

```
MySQL 8.0 Command Line Client
-> RETURNS CHAR(1)
-> DETERMINISTIC
-> BEGIN
->     DECLARE grade CHAR(1);
->
->     IF score >= 80 THEN
->         SET grade = 'A';
->     ELSEIF score >= 60 THEN
->         SET grade = 'B';
->     ELSEIF score >= 40 THEN
->         SET grade = 'C';
->     ELSE
->         SET grade = 'F';
->     END IF;
->
->     RETURN grade;
-> END //
```

Query OK, 0 rows affected (0.00 sec)

```
mysql>
mysql> DELIMITER ;
mysql> SELECT student_name, marks, GetGrade(marks) AS grade
-> FROM Students;
```

| student_name | marks | grade |
|--------------|-------|-------|
| Alice        | 85    | A     |
| Bob          | 72    | B     |
| Charlie      | 56    | C     |
| David        | 40    | C     |
| Eva          | 90    | A     |

```
-> P.name AS passenger_name,  
-> P.social_network,  
-> P.social_network_id,  
-> Tr.train_name,  
-> Tr.departure,  
-> Tr.destination,  
-> T.seat_number,  
-> T.status  
-> FROM Tickets T  
-> JOIN Passengers P ON T.passenger_id = P.passenger_id  
-> JOIN Trains Tr ON T.train_id = Tr.train_id;
```

| ticket_id | passenger_name | social_network | social_network_id | train_name | departure | destination | seat_number | status |
|-----------|----------------|----------------|-------------------|------------|-----------|-------------|-------------|--------|
| 1         | John Doe       | Facebook       | john.fb123        | Express A  | New York  | Washington  | 12A         | Booked |
| 2         | Jane Smith     | Twitter        | jane_tw           | Local B    | Boston    | New York    | 5B          | Booked |

2 rows in set (0.00 sec)

```
mysql> SELECT  
-> P.name, P.social_network_id, T.ticket_id, Tr.train_name, T.seat_number  
-> FROM Tickets T  
-> JOIN Passengers P ON T.passenger_id = P.passenger_id  
-> JOIN Trains Tr ON T.train_id = Tr.train_id  
-> WHERE P.social_network = 'Facebook';
```

| name     | social_network_id | ticket_id | train_name | seat_number |
|----------|-------------------|-----------|------------|-------------|
| John Doe | john.fb123        | 1         | Express A  | 12A         |

1 row in set (0.00 sec)

```
mysql> UPDATE Tickets  
-> SET status = 'Cancelled'  
-> WHERE ticket_id = 2;  
Query OK, 1 row affected (0.01 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> DELETE FROM Tickets  
-> WHERE ticket_id = 2;
```

MySQL 8.0 Command Line Client

```
mysql> INSERT INTO AdmissionForm (student_name, date_of_birth, course_applied, admission_date)
-> VALUES
-> ('John Doe', '2003-05-15', 'Computer Science', '2025-09-01'),
-> ('Jane Smith', '2004-08-20', 'Mechanical Engineering', '2025-09-01'),
-> ('Alice Johnson', '2003-12-10', 'Civil Engineering', '2025-09-02');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT * FROM AdmissionForm;
```

| admission_id | student_name  | date_of_birth | course_applied         | admission_date | status  |
|--------------|---------------|---------------|------------------------|----------------|---------|
| 1            | John Doe      | 2003-05-15    | Computer Science       | 2025-09-01     | Pending |
| 2            | Jane Smith    | 2004-08-20    | Mechanical Engineering | 2025-09-01     | Pending |
| 3            | Alice Johnson | 2003-12-10    | Civil Engineering      | 2025-09-02     | Pending |

3 rows in set (0.00 sec)

```
mysql> SELECT student_name, admission_date, status
-> FROM AdmissionForm
-> WHERE course_applied = 'Computer Science';
```

| student_name | admission_date | status  |
|--------------|----------------|---------|
| John Doe     | 2025-09-01     | Pending |

