# **LAB 4 MÔN NODEJS**

## Chuẩn bị

* Tạo database **labnodejs** như lab3 (nếu có database rồi thì khỏi thực hiện bước này)
* Cài đặt module express: **npm install -g express**
* Cài đặt module express-generator : **npm -g install express-generator**

## Bài 1: Tạo project mvc với express và kết nối database

### Tạo project

* Tạo project: **express --ejs Lab4**
* Chuyển vào folder lab4, cài các module cơ bản:  **npm install**
* Cài thêm 2 module : mysql và nodemon
* Chạy project: **npm start**

### Tạo model kết nối đến database

Trong lab4, tạo folder **models** và file **models/database.js** Xong code như sau:

var mysql = require('mysql');

var db = mysql.createConnection({

   host: 'localhost',

   user: 'root',

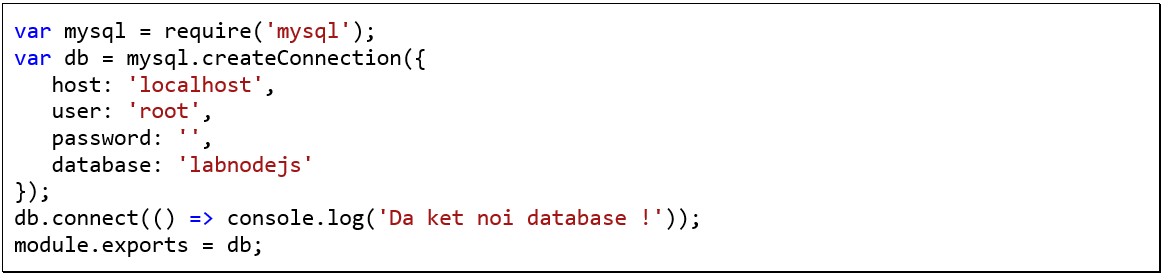
   password: '',

   database: 'labnodejs'

});

db.connect(() => console.log('Da ket noi database !'));

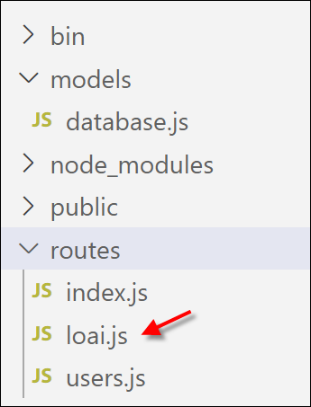
module.exports = db;



## Bài 2: Thực hiện controller loại

### Tạo controller loai

1. Trong folder **routes**, tạo file **loai.js** và code nhúng model + định nghĩa các route

var express = require('express');

var router = express.Router();

var db = require('../models/database');

router.**get**('/', function(req, res) {

    res.send('Danh sách loại');

});

router.**get**('/addnew', function(req, res) {

    res.send('Form thêm loại');

});

router.**post**('/store', function(req, res) {

    //nhận dữ liệu từ addnew để thêm record vào db

});

router.**get**('/edit/:id', function(req, res) {

  var id = req.params.id;

  res.send('Form chỉnh loại' + id);

});

router.**post**('/update', function(req, res) {

    //nhận dữ liệu từ edit để cập nhật vào db

});

