[Bài tập] Quản lý thông tin Nhân viên

**1. Mục tiêu bài tập**

Xây dựng một ứng dụng web quản lý nhân viên với các chức năng CRUD đầy đủ, triển khai trên **Tomcat 11**, sử dụng **JDK 21** và kết nối cơ sở dữ liệu **MySQL 8** qua JDBC.

**2. Yêu cầu kỹ thuật**

* **Ngôn ngữ:** Java (JDK 21)
* **Web server:** Tomcat 11 (Jakarta Servlet API 6.x)
* **Cơ sở dữ liệu:** MySQL 8
* **Build tool:** Gradle (sinh file .war)
* **View:** JSP + JSTL (Jakarta JSTL 3.0)
* **Kết nối DB:** JDBC thuần
* **Mô hình:** MVC (Model – View – Controller)
* **Namespace Servlet:** Sử dụng jakarta.servlet.\* (không dùng javax.\*)

**3. Cơ sở dữ liệu**

Tạo cơ sở dữ liệu employee\_manager với bảng employees có các cột:

1. id (BIGINT, khóa chính, tự tăng)
2. full\_name (VARCHAR, bắt buộc)
3. email (VARCHAR, bắt buộc, không trùng lặp)
4. phone (VARCHAR)
5. department (VARCHAR)
6. salary (DECIMAL, mặc định 0, không âm)
7. hire\_date (DATE, bắt buộc)
8. created\_at (TIMESTAMP, mặc định CURRENT\_TIMESTAMP)

**4. Chức năng bắt buộc**

**4.1 Danh sách nhân viên (List)**

* Hiển thị toàn bộ nhân viên từ DB.
* Cho phép tìm kiếm theo tên hoặc email (q query param).
* Cho phép sắp xếp theo:
  + Tên (A → Z)
  + Lương (cao → thấp)
  + Ngày vào làm (mới → cũ)
* Có nút **Thêm mới** để vào trang tạo nhân viên.

**4.2 Thêm mới nhân viên (Create)**

* Form nhập các thông tin: Họ tên\*, Email\*, Phone, Phòng ban, Lương, Ngày vào làm.
* Validate:
  + Họ tên và Email bắt buộc.
  + Email không trùng với nhân viên khác.
* Sau khi lưu, quay lại trang danh sách.

**4.3 Chỉnh sửa nhân viên (Update)**

* Lấy dữ liệu nhân viên theo id và hiển thị trong form.
* Cho phép chỉnh sửa và lưu lại.
* Validate tương tự chức năng Thêm mới.

**4.4 Xóa nhân viên (Delete)**

* Xóa theo id (POST request).
* Yêu cầu xác nhận trước khi xóa.

**5. Giao diện (UI)**

* Giao diện đơn giản, dùng HTML + CSS cơ bản. (Ưu tiên sử dụng Bootstrap)
* Sử dụng JSP để render dữ liệu.
* Sử dụng JSTL để lặp danh sách, xử lý điều kiện.
* Các file JSP đặt trong WEB-INF/views để tránh truy cập trực tiếp.

**6. Yêu cầu triển khai**

* **Model**: Lớp Employee chứa các thuộc tính tương ứng bảng DB.
* **DAO**:
  + Interface EmployeeDAO định nghĩa các phương thức CRUD.
  + Lớp JdbcEmployeeDAO hiện thực kết nối MySQL bằng JDBC.
* **Controller (Servlet)**:
  + Một servlet cho mỗi chức năng (List, Create, Edit, Delete) hoặc gộp logic hợp lý.
* **View (JSP)**: Hiển thị dữ liệu và form.
* **Config DB**: Thông tin kết nối đặt trong db.properties, đọc bằng Java.
* **Build & Deploy**:
  + Build dự án bằng Gradle → sinh file .war.

Deploy file .war lên Tomcat 11.