

REPORT ASSIGNMENT LAB 4

Student Name: Nguyễn Tiến Anh

Student ID: ITITDK22128

Course: Web Application Development

Lab 3: JavaScript Fundamentals

Github: https://github.com/TienAnh0108/Lab4_Exercise.git

Ex 1.2:

```
conn = DriverManager.getConnection(  
    "jdbc:mysql://localhost:3306/student_management",  
    "root",  
    "Tienanh0108!"  
);
```

The line uses the static method `getConnection()` from the `DriverManager` class to create a `Connection` object (`conn`).

`//localhost:3306`: Defines the location and port of the MySQL server. `localhost` means the server is running on the same machine as the application, and `3306` is the standard MySQL port.

Second line: `root` – username of mysql

Third line: `Tienanh0108!` – password of mysql

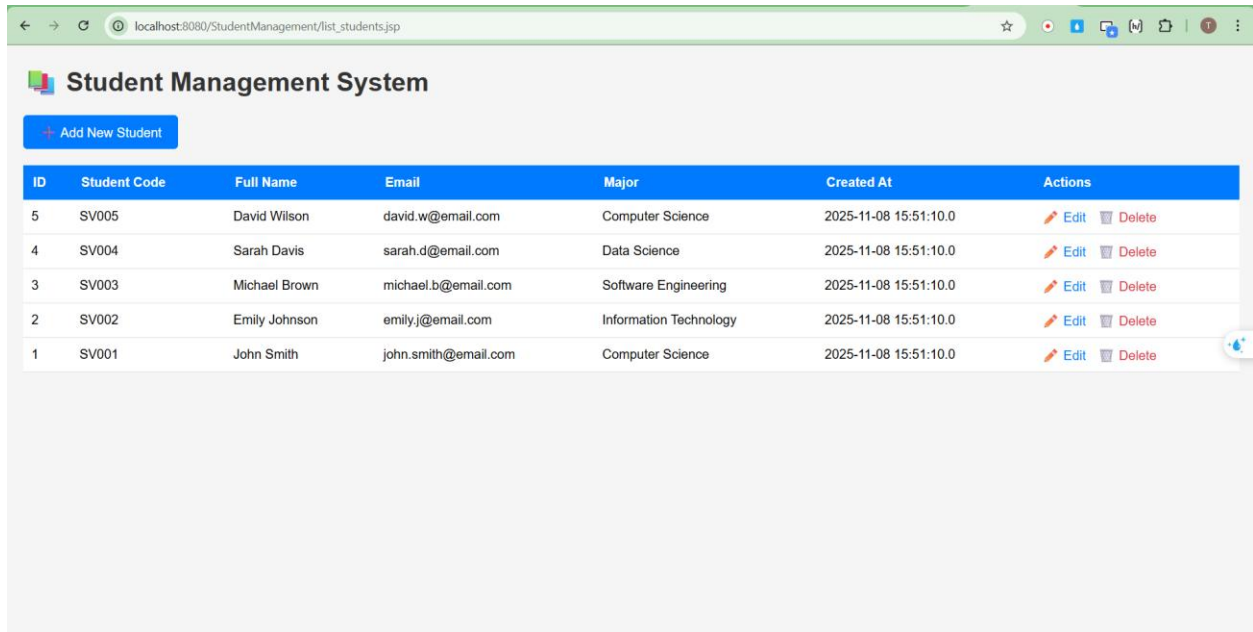
```
stmt = conn.createStatement();  
String sql = "SELECT * FROM students ORDER BY id DESC";  
rs = stmt.executeQuery(sql);
```

`stmt = conn.createStatement()`: This line creates a `Statement` object (`stmt`) using the established database connection (`conn`).

`String sql = "SELECT * FROM students ORDER BY id DESC"`: This line defines a standard SQL query string and stores it in a variable named `sql`.

`rs = stmt.executeQuery(sql)`: This is the execution command. It **sends the SQL query** defined in the `sql` string to the database via the `Statement` object (`stmt`).