router.**get**('/delete/:id', function(req, res) {

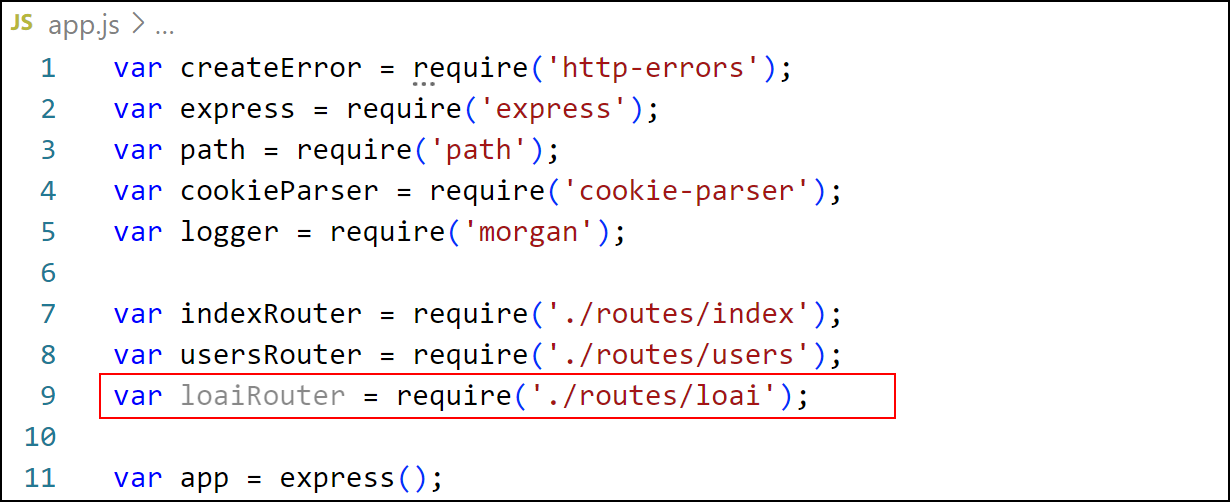
  var id = req.params.id;

  res.send('Xóa loai');

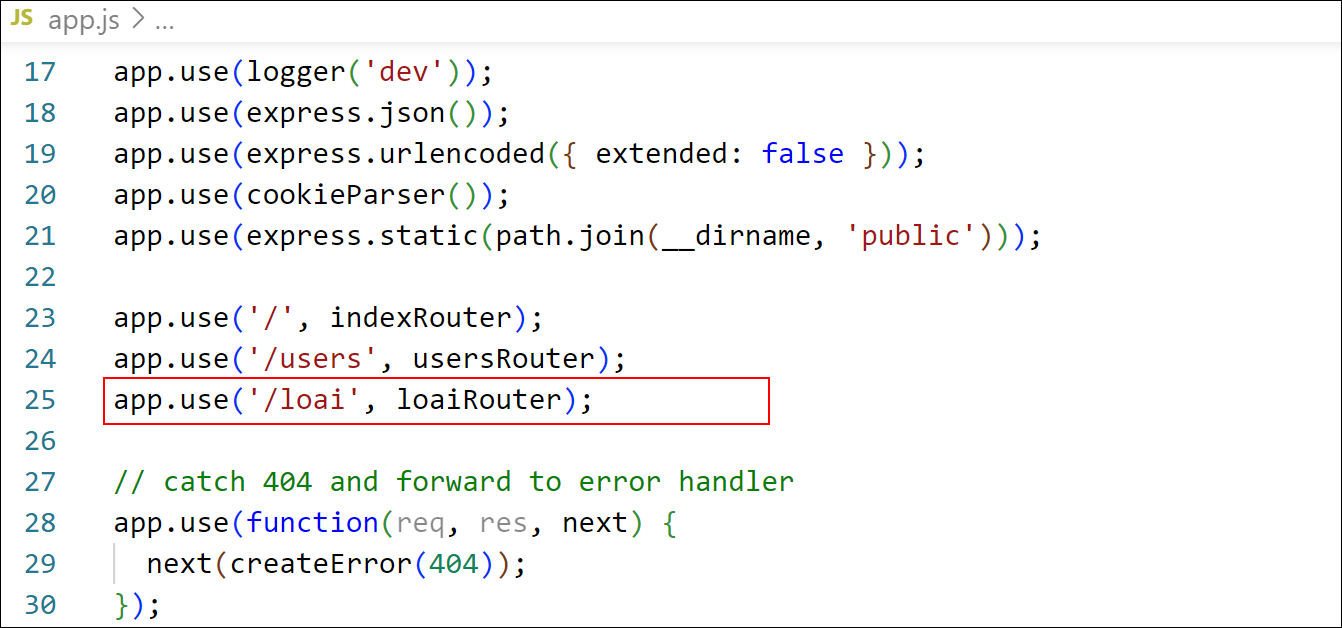
});

module.exports = router;

