# **LAB 5 MÔN NODEJS**

## Chuẩn bị

* Tạo database **labnodejs** như lab3 (nếu có database rồi thì khỏi thực hiện bước này)
* Vào <https://www.postman.com/downloads/> để download và cài postman (rất dễ)

## Bài 1: Tạo project và kết nối database

### Tạo project

* Tạo project: **express --ejs lab5**
* Chuyển vào folder lab5, 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 lab5, 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 Restful API với table sach

### Tạo controller sach

1. Trong folder **routes**, tạo file **sach.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, next) {

   // trả về mảng các record dạng json

});

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

   //tiếp nhận dữ liệu trong body request và lưu database

   //trả về thông báo đã chèn (dạng json)

});

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

   //tiếp nhận id của reord + lấy record từ database

   //trả về chi tiết record dạng json

});

router.put('/:id', function(req, res, next) {

   //tiếp nhận data trong body request và lưu vào database

   //trả về thông báo json đã cập nhật

});

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

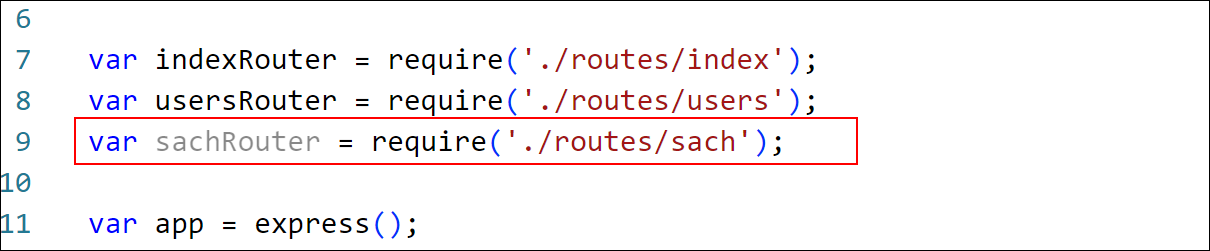
   //tiếp nhận id trong url và xóa record trong database

   //trả về thông báo json đã xóa

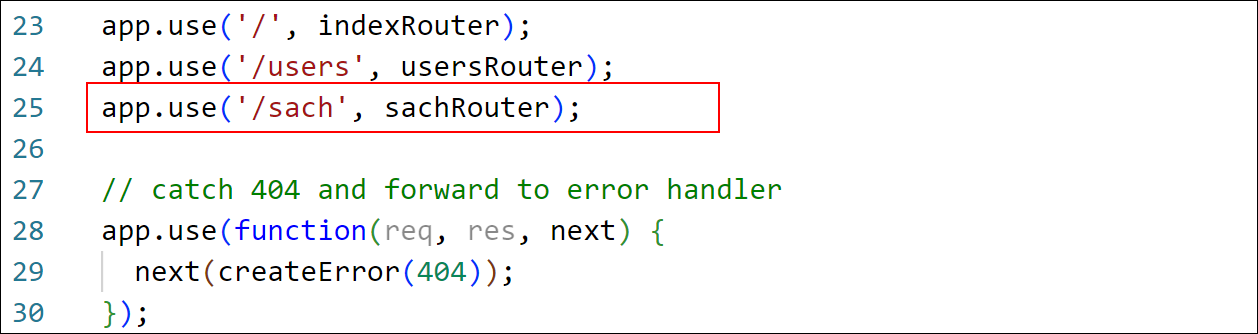
});

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 sách



### Chức năng hiện danh sách các record

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

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

    let sql = `SELECT id, tenSach, moTa, urlHinh FROM sach`;

