



INSTITUTO TECNOLÓGICO SUPERIOR DE JEREZ



INGENIERÍA EN SISTEMAS COMPUTACIONALES

TALLER DE BASES DE DATOS

9° SEMESTRE

I.S.C. SALVADOR ACEVEDO SANDOVAL

“TRIGGERS”

ALBAR DE LA TORRE GARCÍA

No. Control: 16070122

Correo: [albar00@hotmail.com](mailto:albar00@hotmail.com)

JEREZ ZACATECAS

10 DE DICIEMBRE DEL 2020

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

Query OK, 0 rows affected (0.12 sec)

mysql> CREATE TABLE employees_audit (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   emp_no INT NOT NULL,
  ->   last_name VARCHAR(50) NOT NULL,
  ->   changedat DATETIME DEFAULT NULL,
  ->   action VARCHAR(50) DEFAULT NULL
  -> );
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE TRIGGER before_employee_update
  ->   BEFORE UPDATE ON employees
  ->   FOR EACH ROW
  ->   INSERT INTO employees_audit
  ->   SET action = 'update',
  ->       emp_no = OLD.emp_no,
  ->       last_name = last_name,
  ->       changedat = NOW();
Query OK, 0 rows affected (0.02 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> SHOW TRIGGERS;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Trigger                               | Event | Table | Statement                               | sql_mode | Definer           | character_set_client | coll |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| before_employee_update | UPDATE | employees | INSERT INTO employees_audit
SET action = 'update',
  employeeNumber = employeeNumber,
  lastname = lastname,
  changedat = NOW() | BEFORE | 2020-12-10 14:06:40.18 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION | root@localhost | cp850 | cp850_general_ci |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.02 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> UPDATE employees
  -> SET
  ->   last_name = 'Facello2'
  -> WHERE
  ->   emp_no = 10001;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT * FROM employees_audit;
+----+-----+-----+-----+-----+
| id | emp_no | last_name | changedat | action |
+----+-----+-----+-----+-----+
| 1  | 10001 | Facello2  | 2020-12-10 14:17:48 | update |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> CREATE TABLE billings (
  ->   billingNo INT AUTO_INCREMENT,
  ->   customerNo INT,
  ->   billingDate DATE,
  ->   amount DEC(10 , 2 ),
  ->   PRIMARY KEY (billingNo)
  -> );
Query OK, 0 rows affected (0.12 sec)

mysql>
```

```
Selecionar C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
Query OK, 0 rows affected (0.12 sec)

mysql> DELIMITER $$
mysql> CREATE TRIGGER before_billing_update
  ->   BEFORE UPDATE
  ->   ON billings FOR EACH ROW
  ->   BEGIN
  ->     IF new.amount > old.amount * 10 THEN
  ->       SIGNAL SQLSTATE '45000'
  ->       SET MESSAGE_TEXT = 'New amount cannot be 10 times greater than the current amount.';
  ->     END IF;
  ->   END$$
Query OK, 0 rows affected (0.02 sec)

mysql> DELIMITER ;
mysql> SHOW TRIGGERS;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Trigger | Event | Table | Statement | Timing | Created | sql_mode |
+-----+-----+-----+-----+-----+-----+-----+
| before_billing_update | UPDATE | billings | BEGIN
  IF new.amount > old.amount * 10 THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'New amount cannot be 10 times greater than the current amount.';
  END IF;
END | BEFORE | 2020-12-10 14:19:35.19 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900
_ai_ci |
| before_employee_update | UPDATE | employees | INSERT INTO employees_audit
  SET action = 'update',
  emp_no = OLD.emp_no,
  last_name = last_name,
  changedat = NOW() | BEFORE | 2020-12-10 14:14:31.58 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
2 rows in set (0.01 sec)

