

LAB 2

Step1 ເຕີຍ່ນ DB

```
CREATE DATABASE moviedb;
SHOW DATABASES;

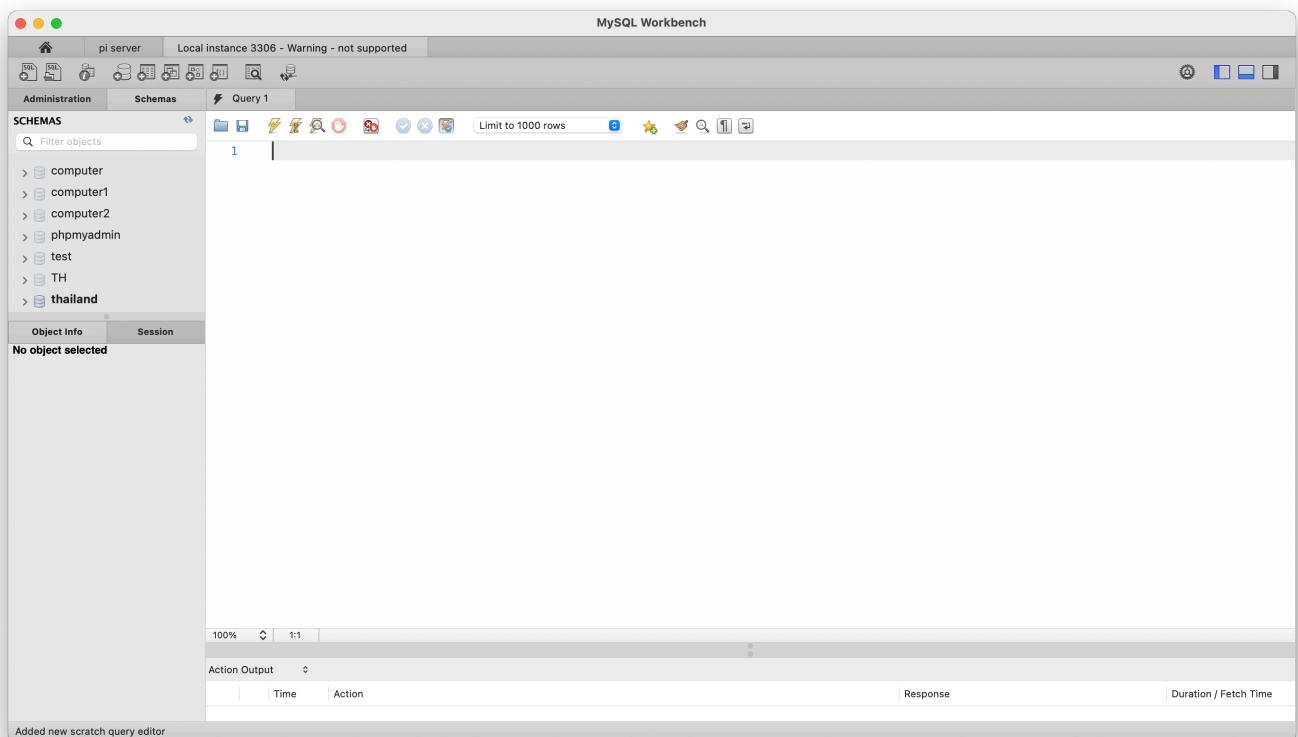
USE moviedb;
SHOW TABLES;

CREATE TABLE movies(title VARCHAR(50) NOT NULL,genre VARCHAR(30) NOT NULL,director
VARCHAR(60) NOT NULL,release_year INT NOT NULL,PRIMARY KEY(title));

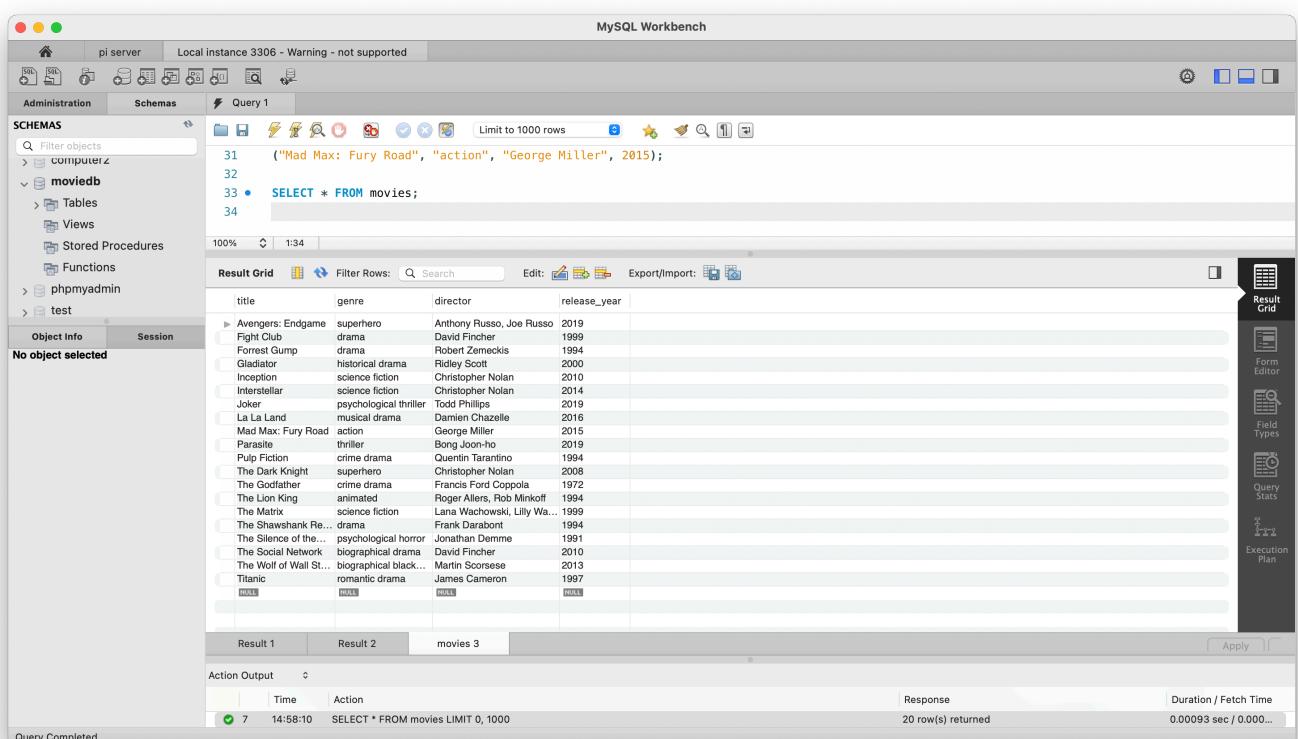
INSERT INTO movies (title, genre, director, release_year) VALUES
("Joker", "psychological thriller", "Todd Phillips", 2019),
("Inception", "science fiction", "Christopher Nolan", 2010),
("The Godfather", "crime drama", "Francis Ford Coppola", 1972),
("Pulp Fiction", "crime drama", "Quentin Tarantino", 1994),
("The Dark Knight", "superhero", "Christopher Nolan", 2008),
("Forrest Gump", "drama", "Robert Zemeckis", 1994),
("The Matrix", "science fiction", "Lana Wachowski, Lilly Wachowski", 1999),
("Fight Club", "drama", "David Fincher", 1999),
("The Shawshank Redemption", "drama", "Frank Darabont", 1994),
("The Social Network", "biographical drama", "David Fincher", 2010),
("Interstellar", "science fiction", "Christopher Nolan", 2014),
("Parasite", "thriller", "Bong Joon-ho", 2019),
("Gladiator", "historical drama", "Ridley Scott", 2000),
("Avengers: Endgame", "superhero", "Anthony Russo, Joe Russo", 2019),
("Titanic", "romantic drama", "James Cameron", 1997),
("The Silence of the Lambs", "psychological horror", "Jonathan Demme", 1991),
("La La Land", "musical drama", "Damien Chazelle", 2016),
("The Lion King", "animated", "Roger Allers, Rob Minkoff", 1994),
("The Wolf of Wall Street", "biographical black comedy", "Martin Scorsese", 2013),
("Mad Max: Fury Road", "action", "George Miller", 2015);

SELECT * FROM movies;
```

