

## WEEK 2:

### TRIGGERS:

```
mysql> CREATE TRIGGER before_student_update
-> BEFORE UPDATE
-> ON Student
-> FOR EACH ROW
-> BEGIN
-> IF NEW.dept IS NULL THEN
-> SIGNAL SQLSTATE '45000'
-> SET MESSAGE_TEXT ='dept cannot be NULL';
-> END IF;
-> END $$
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> DELIMITER;
-> ;
-> ^C
```

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER before_student_delete
-> BEFORE DELETE
-> ON Student
-> FOR EACH ROW
-> BEGIN
-> INSERT INTO Student_log(id,dept,age)
-> VALUES(OLD.id,'DELETE',NOW());
-> END $$
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> CREATE TRIGGER before_student_delete
-> BEFORE DELETE
-> ON Student
-> FOR EACH ROW
-> BEGIN
-> INSERT INTO Student_log(id,dept,age)
-> VALUES(OLD.id,'DELETE',NOW());
-> END $$
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> CREATE TRIGGER after_Student_insert
-> AFTER INSERT
-> ON Student
-> FOR EACH ROW
-> BEGIN
-> INSERT INTO Student(id,dept,age)
-> VALUES(NEW.id,'INSERT',NOW());
-> END $$
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> DELIMITER ;
```

```

mysql> CREATE TABLE faculty(faculty_id INT PRIMARY KEY, name VARCHAR(50),otp INT);
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO faculty VALUES(123,'Shri',56789);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO faculty VALUES(346,'Hema',984638);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO faculty VALUES(875,'Latha',65549);
Query OK, 1 row affected (0.01 sec)

mysql> DELIMITER $$
mysql> CREATE TRIGGER before_faculty_insert
-> ^C
mysql> CREATE TRIGGER before_faculty_insert_update
-> BEFORE INSERT ON faculty
-> FOR EACH ROW
-> BEGIN
-> IF NEW.otp < 1000 OR NEW.otp > 99999 THEN
-> SIGNAL SQLSTATE '45000'
-> SET MESSAGE_TEXT = 'OTP must be a 4-digit number';
-> END IF;
-> END$$
Query OK, 0 rows affected (0.02 sec)

```

```

mysql> CREATE TABLE faculty(faculty_id INT PRIMARY KEY AUTO_INCREMENT,name VARCHAR(100),salary DECIMAL(10,2),created_at TIMESTAMP);
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE TABLE faculty_audit(audit_id INT PRIMARY KEY AUTO_INCREMENT,faculty_id INT,old_name VARCHAR(100),old_dept VARCHAR(50),old_salary DECIMAL(10,3),updated_at TIMESTAMP);
Query OK, 0 rows affected (0.06 sec)

mysql> DELIMITER $$
mysql> CREATE TRIGGER before_faculty_insert
-> BEFORE INSERT ON faculty
-> FOR EACH ROW
-> BEGIN
-> SET NEW.created_at = NOW();
-> END$$
Query OK, 0 rows affected (0.02 sec)

```

```

mysql> CREATE TRIGGER prevent_negative_salary
-> BEFORE INSERT ON faculty
-> FOR EACH ROW
-> BEGIN
-> IF NEW.salary < 0 THEN
-> SIGNAL SQLSTATE '45000'
-> SET MESSAGE_TEXT = 'Salary cannot be negative';
-> END IF;
-> END$$
Query OK, 0 rows affected (0.02 sec)

```

```

mysql> DELIMITER $$
mysql> CREATE TRIGGER prevent_negative_salary_update
-> BEFORE UPDATE ON faculty
-> FOR EACH ROW
-> BEGIN
-> IF NEW.salary < 0 THEN
-> SIGNAL SQLSTATE '45000'
-> SET MESSAGE_TEXT = 'Salary cannot be negative';
-> END IF;
-> END$$
Query OK, 0 rows affected (0.01 sec)

```

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER after_faculty_insert
-> AFTER INSERT ON faculty
-> FOR EACH ROW
-> BEGIN
->     INSERT INTO faculty_audit
->     (faculty_id, old_name, old_department, old_salary, updated_at)
->     VALUES
->     (NEW.faculty_id, NULL, NULL, NULL, NOW());
-> END$$
Query OK, 0 rows affected (0.02 sec)

mysql> DELIMITER ;
```

