SQL Data Science for Beginners

Data Manipulation: INSERT, UPDATE, DELETE, and REPLACE

Let me explain these fundamental MySQL operations with practical examples using a sample database table.

Sample Table Structure

First, let's create a sample table to work with:

```
CREATE TABLE employees (
    id INT AUTO_INCREMENT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    email VARCHAR(100) UNIQUE,
    department VARCHAR(50),
    salary DECIMAL(10, 2),
    hire_date DATE
);
```

1. INSERT INTO - Adding New Data

The `INSERT INTO` statement adds new records to a table.

Basic Syntax:

```
INSERT INTO table_name (column1, column2, ...)

VALUES (value1, value2, ...);
```

Examples:

```
-- Inserting a single record
INSERT INTO employees (first_name, last_name, email, department,
salary, hire_date)
VALUES ('John', 'Doe', 'john.doe@example.com', 'IT', 75000.00,
'2020-05-15');

-- Inserting multiple records at once
INSERT INTO employees (first_name, last_name, email, department,
salary, hire_date)
VALUES
    ('Jane', 'Smith', 'jane.smith@example.com', 'HR', 65000.00,
'2019-11-20'),
    ('Robert', 'Johnson', 'robert.j@example.com', 'Finance', 82000.00,
'2018-03-10'),
    ('Sarah', 'Williams', 'sarah.w@example.com', 'Marketing', 70000.00,
'2021-02-28');
```

2. UPDATE ... SET - Modifying Existing Data

The `UPDATE` statement modifies existing records in a table.

Basic Syntax:

```
UPDATE table_name

SET column1 = value1, column2 = value2, ...

WHERE condition;
```

Examples:

```
-- Giving a raise to John Doe
UPDATE employees
SET salary = 80000.00
WHERE first_name = 'John' AND last_name = 'Doe';

-- Promoting all IT department employees with a 10% raise
UPDATE employees
SET salary = salary * 1.10,
    department = 'Senior IT'
WHERE department = 'IT';

-- Updating email for a specific employee
UPDATE employees
SET email = 'john.doe.new@example.com'
WHERE id = 1;
```

Important: Always use a WHERE clause with UPDATE to avoid modifying all records accidentally.

3. DELETE FROM - Removing Data

The `DELETE` statement removes records from a table.

Basic Syntax:

```
DELETE FROM table_name

WHERE condition;
```

Examples:

```
-- Deleting a specific employee
DELETE FROM employees
WHERE id = 3;

-- Deleting all employees in HR department hired before 2020
DELETE FROM employees
WHERE department = 'HR' AND hire_date < '2020-01-01';

-- Deleting all records (use with extreme caution!)
DELETE FROM employees;
```

Warning: Omitting the WHERE clause will delete all records in the table.

4. REPLACE INTO - Insert or Replace Data

`REPLACE INTO` either inserts a new record or deletes and re-inserts a record if a primary key or unique index conflict occurs.

Basic Syntax:

```
REPLACE INTO table_name (column1, column2, ...)

VALUES (value1, value2, ...);
```

Examples:

```
-- This will insert a new record since email is unique and doesn't exist
REPLACE INTO employees (id, first_name, last_name, email, department,
salary, hire_date)
VALUES (5, 'Michael', 'Brown', 'michael.b@example.com', 'Sales', 68000.00,
'2022-01-10');

-- This will replace the record with id=2 because it already exists
```

SQL Data Science for Beginners

```
REPLACE INTO employees (id, first_name, last_name, email, department,
salary, hire_date)
VALUES (2, 'Jane', 'Smith-Jones', 'jane.smith@example.com', 'HR',
70000.00, '2019-11-20');
```