

LAB01-THỰC HÀNH TẠO SERVER TRONG NODEJS

Mục tiêu:

- Tạo được server localhost để dùng trong nodejs
- Test các **request** từ Client gửi lên Server và **reponse** từ Server gửi về Client

Tạo project NodeJS

1. Tạo 1 folder với tên: **Lab01-họ tên sv**. Ví dụ: Lab01-HTSon
2. Mở command line và chuyển vào folder mới tạo
3. Chạy lệnh **npm init** để khai báo thông tin cho project Lab01-HTSon sẽ được yêu cầu nhập các thông tin như: name (lấy Lab01-HTSon luôn), version, description, ... hãy nhập vào hoặc nhấn Enter để dùng giá trị mặc định. Khi hoàn tất thì sẽ có 1 file tên **package.json** chứa thông tin các em vừa nhập, hãy mở file này lên xem thử nhé.
4. Cài đặt express : Express là một tập hợp các module quan trọng cho 1 project nodejs.

npm install express

5. Tạo file index.js và viết code

//tạo node server, đây coi như là con server localhost mà các em tạo ra

```
//tạo node server
const express = require("express");
const app = express();
const port = 3000;
```

```
//routes
app.get("/", (req, res) => {
    res.send("<h1>Đây là trang home</h1>");
});
```

```
//start server
app.listen(port, () =>{
    console.log(`Ung dung dang chay voi port ${port}`);
});
```

Chạy server: Trong command line, chạy lệnh

node index.js

Xem trang web: Mở trình duyệt và gõ <http://localhost:3000>

⇒ Xuất hiện: Đây là trang home

Đối tượng request trong nodejs

Là đối tượng chứa các thông tin trong request gửi đến từ client. Nó chứa các thuộc tính và hàm hữu dụng để các em dùng. Đối tượng request là tham số đầu tiên trong hàm callback của 1 route trong nodejs. Ví dụ trong route bên dưới thì biến req chính là đối tượng request

```
app.get("/", (req, res) => {  
    res.send("<h1>Đây là trang home</h1>");  
});
```

Các thuộc tính thường dùng trong đối tượng request

Đối tượng request trong nodejs chứa các thuộc tính sau mà các em sẽ thường hay dùng:

♣ **req.body**: là đối tượng chứa các cặp key-value của dữ liệu được gửi lên từ client.

Ví dụ: {'tensp': 'Iphone 15', 'gia': '3000000'}

=> Khi cần lấy các dữ liệu submit lên từ trình duyệt, bạn sẽ dùng đến req.body

♣ **req.params**: là đối tượng chứa các tham số của 1 route.

Ví dụ với route

```
app.get('/inventor/:id', (req, res) => {  
    console.log(req.params);  
})
```

=> Khi request **http://localhost:3000/inventor/1**, đối tượng params sẽ là { id: '1' }

♣ **req.query**: là đối tượng chứa query string của route.

Ví dụ với route sau

```
app.get('/inventors', (req, res) => { console.log(req.query);  
});
```

=> Khi request địa chỉ **http://localhost:3000/inventors/?limit=5&lang=vi**, giá trị của req.query sẽ là { limit: '5', lang: 'vi' }

Đối tượng response trong nodejs

Ngoài việc dùng request để nhận thông tin từ client thì đối tượng response trong nodejs bạn cũng phải dùng để trả ngược dữ liệu về phía client. Đối tượng response là tham số thứ hai trong hàm callback của 1 route. Ví dụ trong route bên dưới thì biến res chính là đối tượng response

```
app.get("/", (req , res ) => {  
    res.send("<h1>Đây là trang home</h1>");  
});
```

Các hàm thường dùng trong đối tượng response

res.send(codehtml): Hàm send dùng để trả về code html cho trình duyệt

res.json(data): Hàm json dùng để trả về dữ liệu json cho trình duyệt.

res.redirect(r): Hàm redirect trong response được dùng để yêu cầu trình duyệt chuyển hướng đến route 1 chỉ định

res.render(viewfile): Hàm render dùng để trả về cho trình duyệt file view chỉ định