## Working with Date:

```
mysql> SELECT * FROM faculty;
+-----+-----+-----+-----+-----+-----+
| faculty_id | name       | salary | created_at           | end_date | start_date |
+-----+-----+-----+-----+-----+-----+
| 4 | Ravi Kumar | 45000.00 | 2026-02-04 17:36:42 | 2026-02-10 | 2026-02-01 |
| 5 | Anita Sharma | 52000.00 | 2026-02-04 17:36:42 | 2026-03-05 | 2026-02-05 |
| 6 | Suresh Rao | 48000.00 | 2026-02-04 17:36:42 | 2026-02-20 | 2026-01-20 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SELECT faculty_id, start_date, end_date, DATEDIFF(end_date, start_date) AS total_days FROM faculty;
+-----+-----+-----+-----+
| faculty_id | start_date | end_date | total_days |
+-----+-----+-----+-----+
| 4 | 2026-02-01 | 2026-02-10 | 9 |
| 5 | 2026-02-05 | 2026-03-05 | 28 |
| 6 | 2026-01-20 | 2026-02-20 | 31 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SELECT faculty_id, start_date, DATE_ADD(start_date, INTERVAL 30 DAY) AS expiry_date FROM faculty;
+-----+-----+-----+
| faculty_id | start_date | expiry_date |
+-----+-----+-----+
| 4 | 2026-02-01 | 2026-03-03 |
| 5 | 2026-02-05 | 2026-03-07 |
| 6 | 2026-01-20 | 2026-02-19 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> SELECT DATE_SUB(CURDATE(), INTERVAL 7 DAY) AS date_before_7_days;
+-----+
| date_before_7_days |
+-----+
| 2026-01-28         |
+-----+
1 row in set (0.00 sec)

mysql> SELECT faculty_id,
->     created_at,
->     DAYNAME(created_at) AS day_name
-> FROM faculty
-> WHERE DAYOFWEEK(created_at) IN (1, 7);
Empty set (0.00 sec)

mysql> DROP TRIGGER before_faculty_insert;
Query OK, 0 rows affected (0.01 sec)

mysql> UPDATE faculty SET created_at = '2026-02-08' WHERE faculty_id = 1;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> UPDATE faculty SET created_at = '2026-02-09' WHERE faculty_id = 2;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> UPDATE faculty SET created_at = '2026-02-07' WHERE faculty_id = 3;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0
```

```

mysql> SELECT faculty_id, created_at, DAYNAME(created_at)
      -> FROM faculty
      -> WHERE DAYOFWEEK(created_at) IN (1,7);
Empty set (0.00 sec)

mysql> SELECT faculty_id, created_at, DAYNAME(created_at)
      -> FROM faculty
      -> WHERE DAYNAME(created_at) IN ('Saturday', 'Sunday');
Empty set (0.00 sec)

mysql> SELECT MONTHNAME(
      ->          STR_TO_DATE('15-02-2026', '%d-%m-%Y')
      ->          ) AS month_name;
+-----+
| month_name |
+-----+
| February   |
+-----+
1 row in set (0.00 sec)

```

Indexes and Views:

```

mysql> ALTER TABLE faculty
      -> ADD email VARCHAR(100);
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> CREATE INDEX idx_faculty_email
      -> ON faculty(email);
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE faculty
      -> ADD username VARCHAR(50);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> CREATE UNIQUE INDEX idx_unique_username
      -> ON faculty(username);
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

```
mysql> SHOW INDEX FROM faculty;
```

| Table   | Non_unique | Key_name            | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
|---------|------------|---------------------|--------------|-------------|-----------|-------------|----------|--------|------|------------|---------|---------------|---------|------------|
| faculty | 0          | PRIMARY             | 1            | faculty_id  | A         | 3           | NULL     | NULL   | NULL | BTREE      |         |               | YES     | NULL       |
| faculty | 0          | idx_unique_username | 1            | username    | A         | 1           | NULL     | NULL   | YES  | BTREE      |         |               | YES     | NULL       |
| faculty | 1          | idx_faculty_email   | 1            | email       | A         | 1           | NULL     | NULL   | YES  | BTREE      |         |               | YES     | NULL       |

