982-12-09	3000.00	NULL	20
7839	KING	PRESIDENT	NULL	1981-11-17	5000.00	NULL	10
7844	TURNER	SALESMAN	7698	1981-09-08	1725.00	0.00	30
7876	ADAMS	CLERK	7788	1983-01-12	1100.00	NULL	20
7900	JAMES	CLERK	7698	1981-12-03	1092.50	NULL	30
7902	FORD	ANALYST	7566	1981-12-03	3000.00	NULL	20
7934	MILLER	CLERK	7782	1982-01-23	1300.00	NULL	10

Delete Statement:

Syntax:

Delete from table_name

Where condition;

Ex. Remove employee whose empno is 7900

Delete from emp

Where empno = 7900;

Ex. Remove all employees whose salary is less than 1000.

```
Delete from emp
Where sal < 1000;
```

Assignment Question:

```
Select concat(TRIM(ename),' .' ) as empName from emp;
```

```
mysql> Select concat(TRIM(ename),'.') as empName from emp;
+---------+
| empName |
+---------+
| SMITH.  |
| ALLEN.  |
| WARD.   |
| JONES.  |
| MARTIN. |
| BLAKE.  |
| CLARK.  |
| SCOTT.  |
```

| KING. |

| TURNER. |

| ADAMS. |

| FORD. |

| MILLER. |

+---------+

JOINS:

Creating a department table:

```
Create table dept (
                DEPTNO INT(2) primary key,
                DNAME VARCHAR(20),
                LOC VARCHAR(15)
);
```

Inserting values in table

```
INSERT INTO DEPT VALUES(10,'ACCOUNTING','NEW YORK');
INSERT INTO DEPT VALUES(20,'RESEARCH','DALLAS');
```

```
INSERT INTO DEPT VALUES(30,'SALES','CHICAGO');
INSERT INTO DEPT VALUES(40,'OPRATIONS','BOSTON');
```

It allows us to retrieve data from multiple tables based on related columns.

Types of joins:

1. Inner Join
2. Left Join (Left Outer Join)
3. Right Join (Right Outer Join)
4. Full Join(Full Outer Join)

* Left Join (Left Outer Join)

It returns all rows from the left table and matched rows from the right table.
Unmatched rows from the right table will display NULL.

SYNTAX:

```
Select columns
From table1
Left join table2
ON table1.column = table2.column;
```

Ex. To retrieve all employees and their department names, even if some employees does not belong to any department.

```
Select emp.ename, emp.job, dept.dname
From emp
Left join dept
On emp.deptno = dept.deptno;
```

```
mysql> Select emp.ename, emp.job, dept.dname
    -> From emp
    -> Left join dept
    -> On emp.deptno = dept.deptno;
+--------+--------------+------------+
| ename  | job          | dname      |
+--------+--------------+------------+
| SMITH  | Senior Clerk | RESEARCH   |
| ALLEN  | SALESMAN     | SALES      |
| WARD   | SALESMAN     | SALES      |
| JONES  | MANAGER      | RESEARCH   |
| MARTIN | SALESMAN     | SALES      |
| BLAKE  | MANAGER      | SALES      |
| CLARK  | MANAGER      | ACCOUNTING |
| SCOTT  | ANALYST      | RESEARCH   |
| KING   | PRESIDENT    | ACCOUNTING |
| TURNER | SALESMAN     | SALES      |
| ADAMS  | CLERK        | RESEARCH   |
```

```
| FORD   | ANALYST       | RESEARCH   |
| MILLER | CLERK         | ACCOUNTING |
+--------+---------------+------------+
```

TASK:
Add an employee who is working in deptno (60) then execute the query and analyze

- Inner join

  Returns only where there is match in both tables.

  SYNTAX:

```
select columns
From table1
Inner join table2
On table1.column = table2.column;
```

  To retrieve employees and their department names.

```
Select emp.ename, emp.job, dept.dname
From emp
Inner join dept
On emp.deptno = dept.deptno;
```

```
mysql> Select emp.ename, emp.job, dept.dname
```

```
    -> From emp
    -> Inner join dept
    -> On emp.deptno = dept.deptno;
+--------+--------------+------------+
| ename  | job          | dname      |
+--------+--------------+------------+
| SMITH  | Senior Clerk | RESEARCH   |
| ALLEN  | SALESMAN     | SALES      |
| WARD   | SALESMAN     | SALES      |
| JONES  | MANAGER      | RESEARCH   |
| MARTIN | SALESMAN     | SALES      |
| BLAKE  | MANAGER      | SALES      |
| CLARK  | MANAGER      | ACCOUNTING |
| SCOTT  | ANALYST      | RESEARCH   |
| KING   | PRESIDENT    | ACCOUNTING |
| TURNER | SALESMAN     | SALES      |
| ADAMS  | CLERK        | RESEARCH   |
| FORD   | ANALYST      | RESEARCH   |
| MILLER | CLERK        | ACCOUNTING |
+--------+--------------+------------+
```

Right Join: It return all rows from the right table and matched rows from the left table. Unmatched rows from the left table will display NULL.

Ex. To retrieve all departments and their employees, including departments with no employees.

```
mysql> select emp.ename,emp.job, dept.dname
    -> from emp
    -> right join dept
    -> on emp.deptno = dept.deptno;
+--------+-------------+------------+
| ename  | job         | dname      |
+--------+-------------+------------+
| MILLER | CLERK       | ACCOUNTING |
| KING   | PRESIDENT   | ACCOUNTING |
| CLARK  | MANAGER     | ACCOUNTING |
| FORD   | ANALYST     | RESEARCH   |
| ADAMS  | CLERK       | RESEARCH   |
| SCOTT  | ANALYST     | RESEARCH   |
| JONES  | MANAGER     | RESEARCH   |
| SMITH  | Senior Clerk | RESEARCH   |
| TURNER | SALESMAN    | SALES      |
| BLAKE  | MANAGER     | SALES      |
| MARTIN | SALESMAN    | SALES      |
| WARD   | SALESMAN    | SALES      |
| ALLEN  | SALESMAN    | SALES      |
| NULL   | NULL        | OPRATIONS  |
+--------+-------------+------------+
```

FULL Outer Join
UNION