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

mysql> DROP TRIGGER before_billing_update;
ERROR 1360 (HY000): Trigger does not exist
mysql> SHOW TRIGGERS;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Trigger | Event | Table | Statement | Timing | Created | sql_mode | Definer | character_set_client | collation_connection |
+-----+-----+-----+-----+-----+-----+-----+-----+
| before_employee_update | UPDATE | employees | INSERT INTO employees_audit
  SET action = 'update',
  emp_no = OLD.emp_no,
  last_name = last_name,
  changedat = NOW() | BEFORE | 2020-12-10 14:14:31.58 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DROP TABLE IF EXISTS WorkCenters;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
mysql> CREATE TABLE WorkCenters (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100) NOT NULL,
  ->   capacity INT NOT NULL
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql> DROP TABLE IF EXISTS WorkCenterStats;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
mysql> CREATE TABLE WorkCenterStats(
  ->   totalCapacity INT NOT NULL
  -> );
Query OK, 0 rows affected (0.02 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

Query OK, 0 rows affected (0.02 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE TRIGGER before_workcenters_insert
  -> BEFORE INSERT
  -> ON WorkCenters FOR EACH ROW
  -> BEGIN
  ->   DECLARE rowcount INT;
  ->
  ->   SELECT COUNT(*)
  ->   INTO rowcount
  ->   FROM WorkCenterStats;
  ->
  ->   IF rowcount > 0 THEN
  ->     UPDATE WorkCenterStats
  ->     SET totalCapacity = totalCapacity + new.capacity;
  ->   ELSE
  ->     INSERT INTO WorkCenterStats(totalCapacity)
  ->     VALUES(new.capacity);
  ->   END IF;
  -> END $$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELIMITER ;
mysql> INSERT INTO WorkCenters(name, capacity)
-> VALUES('Mold Machine',100);
Query OK, 1 row affected (0.02 sec)

mysql> SELECT * FROM WorkCenterStats;
+-----+
| totalCapacity |
+-----+
|          100 |
+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO WorkCenters(name, capacity)
-> VALUES('Packing',200);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM WorkCenterStats;
+-----+
| totalCapacity |
+-----+
|          300 |
+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DROP TABLE IF EXISTS members;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
mysql> CREATE TABLE members (
->   id INT AUTO_INCREMENT,
->   name VARCHAR(100) NOT NULL,
->   email VARCHAR(255),
->   birthDate DATE,
->   PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> DROP TABLE IF EXISTS reminders;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql>
mysql> CREATE TABLE reminders (
->   id INT AUTO_INCREMENT,
->   memberId INT,
->   message VARCHAR(255) NOT NULL,
->   PRIMARY KEY (id , memberId)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELIMITER $$
mysql>
mysql> CREATE TRIGGER after_members_insert
-> AFTER INSERT
-> ON members FOR EACH ROW
-> BEGIN
->     IF NEW.birthDate IS NULL THEN
->         INSERT INTO reminders(memberId, message)
->         VALUES(new.id,CONCAT('Hi ', NEW.name, ', please update your date of birth.'));
->     END IF;
-> END$$
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELIMITER ;
mysql> INSERT INTO members(name, email, birthDate)
-> VALUES
->     ('John Doe', 'john.doe@example.com', NULL),
->     ('Jane Doe', 'jane.doe@example.com', '2000-01-01');
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM members;
+----+-----+-----+-----+
| id | name  | email                | birthDate |
+----+-----+-----+-----+
| 1  | John Doe | john.doe@example.com | NULL      |
| 2  | Jane Doe | jane.doe@example.com | 2000-01-01 |
+----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM reminders;
+----+-----+-----+
| id | memberId | message |
+----+-----+-----+
| 1  | 1        | Hi John Doe, please update your date of birth. |
+----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> CREATE TABLE sales (
  ->   id INT AUTO_INCREMENT,
  ->   product VARCHAR(100) NOT NULL,
  ->   quantity INT NOT NULL DEFAULT 0,
  ->   fiscalYear SMALLINT NOT NULL,
  ->   fiscalMonth TINYINT NOT NULL,
  ->   CHECK(fiscalMonth >= 1 AND fiscalMonth <= 12),
  ->   CHECK(fiscalYear BETWEEN 2000 and 2050),
  ->   CHECK (quantity >=0),
  ->   UNIQUE(product, fiscalYear, fiscalMonth),
  ->   PRIMARY KEY(id)
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO sales(product, quantity, fiscalYear, fiscalMonth)
  -> VALUES
  ->   ('2003 Harley-Davidson Eagle Drag Bike',120, 2020,1),
  ->   ('1969 Corvair Monza', 150,2020,1),
  ->   ('1970 Plymouth Hemi Cuda', 200,2020,1);
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM sales;
+-----+-----+-----+-----+-----+
| id | product                                | quantity | fiscalYear | fiscalMonth |
+-----+-----+-----+-----+-----+
| 1 | 2003 Harley-Davidson Eagle Drag Bike | 120      | 2020      | 1           |
| 2 | 1969 Corvair Monza                    | 150      | 2020      | 1           |
| 3 | 1970 Plymouth Hemi Cuda              | 200      | 2020      | 1           |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELIMITER $$
mysql> CREATE TRIGGER before_sales_update
  -> BEFORE UPDATE
  -> ON sales FOR EACH ROW
  -> BEGIN
  ->   DECLARE errorMessage VARCHAR(255);
  ->   SET errorMessage = CONCAT('The new quantity ',
  ->                               NEW.quantity,
  ->                               ' cannot be 3 times greater than the current quantity ',
  ->                               OLD.quantity);
  ->
  ->   IF new.quantity > old.quantity * 3 THEN
  ->     SIGNAL SQLSTATE '45000'
  ->     SET MESSAGE_TEXT = errorMessage;
  ->   END IF;
  -> END $$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql>
mysql> UPDATE sales
  -> SET quantity = 150
  -> WHERE id = 1;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM sales;
+-----+-----+-----+-----+-----+
| id | product                                | quantity | fiscalYear | fiscalMonth |
+-----+-----+-----+-----+-----+
| 1 | 2003 Harley-Davidson Eagle Drag Bike | 150      | 2020      | 1           |
| 2 | 1969 Corvair Monza                    | 150      | 2020      | 1           |
| 3 | 1970 Plymouth Hemi Cuda              | 200      | 2020      | 1           |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE sales
  -> SET quantity = 500
  -> WHERE id = 1;
ERROR 1644 (45000): The new quantity 500 cannot be 3 times greater than the current quantity 150
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
mysql> CREATE TABLE Sales (
  ->   id INT AUTO_INCREMENT,
  ->   product VARCHAR(100) NOT NULL,
  ->   quantity INT NOT NULL DEFAULT 0,
  ->   fiscalYear SMALLINT NOT NULL,
  ->   fiscalMonth TINYINT NOT NULL,
  ->   CHECK(fiscalMonth >= 1 AND fiscalMonth <= 12),
  ->   CHECK(fiscalYear BETWEEN 2000 and 2050),
  ->   CHECK (quantity >=0),
  ->   UNIQUE(product, fiscalYear, fiscalMonth),
  ->   PRIMARY KEY(id)
  -> );
Query OK, 0 rows affected (0.05 sec)