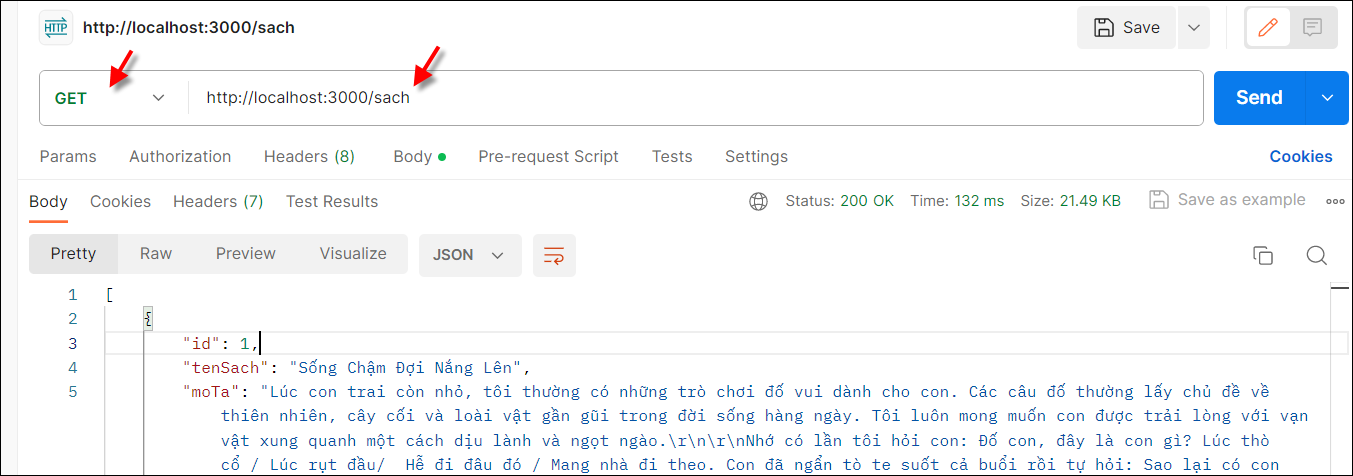
    db.query(sql, function(err, data) {

        res.json(data);

    });

});

* + Test : mở postman và request



### Hiện chi tiết 1 record

* + Trong sach.js, code lại route get /:id để được như sau

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

    let id=req.params.id;

    let sql = 'SELECT \* FROM sach WHERE id = ?'

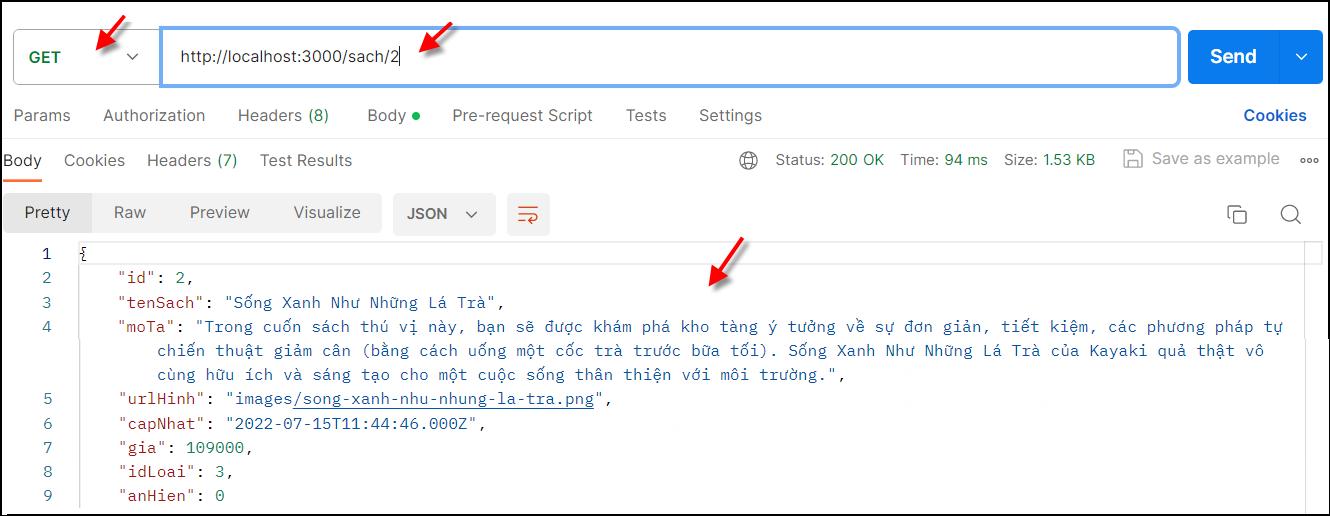
    db.query(sql, id, (err, d) => {

       res.json(d[0]);

