

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

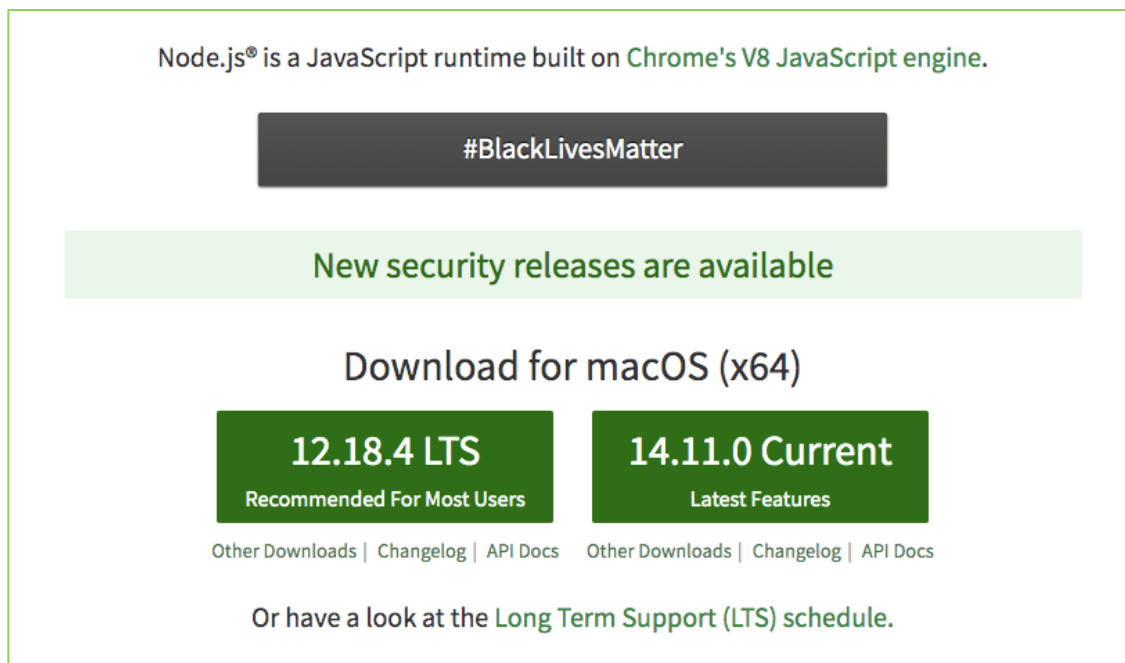
- ✓ Cài đặt và khởi tạo project với NodeJS
- ✓ Cài đặt và sử dụng được các module mở rộng
- ✓ Thực hiện Route
- ✓ Sử dụng tốt với các đối tượng request và response

PHẦN I

Bài 1 (2 điểm)

1. Cài đặt NodeJS

- Vào trang: <https://nodejs.org>, Và tải bản nodejs mới nhất, sau đó cài đặt vào máy



- Kiểm tra version node hiện hành: Mở terminal-> nhập lệnh node -v

```
Last login: Thu Sep 17 07:42:12 on console
congma:~ admin$ node -v
v10.16.3
```

2. Tạo project

- Chuyển tới thư mục chứa project -> nhập lệnh npm init

```
Demo admin$ cd lab01
lab01 admin$ npm init
```

```
Press ^C at any time to quit.
package name: (lab01)
version: (1.0.0)
description:
entry point: (index.js) server.js
test command: Test
git repository:
keywords: Lab01
author: Fpoly
license: (ISC)
About to write to /Users/admin/Desktop/FPT/nodejs/Demo/Lab01/package.json:

{
  "name": "lab01",
  "version": "1.0.0",
  "description": "",
  "main": "server.js",
  "scripts": {
    "test": "Test"
  },
  "keywords": [
    "Lab01"
  ],
  "author": "Fpoly",
  "license": "ISC"
}

Is this OK? (yes)
```

Cung cấp tham số: Nếu để giá trị mặc định thì tại các bước đó ta nhấn enter. Chú ý có thuộc tính 'entry point': mình thay đổi thành **server.js** (mặc định là index.js), nơi này khai báo tập tin được thực thi khi start server

- Cài thêm các package hỗ trợ khác: express, mysql, body-parser

```
npm install --save express
```

- Tạo file **server.js** trong project

```
▼ Lab01
  > node_modules
  {} package-lock.json
  {} package.json
  JS server.js
```