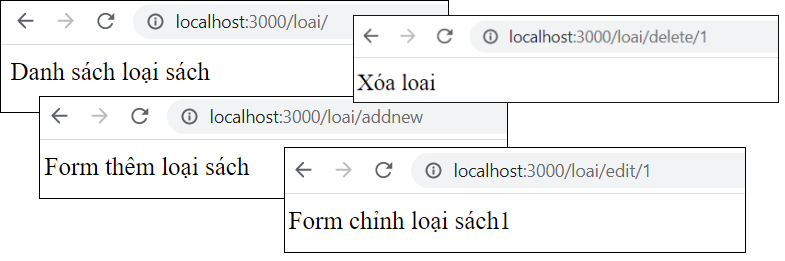
1. Nhúng controller vào ứng dụng: Mở file **app.js** và thêm lệnh như sau:



1. Khai báo đường dẫn cho controller: Trong file app.js , khai báo đường dẫn cho controller loại



1. Test các route, phải hiện như sau



### Chức năng hiện danh sách các record



* + Code lại trong route **/**của controller **loai** để được như sau

router.get('/', function(req, res, next) {

  //res.send('Danh sách loại');

  let sql = `SELECT \* FROM loai`;

  db.query(sql, function(err, data) {

      res.render("loai\_list",{list:data});

  });

});

* + Tạo file **views/loai\_list.ejs** và code

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel= "stylesheet">

<table class="table table-bordered w-75 m-auto">

    <tr> <th>id</th> <th>Tên loại</th> <th>Thứ tự</th> <th>Ẩn hiện</th> </tr>

    <% for (let c of list ) {  %>

    <tr> <td> <%= c.id %> </td>

        <td> <%= c.tenLoai %> </td>

        <td> <%= c.thuTu %> </td>

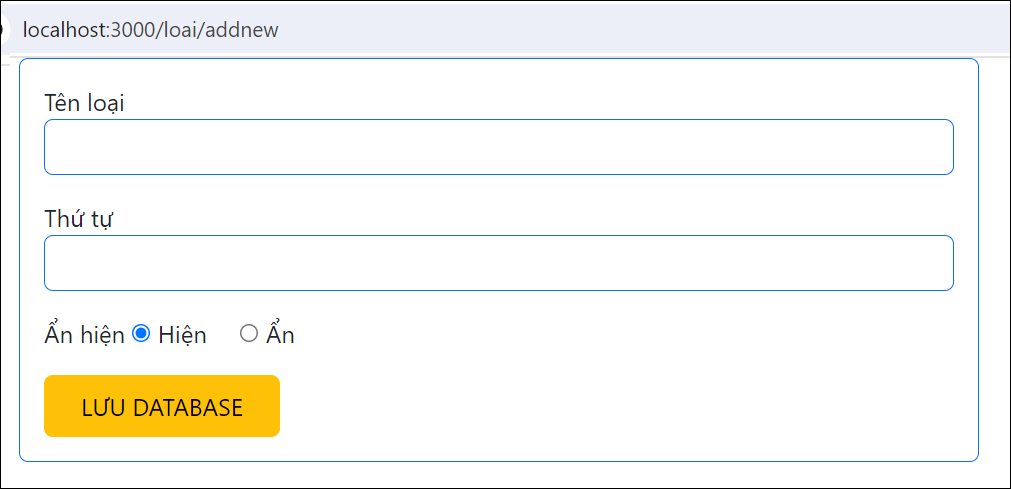
        <td> <%= c.anHien==1 ? "Đang hiện" : "Đang ẩn"%> </td>

    </tr>

    <% } %>

</table>

### Chức năng thêm mới



* + Code lại trong route **addnew**của controller **loai** để được như sau

router.get('/addnew', function(req, res, next) {

  //res.send('Form thêm loại');

  res.render("loai\_addnew");

});

* + Tạo file views/**loai\_addnew.ejs** và code

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">

<form class="col-6 border border-primary p-3 m-auto" action="/loai/store" method="POST">