    });

});

* + Test : mở postman và request



### Chức năng thêm record

* + Trong controller sach.js, code lại route post / để được như sau

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

    let data = req.body;

    let sql = 'INSERT INTO sach SET ?';

    db.query(sql, data, (err, d) => {

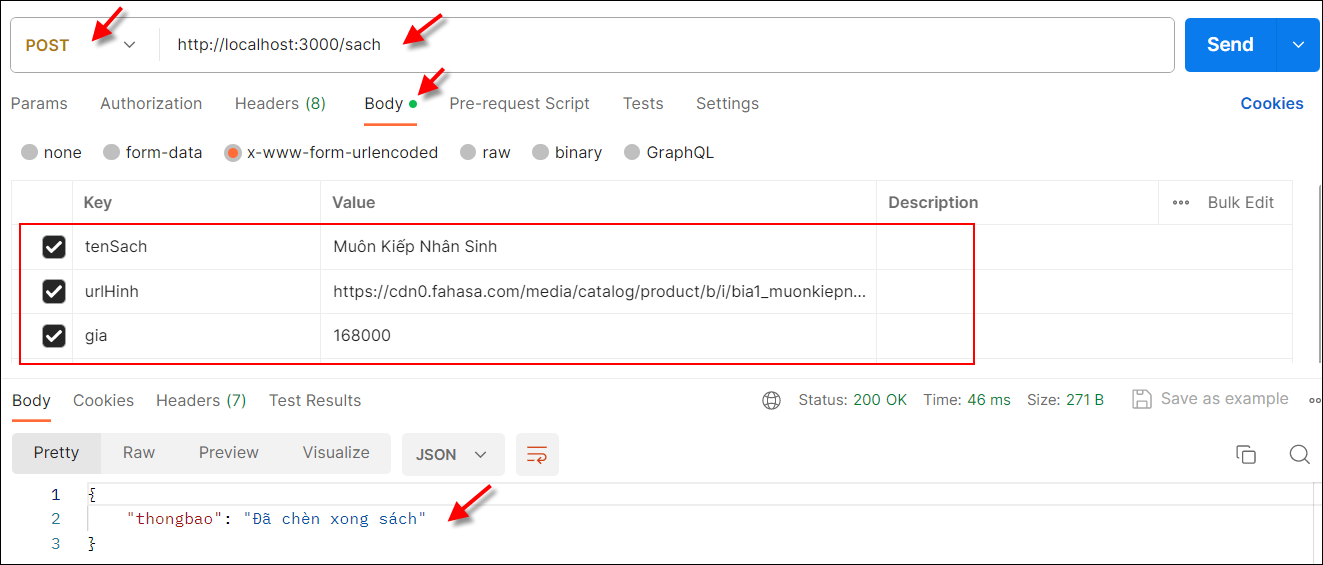
        if (err) throw err;

        res.json({"thongbao":"Đã chèn xong sách"});

    });

});

* + Test : mở postman và request



### Chức năng cập nhật record

* + Trong controller sach.js, code lại route **put  /:id** để được như sau

router.put('/:id', function(req, res, next) {

    let data = req.body;

    let id = req.params.id;

    let sql = 'UPDATE sach SET ? WHERE id = ?';