- Tạo server

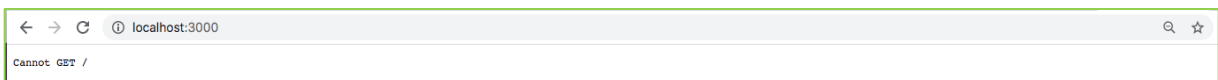
```
JS server.js  X
Lab01 > JS server.js > ...
1  const express=require('express');
2  const app=express();
3  const port=3000;
4
5  app.listen(port,()=>{
6    |   console.log(`ứng dụng đang chạy với port: ${port}`);
7    | })
```

...và chạy server

```
[congma:Lab01 admin$ npm start

> lab01@1.0.0 start /Users/admin/Desktop/FPT/nodejs/Demo/Lab01
> node server.js

ứng dụng đang chạy với port: 3000
```



Lúc này truy cập tới server thông qua http thì chưa có phản hồi. Sinh viên cho biết nguyên nhân

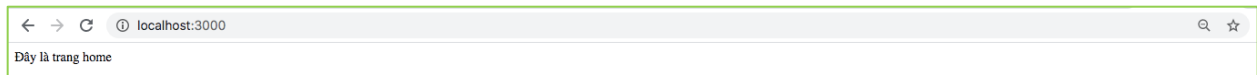
.....

.....

Bài 2 (2 điểm)

Thực hiện route, để server phản hồi lại các yêu cầu từ phía client

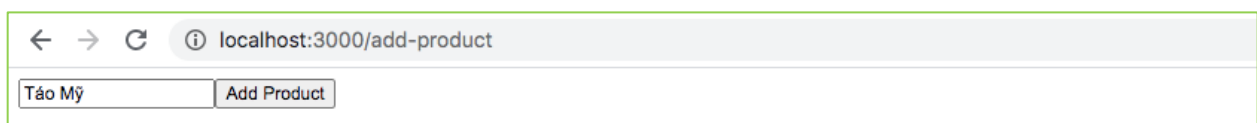
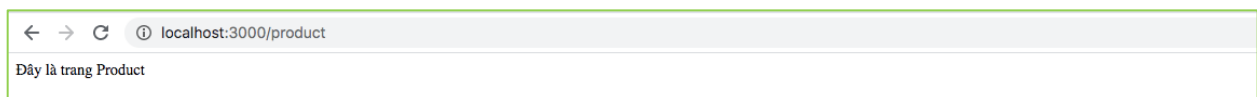
- Trang home, khi người dùng truy cập: <http://localhost:3000/>



Ta cần viết route cho server để phản hồi như sau

```
JS server.js ×
Lab01 > JS server.js > ...
1  const express=require('express');
2  const app=express();
3  const port=3000;
4
5  //router
6  app.get('/', (req, res) => {
7    |   res.send('<p>Đây là trang home</p>');
8    | });
9
10 app.listen(port,()=>{
11   |   console.log(`ứng dụng đang chạy với port: ${port}`);
12   | })
```

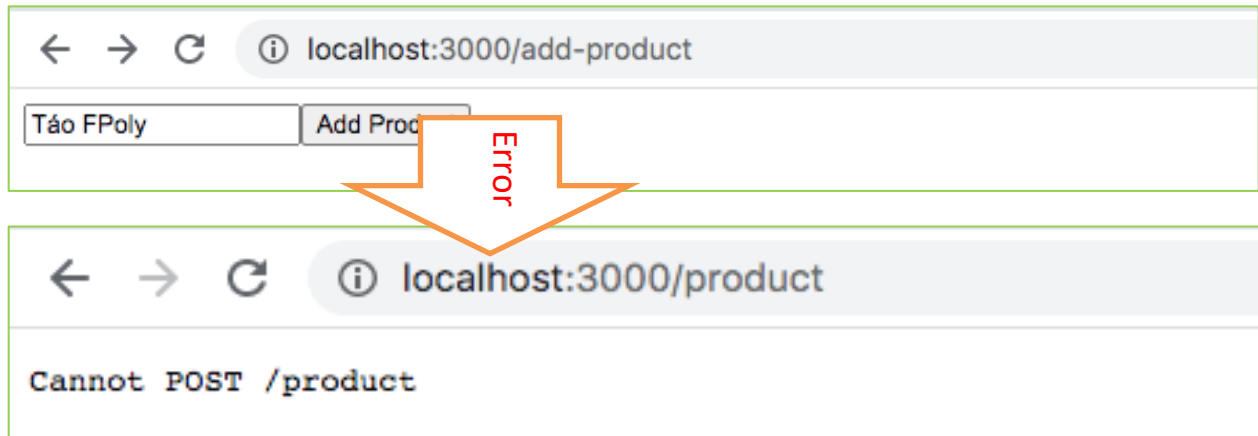
- Tương tự cho trang product, add product, khi người dùng truy cập: <http://localhost:3000/product> hoặc <http://localhost:3000/add-product>



