Assignment: Xây dựng Ứng dụng Quản lý Nhân sự (HRM) Hoàn chỉnh bằng JavaScript Thuần  
Mô tả Tổng quát  
Trong assignment này, bạn sẽ xây dựng một ứng dụng quản lý nhân sự (Human Resource Management - HRM) hoàn chỉnh bằng JavaScript thuần (vanilla JavaScript), không sử dụng bất kỳ framework nào như React, Vue, Angular hoặc bất kỳ thư viện bên thứ ba nào. Ứng dụng phải chạy trên trình duyệt web và sử dụng localStorage để lưu trữ dữ liệu bền vững.  
Mục tiêu là thực hành các tính năng JavaScript nâng cao như:  
  
Cú pháp ES6+ (arrow functions, template literals, destructuring, spread/rest operators, v.v.).  
Class và module (sử dụng import/export để tổ chức mã nguồn thành các file riêng biệt).  
Async/await cho các tác vụ bất đồng bộ (ví dụ: giả lập delay khi lưu dữ liệu vào localStorage).  
Closure và higher-order functions để xử lý logic phức tạp.  
Thao tác DOM để xây dựng và cập nhật giao diện động.  
  
Ứng dụng phải bao gồm ít nhất 12 module riêng biệt, mỗi module xử lý một khía cạnh cụ thể của quản lý nhân sự. Bạn có thể thêm module bổ sung nếu cần để đảm bảo tính thống nhất và hoạt động mượt mà. Tổ chức mã nguồn thành các tệp .js riêng biệt cho từng module, và sử dụng một tệp app.js chính để import và tích hợp chúng.  
Về giao diện:  
  
Sử dụng HTML và CSS cơ bản để tạo cấu trúc và phong cách (giữ ở mức tối thiểu, tập trung vào logic JS).  
Tạo một dashboard chính với menu điều hướng (ví dụ: sidebar hoặc navbar) để chuyển đổi giữa các module.  
Sử dụng event listeners cho các tương tác người dùng (click, submit, input, v.v.).  
Đảm bảo ứng dụng responsive (sử dụng media queries cơ bản trong CSS) và xử lý lỗi mượt mà (kiểm tra dữ liệu đầu vào, hiển thị cảnh báo bằng alert hoặc modal, tránh crash ứng dụng).  
  
Dữ liệu sẽ được lưu trữ trong localStorage dưới dạng JSON (ví dụ: mảng đối tượng cho nhân viên, phòng ban, v.v.). Ứng dụng phải tự kiểm tra và khởi tạo dữ liệu mặc định nếu chưa tồn tại.  
Yêu cầu Kỹ thuật  
  
Ngôn ngữ và Công cụ: Chỉ sử dụng JavaScript thuần, HTML5, và CSS3. Không dùng thư viện bên ngoài.  
Cấu trúc Mã nguồn:  
  
index.html: File chính chứa cấu trúc HTML cơ bản (dashboard, menu), và các thẻ <script type="module"> để import app.js.  
style.css: CSS cơ bản cho giao diện (layout, form, table, button, v.v.). Liên kết qua <link> trong index.html.  
Các tệp .js riêng biệt cho từng module (ví dụ: authModule.js, employeeDbModule.js, v.v.).  
app.js: File chính để import tất cả module, khởi tạo ứng dụng, xử lý routing (chuyển module dựa trên menu), và gắn event listeners.  
  
  
Thực tiễn Tốt:  
  
Tuân thủ separation of concerns (tách biệt logic dữ liệu, giao diện, và xử lý sự kiện).  
Sử dụng comment ngắn gọn để giải thích mã nguồn.  
Mã phải modular, tái sử dụng (ví dụ: hàm chung cho validate input).  
Tự kiểm tra tính năng: Chạy ứng dụng trên trình duyệt, kiểm tra các trường hợp edge (dữ liệu rỗng, lỗi input, v.v.).  
  
  
Chạy Ứng dụng: Mở index.html trên trình duyệt (cần server local nếu dùng module, ví dụ: Live Server trên VS Code).  
  
Các Module Bắt buộc  
Dưới đây là danh sách 12 module bắt buộc. Mỗi module phải được triển khai trong một file .js riêng, export các hàm/class cần thiết, và import vào app.js. Tôi sẽ mô tả chi tiết chức năng của từng module để bạn dễ hiểu và triển khai. Tập trung vào logic JS; giao diện chỉ cần form/table đơn giản.  
  
Module Xác thực (AuthModule):  
  
Xử lý đăng nhập và đăng ký người dùng (admin hoặc HR manager).  
Lưu thông tin người dùng (username, password hashed đơn giản bằng closure) vào localStorage.  
Quản lý phiên làm việc: Kiểm tra token/session khi load app, logout để xóa session.  
Nếu chưa đăng nhập, hiển thị form login; sau đăng nhập, chuyển đến dashboard.  
Sử dụng async/await để giả lập delay kiểm tra credentials.  
  