mysql> INSERT INTO Sales(product, quantity, fiscalYear, fiscalMonth)
  -> VALUES
  ->   ('2001 Ferrari Enzo',140, 2021,1),
  ->   ('1998 Chrysler Plymouth Prowler', 110,2021,1),
  ->   ('1913 Ford Model T Speedster', 120,2021,1);
Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Sales;
+-----+-----+-----+-----+-----+
| id | product                | quantity | fiscalYear | fiscalMonth |
+-----+-----+-----+-----+-----+
| 1  | 2001 Ferrari Enzo      | 140     | 2021      | 1           |
| 2  | 1998 Chrysler Plymouth Prowler | 110     | 2021      | 1           |
| 3  | 1913 Ford Model T Speedster | 120     | 2021      | 1           |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
mysql>
mysql> CREATE TABLE SalesChanges (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   salesId INT,
  ->   beforeQuantity INT,
  ->   afterQuantity INT,
  ->   changedAt TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
  -> );
Query OK, 0 rows affected (0.09 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE TRIGGER after_sales_update
  -> AFTER UPDATE
  -> ON sales FOR EACH ROW
  -> BEGIN
  ->   IF OLD.quantity <> new.quantity THEN
  ->     INSERT INTO SalesChanges(salesId,beforeQuantity, afterQuantity)
  ->     VALUES(OLD.id, OLD.quantity, new.quantity);
  ->   END IF;
  -> END$$
Query OK, 0 rows affected (0.07 sec)