```

3 rows in set (0.01 sec)

mysql> DROP INDEX idx_faculty_email ON faculty;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE faculty
  -> ADD aadhar BIGINT;
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> CREATE INDEX idx_faculty_aadhar
  -> ON faculty(aadhar);
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

```
mysql> SELECT * FROM faculty WHERE aadhar = 123456789012;
Empty set (0.00 sec)

mysql> ALTER TABLE faculty
  -> ADD status VARCHAR(20);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> CREATE INDEX idx_status_createdat
  -> ON faculty(status, created_at);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> SELECT *
  -> FROM faculty
  -> WHERE status = 'ACTIVE'
  -> ORDER BY created_at DESC;
Empty set (0.00 sec)

mysql> SELECT * FROM faculty;
```

| faculty_id | name         | salary   | created_at          | end_date   | start_date | email | username | aadhar | status |
|------------|--------------|----------|---------------------|------------|------------|-------|----------|--------|--------|
| 4          | Ravi Kumar   | 45000.00 | 2026-02-04 17:36:42 | 2026-02-10 | 2026-02-01 | NULL  | NULL     | NULL   | NULL   |
| 5          | Anita Sharma | 52000.00 | 2026-02-04 17:36:42 | 2026-03-05 | 2026-02-05 | NULL  | NULL     | NULL   | NULL   |
| 6          | Suresh Rao   | 48000.00 | 2026-02-04 17:36:42 | 2026-02-20 | 2026-01-20 | NULL  | NULL     | NULL   | NULL   |

```

3 rows in set (0.00 sec)

```

## SQL Built-in Functions:

```
mysql> SELECT
->     email,
->     CONCAT(SUBSTRING_INDEX(email, '@', 1), '123') AS default_password
-> FROM Student;
+-----+-----+
| email          | default_password |
+-----+-----+
| amit@gmail.com | amit123          |
| riya@gmail.com | riya123          |
| rahul@gmail.com | rahul123         |
| sneha@gmail.com | sneha123         |
| arjun@gmail.com | arjun123         |
| kiran@gmail.com | kiran123         |
+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT
->     phone,
->     REGEXP_REPLACE(phone, '^[0-9]', '') AS valid_phone
-> FROM Student;
+-----+-----+
| phone          | valid_phone       |
+-----+-----+
| +919876543210 | 919876543210     |
| 9876543211    | 9876543211       |
| +919876543212 | 919876543212     |
| 9876543213    | 9876543213       |
| +919876543214 | 919876543214     |
| 9876543220    | 9876543220       |
+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT
->     name,
->     CONCAT(
->         UPPER(SUBSTRING(name, 1, 1)),
->         LOWER(SUBSTRING(name, 2))
->     ) AS normalized_name
-> FROM Student;
+-----+-----+
| name          | normalized_name   |
+-----+-----+
| Amit          | Amit             |
| Riya          | Riya             |
| Rahul         | Rahul            |
| Sneha         | Sneha            |
| Arjun         | Arjun            |
| Kiran         | Kiran            |
+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT
->     name,
->     COALESCE(phone, email) AS primary_contact
-> FROM Student;
+-----+-----+
| name          | primary_contact   |
+-----+-----+
| Amit          | +919876543210    |
| Riya          | 9876543211       |
| Rahul         | +919876543212    |
| Sneha         | 9876543213       |
| Arjun         | +919876543214    |
| Kiran         | 9876543220       |
+-----+-----+
6 rows in set (0.00 sec)
```

```

mysql> ALTER TABLE Student
    -> ADD experience_start DATE;
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> UPDATE Student SET experience_start = '2019-06-01' WHERE vtu_no = 'VTU101';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Student SET experience_start = '2020-08-01' WHERE vtu_no = 'VTU102';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Student SET experience_start = '2018-07-01' WHERE vtu_no = 'VTU103';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Student SET experience_start = '2021-01-01' WHERE vtu_no = 'VTU104';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Student SET experience_start = '2017-06-01' WHERE vtu_no = 'VTU105';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT
    -> department,
    -> AVG(TIMESTAMPDIFF(YEAR, experience_start, CURDATE())) AS avg_experience
    -> FROM Student
    -> GROUP BY department;
+-----+-----+
| department | avg_experience |
+-----+-----+
| CSE       | 7.0000       |
| ISE       | 5.0000       |
| ECE       | 5.0000       |
+-----+-----+
3 rows in set (0.00 sec)