  
Module Cơ sở dữ liệu Nhân viên (EmployeeDbModule):  
  
Quản lý lưu trữ dữ liệu nhân viên dưới dạng mảng đối tượng trong localStorage (mỗi nhân viên có thuộc tính: id, name, departmentId, positionId, salary, hireDate, v.v.).  
Các hàm: getAllEmployees(), getEmployeeById(id), saveEmployees(employees) – sử dụng JSON.parse/stringify.  
Khởi tạo dữ liệu mặc định nếu localStorage rỗng (ví dụ: 5 nhân viên mẫu).  
Sử dụng higher-order functions để filter/sort dữ liệu.  
  
  
Module Thêm Nhân viên (AddEmployeeModule):  
  
Tạo form DOM động (input cho name, id tự generate, department select, position select, salary, v.v.).  
Logic submit: Validate input (không rỗng, salary > 0), thêm vào db qua EmployeeDbModule.  
Hiển thị thông báo thành công/thất bại.  
Sử dụng event listeners cho form submit và input change.  
  
  
Module Sửa Nhân viên (EditEmployeeModule):  
  
Tạo form tìm kiếm (input id hoặc name) để load thông tin nhân viên hiện có.  
Hiển thị form edit với dữ liệu preload, cập nhật thông tin (gọi update từ EmployeeDbModule).  
Xử lý xác nhận trước khi save, validate input tương tự add.  
Sử dụng closure để lưu trạng thái edit.  
  
  
Module Xóa Nhân viên (DeleteEmployeeModule):  
  
Tìm kiếm nhân viên (tương tự edit), hiển thị confirm dialog (sử dụng window.confirm).  
Xóa khỏi db nếu xác nhận, cập nhật localStorage.  
Cập nhật dashboard hoặc list nếu cần (gọi refresh từ app.js).  
  
  
Module Tìm kiếm Nhân viên (SearchEmployeeModule):  
  
Tạo form tìm kiếm nâng cao: input cho name (regex), department, salary range (min-max).  
Logic: Sử dụng filter() với higher-order functions và RegExp để lọc dữ liệu từ db.  
Hiển thị kết quả dưới dạng table DOM, với sorting (ví dụ: theo salary).  
  
  
Module Quản lý Phòng ban (DepartmentModule):  
  
Quản lý danh sách phòng ban (mảng đối tượng: id, name, managerId).  
Các hàm: addDepartment(name), editDepartment(id, newName), deleteDepartment(id), getAllDepartments().  
Lưu vào localStorage riêng, liên kết với nhân viên qua departmentId.  
Hiển thị list table với button add/edit/delete.  
  
  
Module Quản lý Vị trí (PositionModule):  
  
Quản lý vị trí công việc (mảng: id, title, description, salaryBase).  
Các hàm: addPosition(title, desc), editPosition(id, updates), deletePosition(id), getAllPositions().  
Gán vị trí cho nhân viên khi add/edit (select dropdown từ list positions).  
Lưu vào localStorage, sử dụng async/await cho save.  
  
  
Module Quản lý Lương (SalaryModule):  
  
Tính toán lương: Cập nhật salary, thêm bonus/deduction (thuộc tính trong nhân viên).  
Tạo bảng lương: Generate table cho all employees với netSalary = salary + bonus - deduction.  
Các hàm: calculateNetSalary(employee), generatePayrollReport() – return array cho display.  
Sử dụng map/reduce cho tính toán.  
  
  
Module Theo dõi Chấm công (AttendanceModule):  
  
Ghi chấm công hàng ngày: Mảng attendance logs (date, employeeId, checkIn, checkOut).  
Các hàm: checkIn(employeeId), checkOut(employeeId), getAttendanceReport(employeeId, fromDate, toDate).  
Tính tổng giờ làm (sử dụng Date objects), lưu vào localStorage.  
Hiển thị calendar hoặc table cho report.  
  
  
Module Quản lý Nghỉ phép (LeaveModule):  
  
Xử lý yêu cầu nghỉ phép: Mảng leaves (employeeId, startDate, endDate, type: annual/sick, status: pending/approved).  
Các hàm: requestLeave(employeeId, dates, type), approveLeave(leaveId), getLeaveBalance(employeeId) – ví dụ: 20 ngày annual default.  
Theo dõi và cập nhật balance khi approve.  
Hiển thị list requests với button approve/reject.  
  
  
Module Đánh giá Hiệu suất (PerformanceModule):  
  
Thêm đánh giá: Mảng reviews (employeeId, date, rating: 1-5, feedback).  
Các hàm: addReview(employeeId, rating, feedback), getAverageRating(employeeId) – sử dụng reduce để tính trung bình.  
Hiển thị report: Table với average rating và list reviews.  
Sử dụng sort để hiển thị top performers.