    <div class="mb-3">

        Tên loại <input class="form-control border-primary" name="tenLoai" id="tenLoai">

    </div>

    <div class="mb-3">

        Thứ tự <input class="form-control border-primary" name="thuTu" id="thuTu">

    </div>

    <div class="mb-3"> Ẩn hiện

        <input type="radio" name="anHien" value="1" checked> Hiện &nbsp; &nbsp;

        <input type="radio" name="anHien" value="0"> Ẩn

    </div>

    <div class="mt-3">

        <button type="submit" class="btn btn-warning p-4 py-2">LƯU DATABASE</button>

    </div>

</form>

* + **Lưu record vào database**: Code lại route **store**của controller **loai** để được như sau

router.post('/store', function(req, res) {

  //nhận dữ liệu từ addnew để thêm record vào db

  let tl = req.body.tenLoai;

  let t = req.body.thuTu;

  let ah = req.body.anHien;

  let loai = { tenLoai:tl, thuTu:t, anHien:ah}

  db.query('insert into loai SET ?', loai, function(err, data) {

      if (err) throw err;

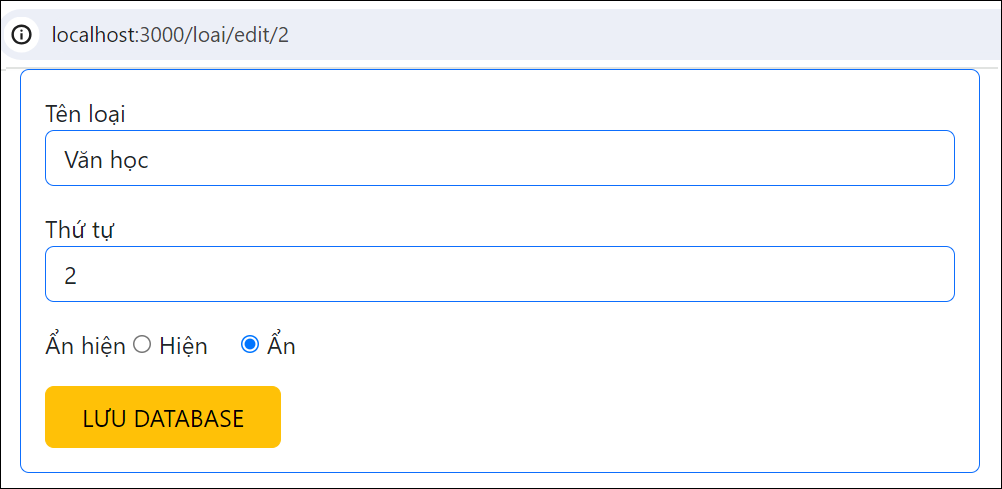
      res.redirect("/loai/");

  });

});

* + Test : submit form addnew , dữ liệu mới phải có trong database

### Chức năng hiệu chỉnh



* + Code lại trong route **edit**của controller **loai** để được như sau

router.get('/edit/:id', function(req, res) {

  let id = req.params.id;

  let sql = `SELECT id, tenLoai, thuTu, anHien FROM loai where id=${id}`;