1 row in set (0.00 sec)

```
mysql> UPDATE AdmissionForm
-> SET status = 'Approved'
-> WHERE admission_id = 1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> DELETE FROM AdmissionForm
-> WHERE admission_id = 3;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> CREATE TABLE Students (  
->   student_id INT AUTO_INCREMENT PRIMARY KEY,  
->   student_name VARCHAR(100),  
->   admission_date DATE  
-> );  
Query OK, 0 rows affected (0.04 sec)
```

```
mysql>  
mysql> CREATE TABLE StudentAudit (  
->   audit_id INT AUTO_INCREMENT PRIMARY KEY,  
->   student_id INT,  
->   action VARCHAR(50),  
->   action_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
-> );  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> DELIMITER //  
mysql>  
mysql> CREATE TRIGGER after_student_insert  
-> AFTER INSERT ON Students  
-> FOR EACH ROW  
-> BEGIN  
->   INSERT INTO StudentAudit (student_id, action)  
->   VALUES (NEW.student_id, 'INSERT');  
-> END; //  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>  
mysql> DELIMITER ;  
mysql> INSERT INTO Students (student_name, admission_date) VALUES ('Alice', '2025-09-01');  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM StudentAudit;  
+-----+-----+-----+-----+  
| audit_id | student_id | action | action_time |  
+-----+-----+-----+-----+  
| 1 | 1 | INSERT | 2025-09-02 09:54:35 |  
+-----+-----+-----+-----+  
1 row in set (0.00 sec)
```

```
-> IF done THEN
->     LEAVE read_loop;
-> END IF;
->
-> -- Here you can process each row
-> -- For demonstration, let's just select the student name
-> SELECT CONCAT('Processing student: ', sname) AS info;
-> END LOOP;
->
-> CLOSE cur_students;
-> END //
```

Query OK, 0 rows affected (0.03 sec)

```
mysql>
mysql> DELIMITER ;
mysql> CALL ProcessStudents();
```

```
+-----+
| info  |
+-----+
| Processing student: Alice |
+-----+
```

1 row in set (0.01 sec)

```
+-----+
| info  |
+-----+
| Processing student: Bob |
+-----+
```

1 row in set (0.07 sec)

```
+-----+
| info  |
+-----+
| Processing student: Charlie |
+-----+
```

1 row in set (0.10 sec)

Query OK, 0 rows affected (0.11 sec)

```
mysql>
```

```
+-----+-----+-----+-----+
| bus_number | destination | departure_time | seats |
+-----+-----+-----+-----+
| BUS102     | Chennai    | 2025-09-02 10:30:00 | 50 |
+-----+-----+-----+-----+
1 row in set (0.02 sec)
```

```
mysql> SELECT U.name, B.bus_number, B.source, B.destination,
->         T.seat_no, P.payment_status, Q.qr_data
-> FROM Tickets T
-> JOIN Users U ON T.user_id = U.user_id
-> JOIN Buses B ON T.bus_id = B.bus_id
-> JOIN Payments P ON T.ticket_id = P.ticket_id
-> JOIN QR_Codes Q ON T.ticket_id = Q.ticket_id
-> WHERE T.ticket_id = 1;
```

```
+-----+-----+-----+-----+-----+-----+-----+
| name      | bus_number | source   | destination | seat_no | payment_status | qr_data
+-----+-----+-----+-----+-----+-----+-----+
| Amit Sharma | BUS101    | Hyderabad | Bangalore   | 12      | Completed      | QR:BUS101-Seat12-User1-Ticket1
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> UPDATE Tickets
-> SET status='Cancelled'
-> WHERE ticket_id=1;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> SELECT seats - COUNT(T.ticket_id) AS available_seats
-> FROM Buses B
-> LEFT JOIN Tickets T ON B.bus_id = T.bus_id AND T.status='Booked'
-> WHERE B.bus_id = 1
-> GROUP BY B.bus_id;
```

```
+-----+
| available_seats |
+-----+
| 40              |
+-----+
```

```
mysql>
mysql> INSERT INTO Messages (text) VALUES
-> ('Hello World'),
-> ('MySQL String Functions'),
-> ('ChatGPT is awesome'),
-> ('Data Science');

Query OK, 4 rows affected (0.01 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