mysql>
mysql> DELIMITER ;
mysql>
```



```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql>
mysql> DELIMITER ;
mysql> UPDATE Sales
  -> SET quantity = 350
  -> WHERE id = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM SalesChanges;
+-----+-----+-----+-----+-----+
| id | salesId | beforeQuantity | afterQuantity | changedAt |
+-----+-----+-----+-----+-----+
| 1 | 1 | 140 | 350 | 2020-12-10 14:34:42 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> UPDATE Sales
  -> SET quantity = CAST(quantity * 1.1 AS UNSIGNED);
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> SELECT * FROM SalesChanges;
+-----+-----+-----+-----+-----+
| id | salesId | beforeQuantity | afterQuantity | changedAt |
+-----+-----+-----+-----+-----+
| 1 | 1 | 140 | 350 | 2020-12-10 14:34:42 |
| 2 | 1 | 350 | 385 | 2020-12-10 14:34:53 |
| 3 | 2 | 110 | 121 | 2020-12-10 14:34:53 |
| 4 | 3 | 120 | 132 | 2020-12-10 14:34:53 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELIMITER $$
mysql>
mysql> CREATE TRIGGER before_salaries_delete
  -> BEFORE DELETE
  -> ON salaries FOR EACH ROW
  -> BEGIN
  -> INSERT INTO SalaryArchives(employeeNumber,validFrom,amount)
  -> VALUES(OLD.employeeNumber,OLD.validFrom,OLD.amount);
  -> END$$
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DELETE FROM salaries
  -> WHERE employeeNumber = 1002;
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM SalaryArchives;
+-----+-----+-----+-----+-----+
| id | employeeNumber | validFrom | amount | deletedAt |
+-----+-----+-----+-----+-----+
| 1 | 1002 | 2000-01-01 | 50000.00 | 2020-12-10 14:39:05 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> DELETE FROM salaries;
Query OK, 2 rows affected (0.01 sec)

mysql> SELECT * FROM SalaryArchives;
+-----+-----+-----+-----+-----+
| id | employeeNumber | validFrom | amount | deletedAt |
+-----+-----+-----+-----+-----+
| 1 | 1002 | 2000-01-01 | 50000.00 | 2020-12-10 14:39:05 |
| 2 | 1056 | 2000-01-01 | 60000.00 | 2020-12-10 14:39:18 |
| 3 | 1076 | 2000-01-01 | 70000.00 | 2020-12-10 14:39:18 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> CREATE TABLE Salaries (
  ->     employeeNumber INT PRIMARY KEY,
  ->     salary DECIMAL(10,2) NOT NULL DEFAULT 0
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO Salaries(employeeNumber,salary)
  -> VALUES
  ->     (1002,5000),
  ->     (1056,7000),
  ->     (1076,8000);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> DROP TABLE IF EXISTS SalaryBudgets;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
mysql> CREATE TABLE SalaryBudgets(
  ->     total DECIMAL(15,2) NOT NULL
  -> );
Query OK, 0 rows affected (0.05 sec)

mysql> INSERT INTO SalaryBudgets(total)
  -> SELECT SUM(salary)
  -> FROM Salaries;
Query OK, 1 row affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM SalaryBudgets;
+-----+
| total |
+-----+
| 20000.00 |
+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql>
mysql> CREATE TRIGGER after_salaries_delete
  -> AFTER DELETE
  -> ON Salaries FOR EACH ROW
  -> UPDATE SalaryBudgets
  -> SET total = total - old.salary;
Query OK, 0 rows affected (0.01 sec)

mysql> DELETE FROM Salaries
  -> WHERE employeeNumber = 1002;
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM SalaryBudgets;
+-----+
| total |
+-----+
| 15000.00 |
+-----+
1 row in set (0.00 sec)

mysql> DELETE FROM Salaries;
Query OK, 2 rows affected (0.01 sec)