Result:

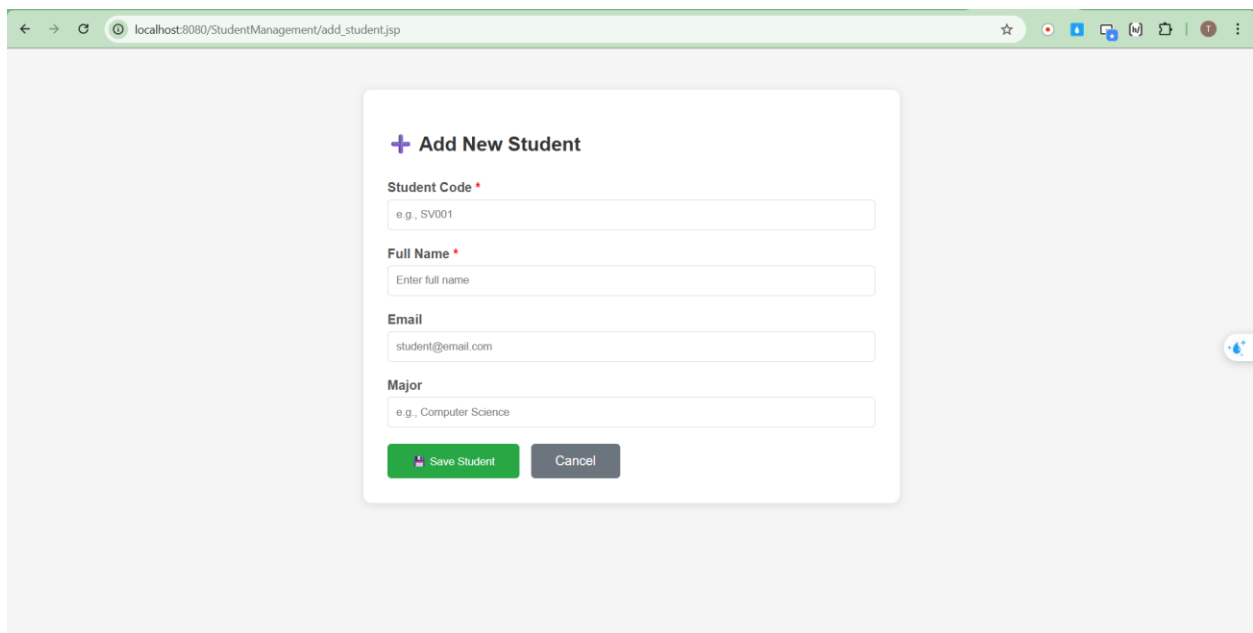


ID	Student Code	Full Name	Email	Major	Created At	Actions
5	SV005	David Wilson	david.w@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	2025-11-08 15:51:10.0	Edit Delete
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	2025-11-08 15:51:10.0	Edit Delete
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	2025-11-08 15:51:10.0	Edit Delete
1	SV001	John Smith	john.smith@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete

Ex 2.1:

```
<a href="add_student.jsp" class="btn">+ Add New Student</a>
```

This HTML snippet creates a hyperlink (a) that functions as a button to navigate the user to the form for adding a new student.



+ Add New Student

Student Code *

Full Name *

Email

Major

[Save Student](#) [Cancel](#)