    db.query(sql, [data, id], (err, d) => {

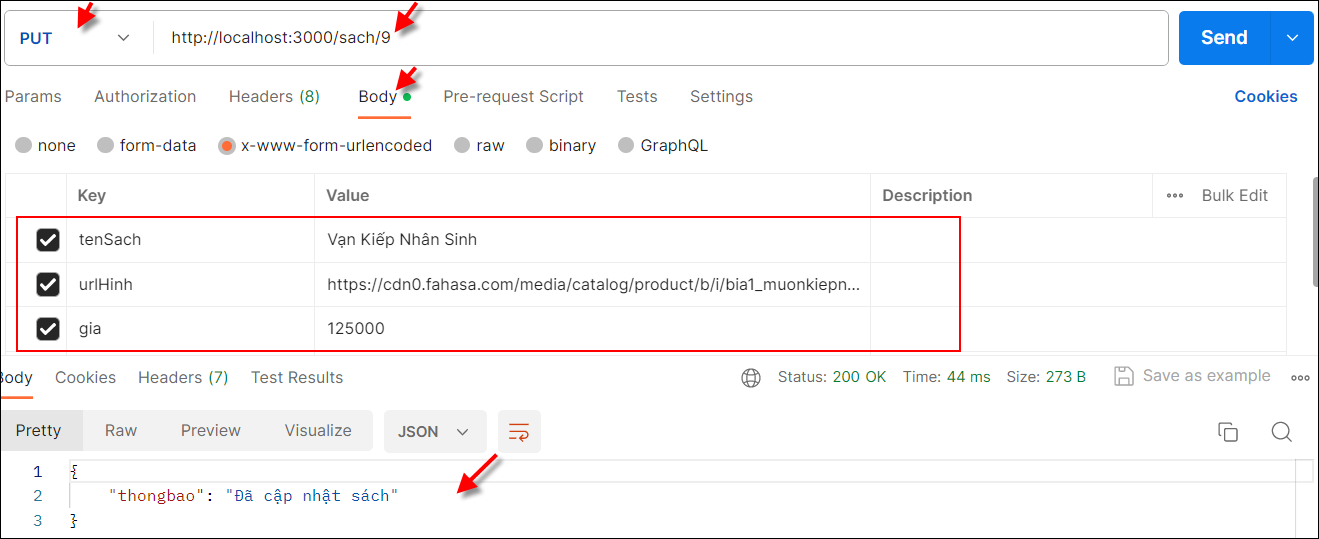
        if (err) throw err;

        res.json({"thongbao": 'Đã cập nhật sách'});

    });

});

* + Test : mở postman và request



### Chức năng xóa record

* + Trong controller sach.js, code lại route **put  /:id** để được như sau

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

    let id = req.params.id;

    let sql = 'DELETE FROM sach WHERE id = ?'

    db.query(sql, id , (err, d) => {

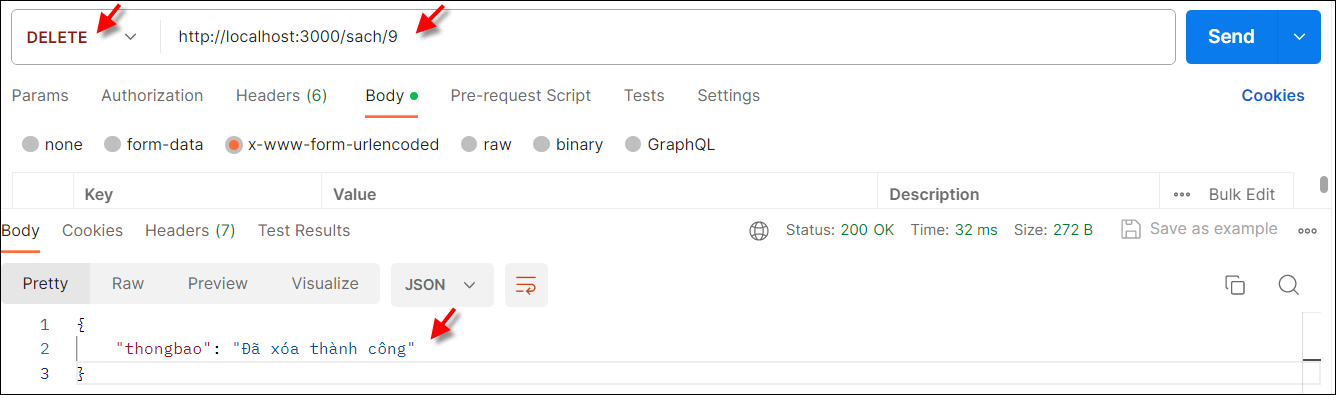
        if (err) throw err;

        res.json({"thongbao": 'Đã xóa thành công'});

    });

});

* + Test : mở postman và request



## Bài 3: Thực hiện Restful API với table loai

Thực hiện tương tự bài 2 nhưng với table loại