Ta cần viết route cho server để phản hồi

```
5  //router
6  app.get('/', (req, res) => {
7    |   res.send('<p>Đây là trang home</p>');
8    | });
9  app.get('/product', (req, res, next) => {
10   |   res.send('<p>Đây là trang Product</p>');
11   | });
12 app.get('/add-product', (req, res, next) => {
13   |   res.send('<form action="/product" method="POST"><input type="text" name="productName"><button type="submit">Add Product</button></form>');
14   | });
```

- Lỗi khi thực hiện chuyển trang khi nhấn nút 'Add Product'



Sinh viên tìm hiểu nguyên nhân và thực hiện khắc phục

.....

.....

.....

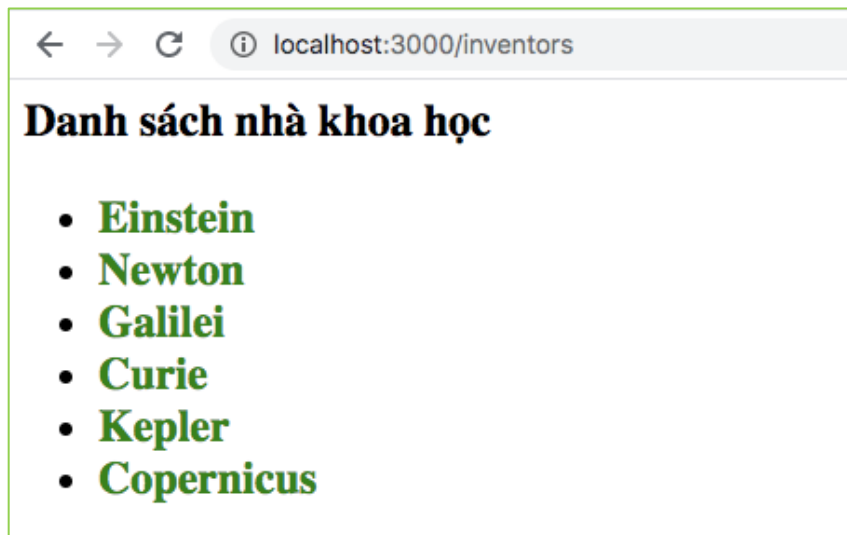
.....

PHẦN II

Bài 3 (2 điểm)

Xây dựng ứng dụng thông tin nhà khoa học như sau:

Liệt kê danh sách nhà khoa học



Chọn nhà khoa học nào thì cho thông tin chi tiết nhà khoa học đó



Hướng dẫn:

- Thêm data danh sách nhà khoa học

```
//data
const inventors = [
  { id: 1, first: 'Albert', last: 'Einstein', year: 1879, passed: 1955 },
  { id: 2, first: 'Isaac', last: 'Newton', year: 1643, passed: 1727 },
  { id: 3, first: 'Galileo', last: 'Galilei', year: 1564, passed: 1642 },
  { id: 4, first: 'Marie', last: 'Curie', year: 1867, passed: 1934 },
  { id: 5, first: 'Johannes', last: 'Kepler', year: 1571, passed: 1630 },
  { id: 6, first: 'Nicolaus', last: 'Copernicus', year: 1473, passed: 1543 }
];
```

- Route và xử lý dữ liệu trả về cho trang danh sách nhà khoa học

```
app.get('/inventors', (req, res) => {  
  let list=<h2>Danh sách nhà khoa học<ul>;  
  inventors.forEach(e => {  
    list+=`<li><a style="text-decoration:none;color:green;" href="/inventor/${e.id}">${e.last}</a></li>`;   
  });  
  list+=`</ul></h2>`;   
  res.send(list);  
});
```