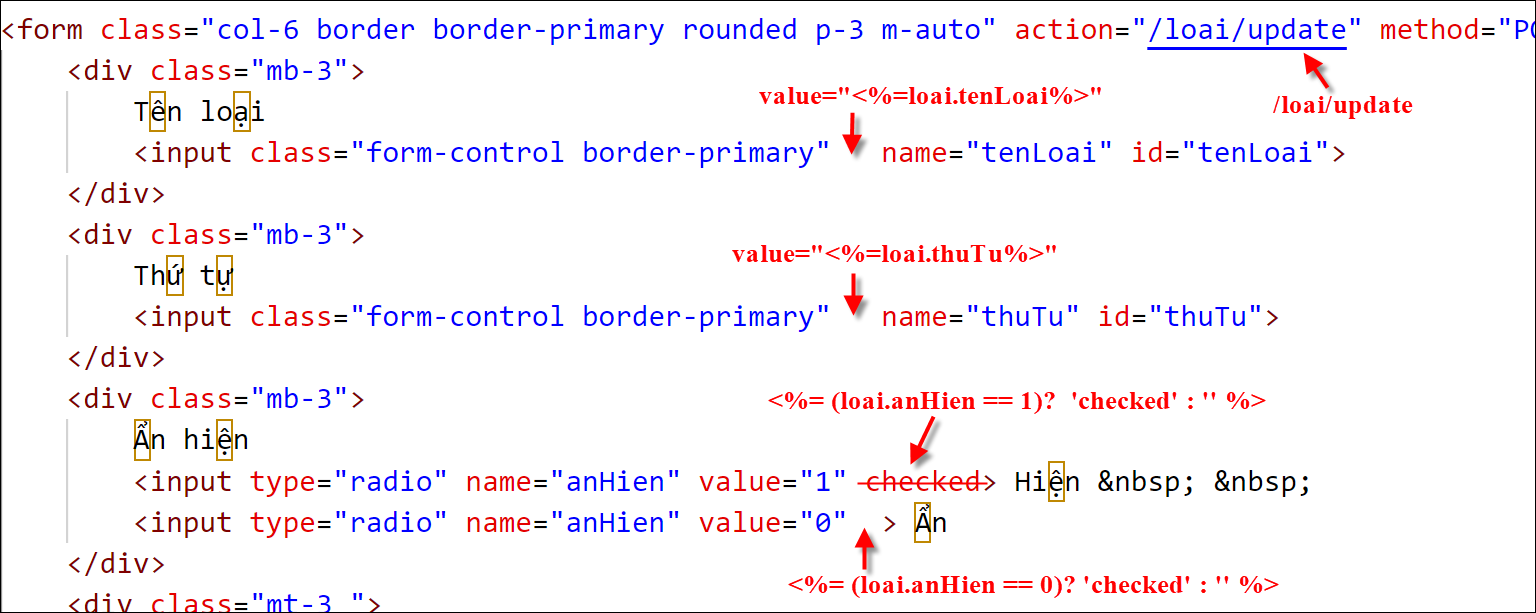
  db.query(sql, function(err, arr) {

    res.render("loai\_edit", { loai:arr[0]});

  });

});

* + Copy file **loai\_addnew** thành **loai\_edit.ejs** và thực hiện chỉnh như sau



* + Thêm code sau vào trong form (chỗ nào cũng được) để giữ id của loại cần cập nhật, sẽ dùng khi lưu database

<input type="hidden" name="id" value="<%=loai.id%>">

* + Test: nhập được dẫn edit với các id khác nhau sẽ hiện thông tin của loại trong form
  + Cập nhật record vào database: Code lại trong route **update**của controller **loai** để được như sau

router.post('/update/', function(req, res) {

  //nhận dữ liệu từ edit để cập nhật vào db

  let id = req.body.id

  let tenloai = req.body.tenLoai

  let thutu = req.body.thuTu

  let anhien = req.body.anHien

  let sql =  `UPDATE loai SET tenLoai=?,thuTu=?, anHien=? WHERE id = ?`

  db.query( sql,  [tenloai, thutu, anhien, id],  function(err, data) {

     if (data.affectedRows == 0) console.log(`Không có id loại ${id}`);

     res.redirect("/loai/");

 })

});

* + Test: nhập giá trị mới trong form edit và lưu, sẽ thấy tác dụng

### Chức năng xóa record

* + Code lại trong route **delete**của controller **loai** để được như sau

router.get('/delete/:id', function(req, res) {

  let id = req.params.id;

  let sql= `DELETE FROM loai WHERE id = ?`;

  db.query(sql, [id], function(err, data) {

    if (data.affectedRows==0) {

        console.log(`Không có loại ${id} để xóa`);

    }

    res.redirect('/loai');

  })

});

* + Test: nhập đường dẫn delete với các id thích hợp (ví dụ <http://localhost:3000/loai/delete/9>) sẽ thấy loại bị xóa

## Bài 3: Thực hiện controller sách

Thực hiện tương tự controller loại trong bài 2