```

## User-Defined Functions:

```
mysql> DELIMITER $$
mysql>
mysql> CREATE FUNCTION net_salary(salary DECIMAL(10,2))
    -> RETURNS DECIMAL(10,2)
    -> DETERMINISTIC
    -> BEGIN
    ->     RETURN salary - (salary * 0.10);
    -> END$$
```

Query OK, 0 rows affected (0.01 sec)

```
mysql>
mysql> DELIMITER ;
mysql> SELECT net_salary(50000);
```

```
+-----+
| net_salary(50000) |
+-----+
|          45000.00 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> DELIMITER $$
mysql>
mysql> CREATE FUNCTION user_activity(last_login DATE)
    -> RETURNS VARCHAR(10)
    -> DETERMINISTIC
    -> BEGIN
    ->     IF DATEDIFF(CURDATE(), last_login) <= 30 THEN
    ->         RETURN 'Active';
    ->     ELSE
    ->         RETURN 'Inactive';
    ->     END IF;
    -> END$$
```

Query OK, 0 rows affected (0.02 sec)



```

mysql>
mysql> DELIMITER ;
mysql> SELECT user_activity('2025-01-20');
+-----+
| user_activity('2025-01-20') |
+-----+
| Inactive                     |
+-----+
1 row in set (0.00 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE FUNCTION tax_slab(salary DECIMAL(10,2))
-> RETURNS INT
-> DETERMINISTIC
-> BEGIN
->   IF salary <= 300000 THEN
->     RETURN 0;
->   ELSEIF salary <= 600000 THEN
->     RETURN 10;
->   ELSEIF salary <= 1000000 THEN
->     RETURN 20;
->   ELSE
->     RETURN 30;
->   END IF;
-> END$$
Query OK, 0 rows affected (0.02 sec)

```

```

mysql>
mysql> DELIMITER ;
mysql> SELECT tax_slab(850000);
+-----+
| tax_slab(850000) |
+-----+
|                20 |
+-----+
1 row in set (0.00 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE FUNCTION experience_category(years INT)
-> RETURNS VARCHAR(20)
-> DETERMINISTIC
-> BEGIN
->   IF years < 2 THEN
->     RETURN 'Fresher';
->   ELSEIF years BETWEEN 2 AND 5 THEN
->     RETURN 'Junior';
->   ELSEIF years BETWEEN 6 AND 10 THEN
->     RETURN 'Mid';
->   ELSE
->     RETURN 'Senior';
->   END IF;
-> END$$
Query OK, 0 rows affected (0.01 sec)

```

```

mysql>
mysql> DELIMITER ;
mysql> SELECT experience_category(7);
+-----+
| experience_category(7) |
+-----+
| Mid                    |
+-----+
1 row in set (0.00 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE FUNCTION late_fee(due_date DATE, paid_date DATE)
    -> RETURNS INT
    -> DETERMINISTIC
    -> BEGIN
    ->   DECLARE days_late INT;
    ->   DECLARE fee INT;
    ->
    ->   SET days_late = DATEDIFF(paid_date, due_date);
    ->
    ->   IF days_late <= 0 THEN
    ->     SET fee = 0;
    ->   ELSE
    ->     SET fee = days_late * 50;
    ->     IF fee > 1000 THEN
    ->       SET fee = 1000;
    ->     END IF;
    ->   END IF;
    ->
    ->   RETURN fee;
    -> END$$
Query OK, 0 rows affected (0.01 sec)

```

```

mysql>
mysql> DELIMITER ;
mysql> SELECT late_fee('2025-01-01', '2025-02-01');
+-----+
| late_fee('2025-01-01', '2025-02-01') |
+-----+
|                                     1000 |
+-----+
1 row in set (0.00 sec)

```