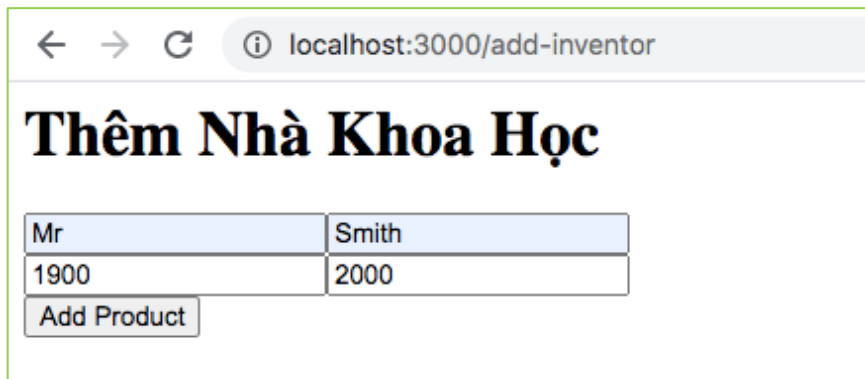
- Route và xử lý dữ liệu request (req.params) cho trang chi tiết nhà khoa học

```
app.get('/inventor/:id', (req, res) => {  
  let id=req.params.id;  
  inventor=inventors.find(e=>e.id==id);  
  info=`<h2>Thông tin chi tiết nhà khoa học:Full name: ${inventor.first} ${inventor.last}, Year: ${inventor.year},  
Passed: ${inventor.passed}</h2>`;   
  res.send(info);  
});
```

Yêu cầu thêm: Sinh viên xử lý thêm kết quả trả về (response) của trang chi tiết là dạng json, status

Bài 4 (2 điểm)

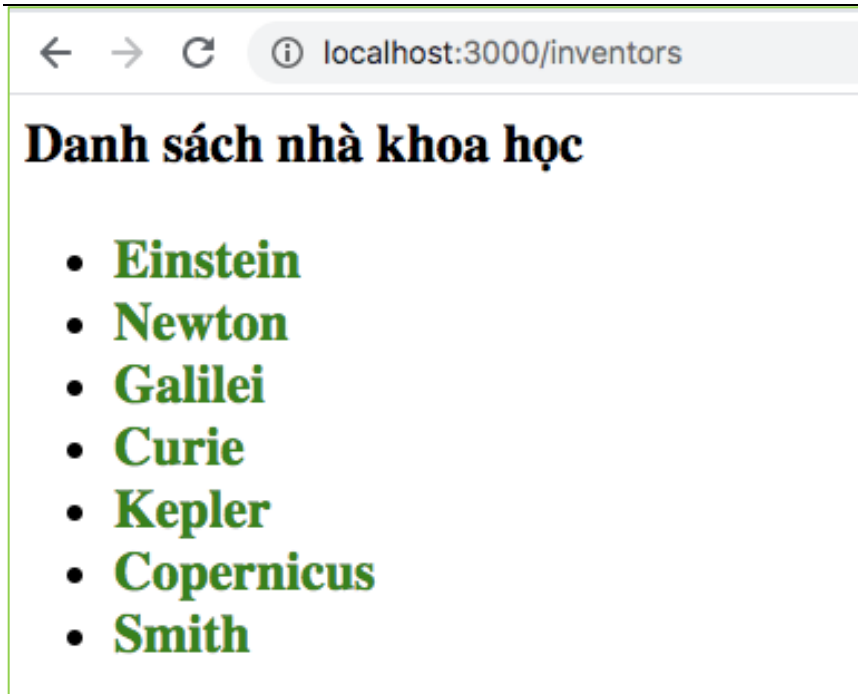
Xây dựng trang thêm nhà khoa học



Mr	Smith
1900	2000

Add Product

Sau khi thêm thành công ứng dụng trở về trang danh sách nhà khoa học



Hướng dẫn

- Thêm package để nhận dữ liệu từ form (dòng tô màu)

```
const express=require('express');  
var bodyParser = require('body-parser');  
const app=express();  
const port=3000;  
app.use(bodyParser.urlencoded());
```

- Viết route để load form thêm

```
app.get('/add-inventor', (req, res) => {  
  res.send('<h1>Thêm Nhà Khoa Học</h1><form action="/inventor" method="POST"><input type="text" name="first" placeholder="input first name"><input type="text" name="last" placeholder="input last name"><br><input type="number" name="year" placeholder="Year"><input type="number" name="passed" placeholder="passed"><br><button type="submit">Add Product</button></form>');  
});
```

- Cuối cùng thêm route cho sự kiện submit của nút 'Add product'. Các dòng tô màu là sử dụng đối tượng request và response, sinh viên cho biết ý nghĩa của từng dòng?


```
app.post('/inventor', (req, res) => {  
  let newInventor=req.body;  
  newInventor.id=inventors.length+1;  
  inventors.push(newInventor);  
  res.redirect('/inventors');  
});
```

Bài 5 (2 điểm)

Giảng viên cho thêm

*** Yêu cầu nộp bài:

SV nén file (hoặc share thư mục google drive) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.

--- Hết ---