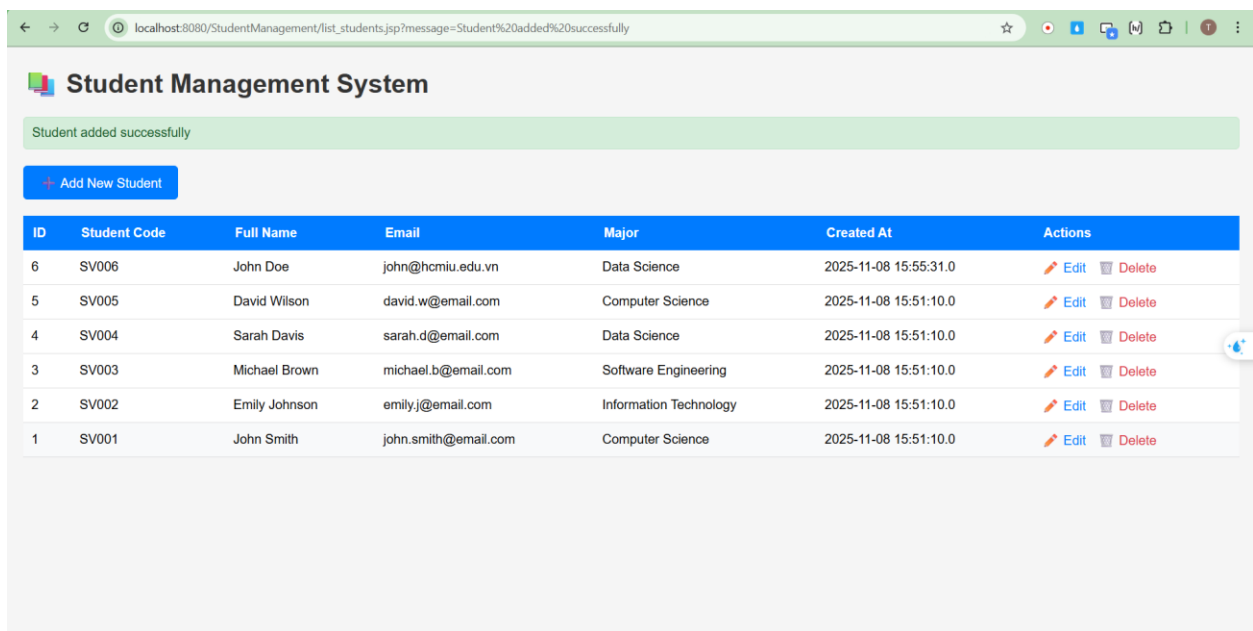
Ex 2.2:

```
if (studentCode == null || studentCode.trim().isEmpty() ||
    fullName == null || fullName.trim().isEmpty()) {
    response.sendRedirect("add_student.jsp?error=Required fields are missing");
    return;
}
```

The code checks two required variables, `studentCode` and `fullName`, to see if either of them is missing or empty:

- `studentCode == null || studentCode.trim().isEmpty()`:
 - `studentCode == null`: Checks if the parameter was not received at all (e.g., the input field name was misspelled in the form).
 - `studentCode.trim().isEmpty()`: Checks if the received string, after removing leading and trailing whitespace (`trim()`), is empty. This catches inputs where the user only typed spaces.
- `||` (OR operator): If the condition for `studentCode` is true OR the condition for `fullName` is true, the entire block executes.
- `fullName == null || fullName.trim().isEmpty()`: Performs the exact same check for the `fullName` field.

Testcase 1:



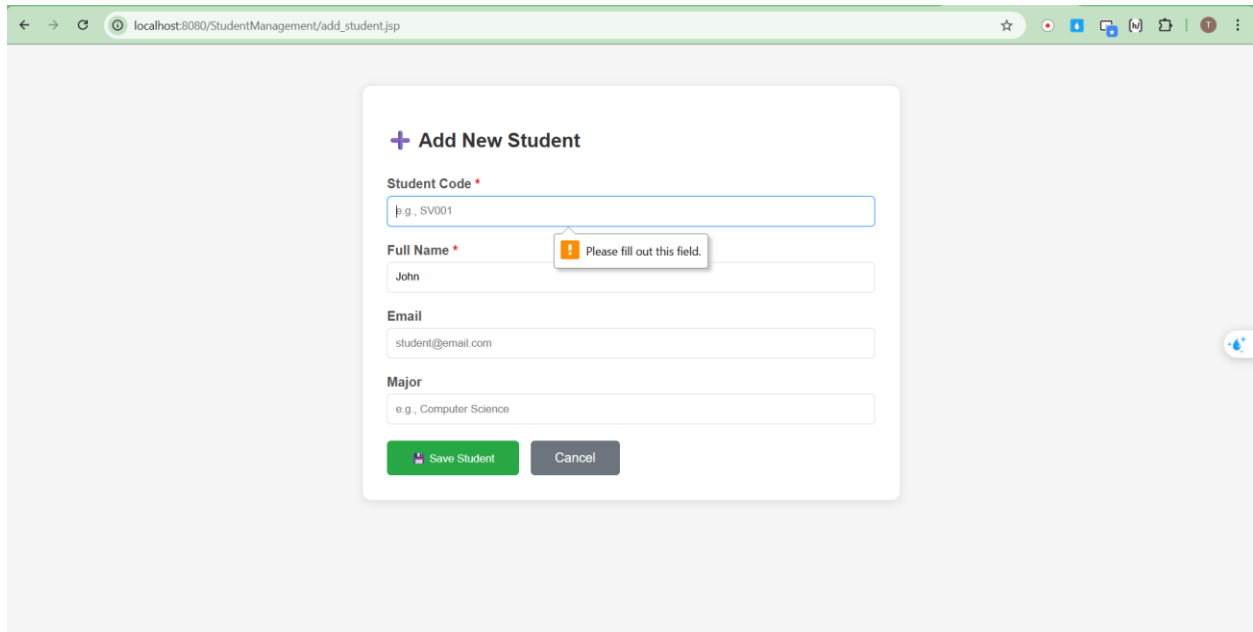
Student Management System

Student added successfully

[Add New Student](#)

ID	Student Code	Full Name	Email	Major	Created At	Actions
6	SV006	John Doe	john@hcmiu.edu.vn	Data Science	2025-11-08 15:55:31.0	Edit Delete
5	SV005	David Wilson	david.w@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	2025-11-08 15:51:10.0	Edit Delete
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	2025-11-08 15:51:10.0	Edit Delete
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	2025-11-08 15:51:10.0	Edit Delete
1	SV001	John Smith	john.smith@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete

Testcase 2:



localhost:8080/StudentManagement/add_student.jsp


+ Add New Student

Student Code *
e.g., SV001

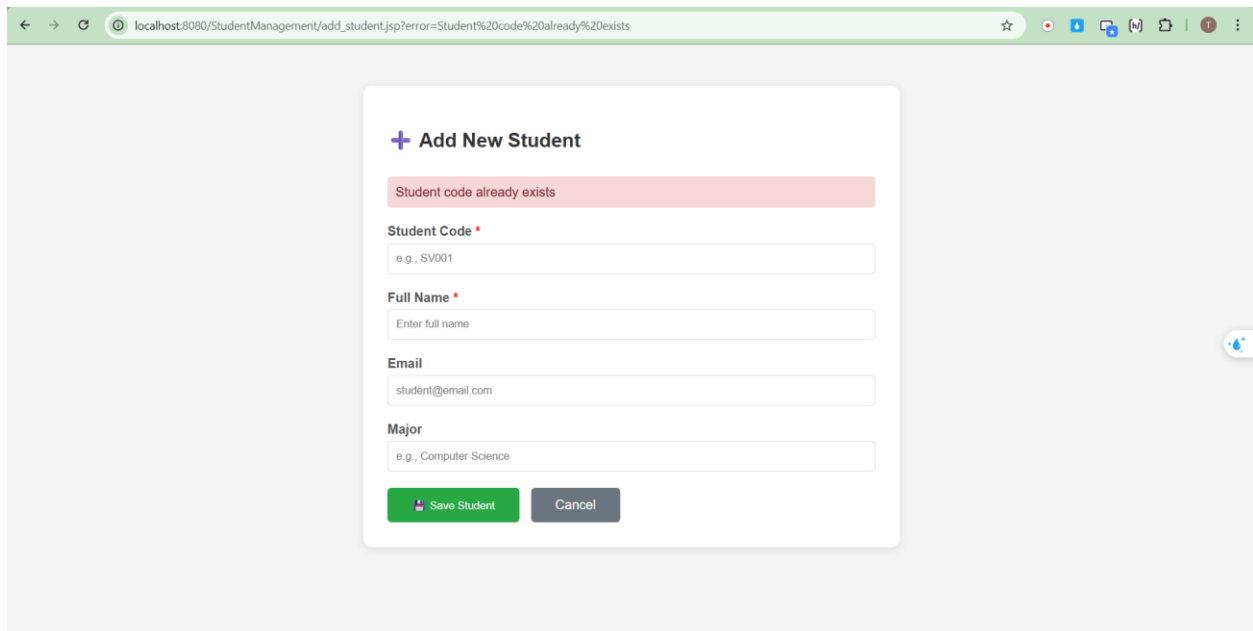
Full Name *
John

Email
student@email.com

Major
e.g., Computer Science

 Save Student Cancel

Testcase 3:



localhost:8080/StudentManagement/add_student.jsp?error=Student%20code%20already%20exists