mysql> SELECT * FROM SalaryBudgets;
+-----+
| total |
+-----+
| 0.00 |
+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> DROP TABLE IF EXISTS accounts;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
mysql> CREATE TABLE accounts (
  ->     accountId INT AUTO_INCREMENT,
  ->     name VARCHAR(100) NOT NULL,
  ->     amount DECIMAL(10, 2) NOT NULL,
  ->     PRIMARY KEY (accountId),
  ->     CHECK(amount >= 0)
  -> );
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO accounts(name, amount)
  -> VALUES
  ->     ('John Doe', 1000),
  ->     ('Jane Bush', 500);
Query OK, 2 rows affected (0.08 sec)
Records: 2 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM accounts;
+-----+-----+-----+
| accountId | name   | amount |
+-----+-----+-----+
|          1 | John Doe | 1000.00 |
|          2 | Jane Bush | 500.00 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE Withdraw(
  ->     fromAccountId INT,
  ->     withdrawAmount DEC(10,2)
  -> )
  -> BEGIN
  ->     IF withdrawAmount <= 0 THEN
  ->         SIGNAL SQLSTATE '45000'
  ->         SET MESSAGE_TEXT = 'The withdrawal amount must be greater than zero';
  ->     END IF;
  ->
  ->     UPDATE accounts
  ->     SET amount = amount - withdrawAmount
  ->     WHERE accountId = fromAccountId;
  -> END$$
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> DELIMITER ;
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE CheckWithdrawal(
  ->     fromAccountId INT,
  ->     withdrawAmount DEC(10,2)
  -> )
  -> BEGIN
  ->     DECLARE balance DEC(10,2);
  ->     DECLARE withdrawableAmount DEC(10,2);
  ->     DECLARE message VARCHAR(255);
  ->
  ->     -- get current balance of the account
  ->     SELECT amount
  ->     INTO balance
  ->     FROM accounts
  ->     WHERE accountId = fromAccountId;
  ->
  ->     -- Set minimum balance
  ->     SET withdrawableAmount = balance - 25;
  ->
  ->     IF withdrawAmount > withdrawableAmount THEN
  ->         SET message = CONCAT('Insufficient amount, the maximum withdrawable is ', withdrawableAmount);
  ->         SIGNAL SQLSTATE '45000'
  ->         SET MESSAGE_TEXT = message;
  ->     END IF;
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p
mysql> DELIMITER ;
mysql> DELIMITER $$
mysql>
mysql> CREATE TRIGGER before_accounts_update
  -> BEFORE UPDATE
  -> ON accounts FOR EACH ROW
  -> BEGIN
  ->     CALL CheckWithdrawal (
  ->         OLD.accountId,
  ->         OLD.amount - NEW.amount
  ->     );
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
mysql> CALL withdraw(1, 400);
Query OK, 1 row affected (0.07 sec)

mysql> SELECT * FROM accounts
  -> WHERE accountId = 1;
+-----+-----+-----+
| accountId | name   | amount |
+-----+-----+-----+
|         1 | John Doe | 600.00 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

mysql> CALL withdraw(1, 600);
ERROR 1644 (45000): Insufficient amount, the maximum withdrawable is 575.00
mysql> CALL withdraw(1, 575);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT *
      -> FROM accounts
      -> WHERE accountId = 1;
+-----+-----+-----+
| accountId | name   | amount |
+-----+-----+-----+
|         1 | John Doe | 25.00 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql -P3307 -u root -p

1 row in set (0.00 sec)

mysql> SHOW TRIGGERS
      -> FROM employees;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Trigger                               | Event | Table | Statement | Definer | character_set_client | collation_connection | Database Collation | Timing | Created | sql_mode |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| before_accounts_update                | UPDATE | accounts | BEGIN
  CALL CheckWithdrawal (
    OLD.accountId,
    OLD.amount - NEW.amount
  );
END | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci | BEFORE | 2020-12-10 14:46:41.46 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION |
| before_employee_update                 | UPDATE | employees | INSERT INTO employees_audit
  SET action = 'update',
      emp_no = OLD.emp_no,
      last_name = last_name,
      changedat = NOW()
END | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci | BEFORE | 2020-12-10 14:14:31.58 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION |
| after_members_insert                   | INSERT | members | BEGIN
  IF NEW.birthDate IS NULL THEN
    INSERT INTO reminders(memberId, message)
    VALUES(new.id,CONCAT('Hi ', NEW.name, ', please update your date of birth.'));
  END IF;
END | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci | AFTER | 2020-12-10 14:26:19.82 | STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION |
```