Router là một hành động giúp bạn quản lý toàn bộ các hành động trên Website cũng như quản lý các hoạt động của Request và Response có trên Website.

Tạo router: routes/test.js

Khai báo router trong api.js

var testRouter = require('./routes/test);

…

app.use('/exam', testRouter);

Link : localhost:3000/exam/…

Cấu trúc file router:

var express = require('express');

var router = express.Router();

router.get(‘/name/:param1/age/:param2’, function(req, res, next){

res.send(‘Hello World’);

var data = req.params; // Lấy tất cả param trên url gán vào obj data

var param = req.params.param1

});

router.post() | put() | delete();

module.exports = router;

* Ràng buộc dữ liệu trong Router bằng Regular

route.get(‘/node?js’, function(req, res){ }) // Có thể có hoặc ko có chữ e ở node

route.get(‘/data(-node)?’, function(req, res){ }) // Có thể có hoặc ko có chữ node

route.get(‘/node+js’, function(req, res){ }) // Chữ e xuất hiện ít nhất 1 lần

route.get(‘/data(-node)+’, function(req, res){ }) // -node xuất hiện ít nhận 1 lần

route.get(‘/data\*node’, function(req, res){ })// data-hoc-lap-trinh-node, đầu phải có chữ data, cuối phải có chữ node

route.get(/.\*data$/, function(req, res){ }) // nguyenhuudat-data, cuối cùng phải là chữ data

route.get(/.\*data.\*/ | /data/, function(req, res){ }) // ở giữa phải có chữ data

* Next và Callback

Cách 1:

router.get(‘/url’, function(req,res, next){ … next();}, function(req, res, next){ …});

Cách 2:

var a = function(req, res, next){ … next();}

var b = function(req, res, next){ … next();}

var c = function(req, res, next){ …}

router.get(‘/url’, [a,b,c]);

* Request: là yêu cầu của người dùng lên server thông qua thanh đụa chỉ

req.protocol // http

req.hostname // localhost

req.ip //

req.originalUrl // thanh địa chỉ

req.params // Lấy tất cả tham số trên thanh url

req.query // Lấy tham số kiểu query, index?user=joker&age=27

* Response

res.send(‘Node JS’)

res.headersSent // true – false :kiểm tra dữ liệu nhận về server chưa

res.end() // Ngừng ko làm gì nữa sau khi gửi nhận dữ liệu, tránh hacker tấn công

res.set(‘Content-Type’, ‘text/plain’)

res.get(‘Content-Type’)

res.json({user: ‘joker’, age: 27})

res.redirect(‘/url’) // chuyển trang

res.render(‘index’) // Hiển thị views/index.ejs

res.render(‘index’, function(error, content){

if(error){ … } else {

res.send(content)

}

res.end();

})

res.render(‘index’, {user: ‘joker’}) // Truyền tham số qua views

res.sendStatus(200 | 403 | 404 | 500) // Trạng thái

res.download(‘./file/image.png’ , ‘thay-doi-ten-file.jpg’) // download file