+ Add New Student


Student code already exists

Student Code *
e.g., SV001

Full Name *
Enter full name

Email
student@email.com

Major
e.g., Computer Science

 Save Student Cancel

Ex 3.1:

```
try {  
    Class.forName("com.mysql.cj.jdbc.Driver");  
    conn = DriverManager.getConnection(  
        "jdbc:mysql://localhost:3306/student_management",  
        "root",  
        "Tienanh0108!"  
    );  
  
    String sql = "SELECT * FROM students WHERE id = ?";  
    pstmt = conn.prepareStatement(sql);  
    pstmt.setInt(1, studentId);  
    rs = pstmt.executeQuery();  
}
```

try { ... }: The code is enclosed in a try block, which is essential for handling exceptions like `ClassNotFoundException` (if the JDBC driver isn't found) or `SQLException` (if the connection or query fails).

`Class.forName("com.mysql.cj.jdbc.Driver");`: This line loads the MySQL JDBC driver into memory. This driver is required for the Java application to communicate with the MySQL database server.

`conn = DriverManager.getConnection(...)`: This establishes the physical connection to the database.

The JDBC URL specifies the database type (mysql), location (localhost:3306), and the database name (student_management).

The next two arguments are the username ("root") and password ("Tienanh0108!") used to authenticate the connection.

Result:

localhost:8080/StudentManagement/edit_student.jsp?id=6

Edit Student Information

Student Code

Cannot be changed

Full Name *

Email

Major

[Update](#) [Cancel](#)

Ex 3.2:

Testcase 1:

localhost:8080/StudentManagement/list_students.jsp?message=Student%20updated%20successfully

Student Management System

Student updated successfully

[Add New Student](#)

ID	Student Code	Full Name	Email	Major	Created At	Actions
6	SV006	John Doe	john@hcmiu.edu.vn	Data Science	2025-11-08 15:55:31.0	Edit Delete
5	SV005	David Wilson	david.w@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	2025-11-08 15:51:10.0	Edit Delete
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	2025-11-08 15:51:10.0	Edit Delete
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	2025-11-08 15:51:10.0	Edit Delete
1	SV001	John Doe	john.smith@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete

Testcase 2:

localhost:8080/StudentManagement/edit_student.jsp?id=6

Edit Student Information

Student Code
SV006
Cannot be changed

Full Name *

Email
john@hcmiu.edu.vn

Major
Data Science

Update Cancel

Testcase 3:

localhost:8080/StudentManagement/edit_student.jsp?id=6

Edit Student Information

Student Code
SV006
Cannot be changed

Full Name *

Email
john@hcmiu.edu.vn

Major
Data Science

Update Cancel

Please fill out this field.

Ex 4.1:

Ex 4.2:

Testcase:

The screenshot shows a web browser at `localhost:8080/StudentManagement/list_students.jsp`. A confirmation dialog box is displayed, asking "Are you sure?" with "OK" and "Cancel" buttons. The background shows the "Student Management System" interface with a table of students. The table has columns: ID, Student Code, Full Name, Email, Major, Created At, and Actions. The actions column contains "Edit" and "Delete" links. The table lists 6 students. Below the table is a button "Add New Student".

After clicking "OK", the browser address bar changes to `localhost:8080/StudentManagement/delete_student.jsp?id=6`. The page shows a green message "Student deleted successfully". The table now lists 5 students, with the student having ID 6 removed.

ID	Student Code	Full Name	Email	Major	Created At	Actions
6	SV006	John Doe	john@hcmiu.edu.vn	Data Science	2025-11-08 15:55:31.0	Edit Delete
5	SV005	David Wilson	david.w@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	2025-11-08 15:51:10.0	Edit Delete
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	2025-11-08 15:51:10.0	Edit Delete
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	2025-11-08 15:51:10.0	Edit Delete
1	SV001	John Doe	john.smith@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete

ID	Student Code	Full Name	Email	Major	Created At	Actions
5	SV005	David Wilson	david.w@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	2025-11-08 15:51:10.0	Edit Delete
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	2025-11-08 15:51:10.0	Edit Delete
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	2025-11-08 15:51:10.0	Edit Delete
1	SV001	John Doe	john.smith@email.com	Computer Science	2025-11-08 15:51:10.0	Edit Delete