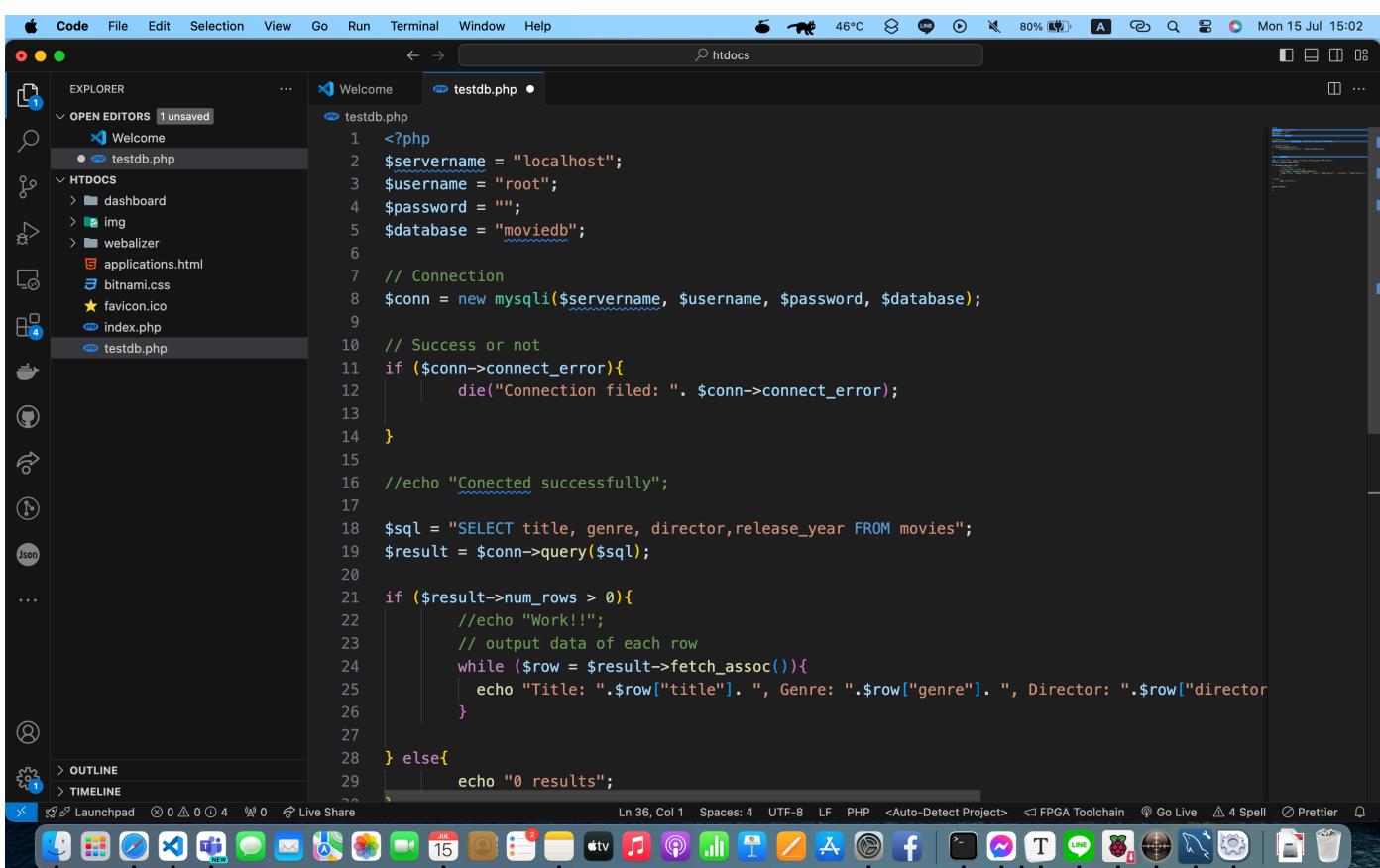
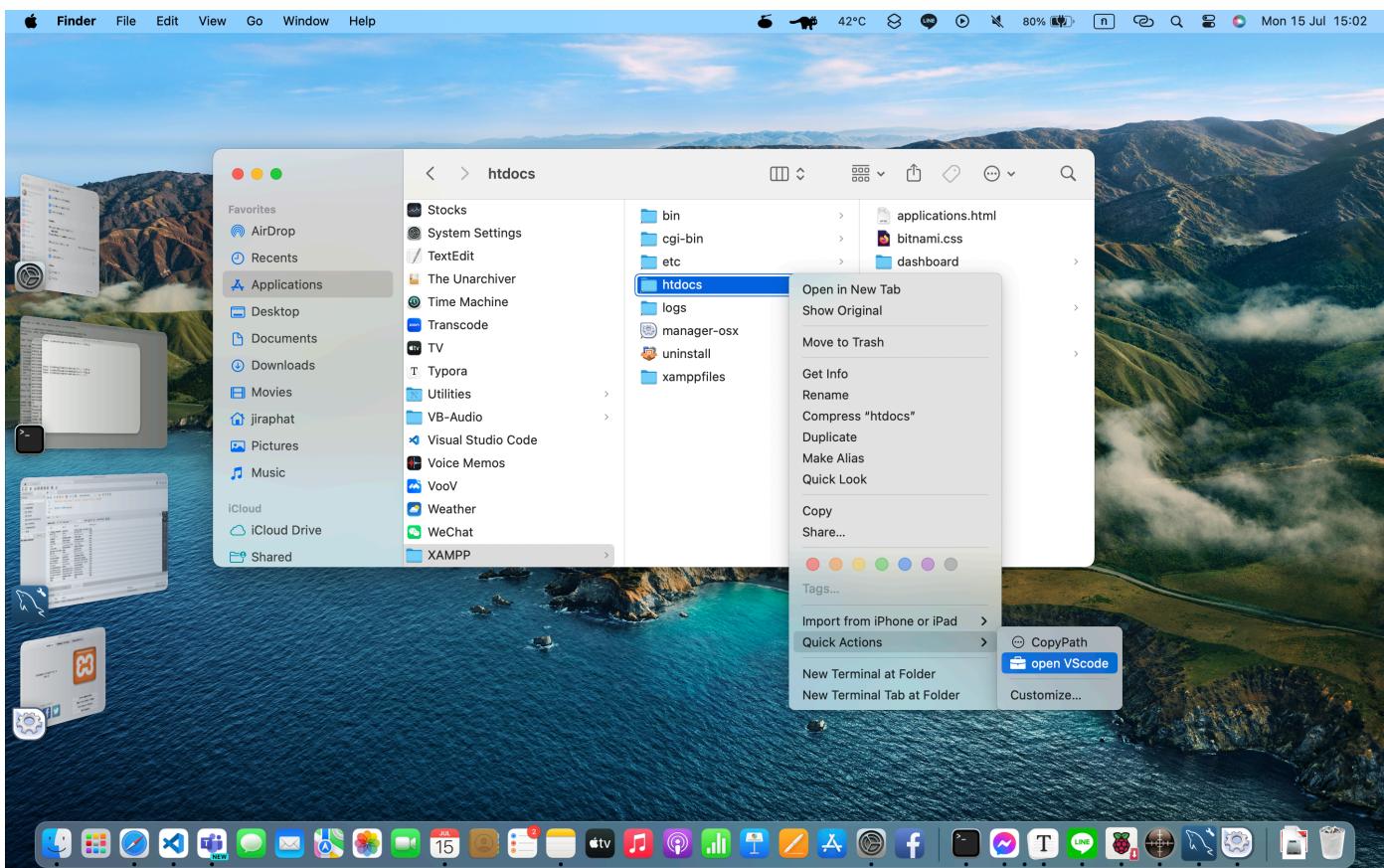
use MySQLWorkbench for connect to MySQL DB



