

Bài 08:

CRUD JAVA và MYSQL

Giảng Viên: ThS. Giang Hào Côn

8.1/ Giới Thiệu

- **CRUD** là viết tắt của **Create, Read, Update, Delete**, là các thao tác cơ bản nhất trong lập trình với cơ sở dữ liệu.
- Java là ngôn ngữ lập trình phổ biến được sử dụng để phát triển nhiều loại ứng dụng khác nhau.
- MySQL là hệ quản trị cơ sở dữ liệu quan hệ được sử dụng rộng rãi bởi vì tính dễ sử dụng, hiệu quả và miễn phí.
- Kết hợp Java và MySQL cho phép bạn tạo các ứng dụng có thể lưu trữ, truy xuất và thao tác dữ liệu một cách hiệu quả.

8.2/ Cài đặt môi trường

Để thực hiện các thao tác CRUD với Java và MySQL, ta cần:

- Cài đặt JDK (Java Development Kit)
- Cài đặt MySQL (Xampp Server)
- Cài đặt IDE (Integrated Development Environment) như Eclipse, IntelliJ IDEA
- Thư viện JDBC (Java Database Connectivity)

8.3/ Kết nối Java với Mysql

- Sử dụng JDBC (Java Database Connectivity)
- Cài đặt JDBC driver
- Viết code kết nối

Thay đổi database_name, username và password cho phù hợp. Port 3306 có thể bỏ qua.

```
import java.sql.*;

public class ConnectMySQL {

    public static Connection getConnection() throws SQLException {
        // Cấu hình kết nối
        String url = "jdbc:mysql://localhost:3306/database_name";
        String username = "root";
        String password = "";

        // Kết nối
        Connection connection = DriverManager.getConnection(url, username, password);

        // Thực hiện thao tác với database
        // ...
        return connection;
    }
}
```

8.4/ Tạo (Create)

- Sử dụng lệnh INSERT INTO
- Chỉ định tên bảng và các cột
- Cung cấp giá trị cho các cột

```
String sql = "INSERT INTO table_name (column1, column2) VALUES (?, ?)";  
  
PreparedStatement statement = connection.prepareStatement(sql);  
statement.setString(1, "value1");  
statement.setInt(2, 10);  
statement.executeUpdate();
```

8.5/ Đọc (Select)

- Sử dụng lệnh SELECT
- Chỉ định các cột muốn lấy
- Có thể sử dụng điều kiện where

```
String sql = "SELECT * FROM table_name";

Statement statement = connection.createStatement();
ResultSet resultSet = statement.executeQuery(sql);

while (resultSet.next()) {
    int id = resultSet.getInt("id");
    String name = resultSet.getString("name");
    // ...
}
```

8.6/ Cập nhật (Update)

- Sử dụng lệnh Update
- Chỉ định các cột muốn cập nhật
- Có thể sử dụng điều kiện where

```
String sql = "UPDATE `table_name` SET `column_name` = ?, ... WHERE `condition` = ?";  
PreparedStatement statement = connection.prepareStatement(sql);
```

```
// Set giá trị cho các tham số  
statement.setString(1, newValue1);  
//..  
statement.setString(n, newValueN);  
// Set điều kiện  
statement.setString(n + 1, conditionValue);  
  
statement.executeUpdate();
```






8.6/ Cập nhật (Update)

- Sử dụng lệnh Delete
- Có thể sử dụng điều kiện where

```
String sql = "DELETE FROM `table_name` WHERE `condition` = ?";  
PreparedStatement statement = connection.prepareStatement(sql);  
  
// Set điều kiện  
statement.setString(1, conditionValue);  
  
statement.executeUpdate();
```


Ví dụ

- Tạo CSDL tên là SampleDB và tạo bảng tên Users với cấu trúc như sau:

Column Name	Datatype
 user_id	INT(11)
 username	VARCHAR(45)
 password	VARCHAR(45)
 fullname	VARCHAR(45)
 email	VARCHAR(45)

```
create database SampleDB;

use SampleDB;

CREATE TABLE `users` (
  `user_id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(45) NOT NULL,
  `password` varchar(45) NOT NULL,
  `fullname` varchar(45) NOT NULL,
  `email` varchar(45) NOT NULL,
  PRIMARY KEY (`user_id`)
);
```

Ví dụ

- Để kết nối đến CSDL SampleDB với user là root và pass là null:

```
String dbURL = "jdbc:mysql://localhost/sampledb";
String username = "root";
String password = "";

try {
    Connection conn = DriverManager.getConnection(dbURL, username, password);

    if (conn != null) {
        System.out.println("Connected");
    }
} catch (SQLException ex) {
    ex.printStackTrace();
}
```

Ví dụ

- Thêm một người dùng mới vào bảng Users với các chi tiết sau:

```
String sql = "INSERT INTO Users (username, password, fullname, email) VALUES (?, ?, ?, ?)";
```

```
PreparedStatement statement = conn.prepareStatement(sql);  
statement.setString(1, "bill");  
statement.setString(2, "secretpass");  
statement.setString(3, "Bill Gates");  
statement.setString(4, "bill.gates@microsoft.com");
```

- username: *bill*
- password: *secretpass*
- fullname: *Bill Gates*
- email: *bill.gates@microsoft.com*

```
int rowsInserted = statement.executeUpdate();  
if (rowsInserted > 0) {  
    System.out.println("A new user was inserted successfully!");  
}
```

Ví dụ

- Truy vấn tất cả các bản ghi từ bảng Users và in ra chi tiết cho từng bản ghi:

```
String sql = "SELECT * FROM Users";

Statement statement = conn.createStatement();
ResultSet result = statement.executeQuery(sql);

int count = 0;

while (result.next()){
    String name = result.getString(2);
    String pass = result.getString(3);
    String fullname = result.getString("fullname");
    String email = result.getString("email");

    String output = "User #%d: %s - %s - %s - %s";
    System.out.println(String.format(output, ++count, name, pass, fullname, email));
}
```

Ví dụ

- Cập nhật thông tin người dùng có tên là “bill”:

```
String sql = "UPDATE Users SET password=?, fullname=?, email=? WHERE username=?";
```

```
PreparedStatement statement = conn.prepareStatement(sql);  
statement.setString(1, "123456789");  
statement.setString(2, "William Henry Bill Gates");  
statement.setString(3, "bill.gates@microsoft.com");  
statement.setString(4, "bill");
```

```
int rowsUpdated = statement.executeUpdate();  
if (rowsUpdated > 0) {  
    System.out.println("An existing user was updated successfully!");  
}
```

Ví dụ

- Xoá thông tin người dùng có tên là “bill”:

```
String sql = "DELETE FROM Users WHERE username=?";
```

```
PreparedStatement statement = conn.prepareStatement(sql);  
statement.setString(1, "bill");
```

```
int rowsDeleted = statement.executeUpdate();  
if (rowsDeleted > 0) {  
    System.out.println("A user was deleted successfully!");  
}
```

CURD Operation Swing MySQL

User Management System

Id

Name

Age

City

ID	NAME	AGE	CITY
1	Ram Kumar	25	Salem
2	Giang Hào Côn	25	Salem
3	Rakesh	25	Salem