**[Bài tập] 2. Xây dựng API quản lý nhân viên**

## 1. Mục tiêu

* Nắm được cách kết hợp giữa Typescript với NodeJS
* Biết cách cấu trúc thư mục hợp lý khi xây dựng API bằng NodeJS kết hợp với Typescript
* Nắm được cách viết stored procedure để xây dựng các API

## 2. Mô tả

* Thiết kế cơ sở dữ liệu bằng hệ quản trị cơ sở dữ liệu MySQL với các bảng sau:
  + employees: **employee\_id**(int) - Khóa chính, **employee\_code**(varchar), **employee\_name**(varchar), **gender** (tinyint), **birth\_date**(date), **email** (varchar), **password**(varchar), **department\_id**(int) - Khóa ngoại với bảng **departments**, **position\_id**(int) - Khóa ngoại với bảng **positions**, **created\_date** (date)
  + departments: **departments\_id** (int) - Khóa chính, **department\_name** (varchar), **created\_date** (date)
  + positions: **position\_id**(int) - Khóa chính, **position\_name** (varchar), **created\_date** (date)
* Cấu trúc thư mục hợp lý (có thể dựa vào cấu trúc thư mục ở phần bài đọc)
* Tiến hành viết các Stored procedure để phục vụ cho việc xây dựng các API sau:
  + API lấy thông tin tất cả nhân viên
  + API lấy thông tin một nhân viên theo id
  + API thêm mới nhân viên
    - Viết các middlewares để validate dữ liệu đầu vào
      * Các trường không được để trống: **employee\_code,**  **employee\_name, email , password, department\_id, position\_id**
      * Định dạng dữ liệu: email phải đúng định dạng, mã nhân viên phải bắt đầu bằng ký tự “**NV**” + **số**
  + API cập nhật thông tin một nhân viên theo id
    - Viết các middleware để validate dữ liệu đầu vào
  + API xóa thông tin một nhân viên theo id
  + API tìm kiếm và phân trang theo **tên hoặc email**
    - Nếu giá trị tìm kiếm không có thì mặc định sẽ hiển thị tất cả bản ghi
    - Tiến hành phân trang cho dữ liệu, mỗi trang hiển thị 20 bản ghi và khi chuyển qua trang thứ 2 thì hiển thị 20 bản ghi tiếp theo
    - Lấy ra số lượng bản ghi tìm thấy (Ví dụ: mặc định nếu không nhập giá trị tìm kiếm sẽ trả ra **totalCount** = 20, nếu khi nhập giá trị tìm kiếm mà có 5 bản ghi thì sẽ trả ra **totalCount**  5
  + API sinh mã mới. Bạn cần lấy ra mã nhân viên **lớn nhất** trong database, sau đó + 1 với mã đó

## 3. Đánh giá

Để hoàn thành bài tập, học viên cần:

* Đưa mã nguồn lên GitHub.
* Dán link của repository lên phần nộp bài trên hệ thống.