```

--+-----+-----+-----+-----+-----+-----+-----+
| text                | replaced_text      | repeated_text      | reversed_text      | right_five_chars   | left_five_chars    | right_padded       | left_padded       |
+-----+-----+-----+-----+-----+-----+-----+
| Hello World         | Hello_World        | Ha! Ha!            | dlrow olleH       | World              | Hello              | Hello World***** | *****Hello Worl
d |
| MySQL String Functions | MySQL_String_Functions | Ha! Ha!            | snoitcnuF gnirts LQSyM | tions              | MySQL              | MySQL String Functio | MySQL String Functi
o |
| ChatGPT is awesome   | ChatGPT_is_awesome  | Ha! Ha!            | emosewa si TPGtahC | esome              | ChatG              | ChatGPT is awesome** | **ChatGPT is awesom
e |
| Data Science         | Data_Science        | Ha! Ha!            | ecneicS ataD      | ience              | Data               | Data Science***** | *****Data Scienc
e |
+-----+-----+-----+-----+-----+-----+-----+
--+
4 rows in set (0.00 sec)

```



```
mysql> DELIMITER //
```

```
mysql>
mysql> CREATE PROCEDURE AddStudent (
  ->     IN name VARCHAR(100),
  ->     IN course_name VARCHAR(50),
  ->     IN score INT
  -> )
  -> BEGIN
  ->     INSERT INTO Students (student_name, course, marks)
  ->     VALUES (name, course_name, score);
  -> END //
```

Query OK, 0 rows affected (0.01 sec)

```
mysql>
```

```
mysql> DELIMITER ;
```

```
mysql> CALL AddStudent('Alice', 'Computer Science', 85);
```

Query OK, 1 row affected (0.01 sec)

```
mysql> CALL AddStudent('Bob', 'Mechanical Engineering', 78);
```

Query OK, 1 row affected (0.01 sec)

```
mysql> CALL AddStudent('Charlie', 'Civil Engineering', 92);
```

Query OK, 1 row affected (0.01 sec)

```
mysql> SELECT * FROM Students;
```

| student_id | student_name | course                 | marks |
|------------|--------------|------------------------|-------|
| 1          | Alice        | Computer Science       | 85    |
| 2          | Bob          | Mechanical Engineering | 78    |
| 3          | Charlie      | Civil Engineering      | 92    |



```
mysql> select database();
```

|            |
|------------|
| database() |
| strings    |

1 row in set (0.00 sec)

```
mysql> SELECT CONCAT('Bus', SPACE(5), 'Ticket') AS result;
```

|                 |
|-----------------|
| result          |
| Bus      Ticket |

1 row in set (0.00 sec)

```
mysql> SELECT SUBSTR('HyderabadToBangalore', 1, 9) AS result;
```

|           |
|-----------|
| result    |
| Hyderabad |

1 row in set (0.00 sec)

```
mysql> SELECT UPPER('bus ticket booking') AS result;
```

|                    |
|--------------------|
| result             |
| BUS TICKET BOOKING |

1 row in set (0.01 sec)

```
mysql> SELECT LOWER('BUS TICKET BOOKING') AS result;
```

|                    |
|--------------------|
| result             |
| bus ticket booking |

1 row in set (0.01 sec)

```
mysql> SELECT TRIM('    BusTicket    ') AS result;
```

|           |
|-----------|
| result    |
| BusTicket |

1 row in set (0.00 sec)