create DB



สร้างไฟล์ testdb.php โดยใช้ vscode



ทดสอบเรียกใช้งาน



Step3 Restful API ด้วย Nodejs

```
cd /Applications/XAMPP/htdocs  
mkdir backend  
cd backend  
npm init
```

```
Terminal Shell Edit View Window Help 44°C 80% Mon 15 Jul 15:06
backend -- zsh - 158x43
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % pwd
/Applications/XAMPP/htdocs/backend
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (backend)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /Applications/XAMPP/xamppfiles/htdocs/backend/package.json:

{
  "name": "backend",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "description": ""
}

Is this OK? (yes)
(base) jiraphat@Jiraphats-MacBook-Air-2 backend %
```



```
Terminal Shell Edit View Window Help 44°C 80% Mon 15 Jul 15:06
backend -- zsh - 158x43
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install express --save
added 64 packages, and audited 65 packages in 5s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
(base) jiraphat@Jiraphats-MacBook-Air-2 backend %
```



สร้างไฟล์ index.js โดยใช้ vim และใส่ sourcecode เข้าไป

```
Terminal Shell Edit View Window Help
backend -- zsh -- 158x43
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install express --save
added 64 packages, and audited 65 packages in 5s
12 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % vim index.js
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % cat index.js
const express = require('express')
const app = express()

app.get('/', (req, res) => {
  res.send('Hello World')
})

app.listen(3001, () => {
  console.log('Start API Server at port 3001.')
})
(base) jiraphat@Jiraphats-MacBook-Air-2 backend %
```



```
npm install -g nodemon
```

```
Terminal Shell Edit View Window Help
backend -- zsh -- 158x43
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install express --save
added 64 packages, and audited 65 packages in 5s
12 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % vim index.js
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % cat index.js
const express = require('express')
const app = express()

app.get('/', (req, res) => {
  res.send('Hello World')
})

app.listen(3001, () => {
  console.log('Start API Server at port 3001.')
})
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install -g nodemon
added 29 packages in 4s
4 packages are looking for funding
  run `npm fund` for details
(base) jiraphat@Jiraphats-MacBook-Air-2 backend %
```



```
nodemon index.js
```

```
Terminal Shell Edit View Window Help
backend — node < node /usr/local/bin/nodemon index.js — 158x43
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install express --save
added 64 packages, and audited 65 packages in 5s
12 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % vim index.js
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % cat index.js
const express = require('express')
const app = express()

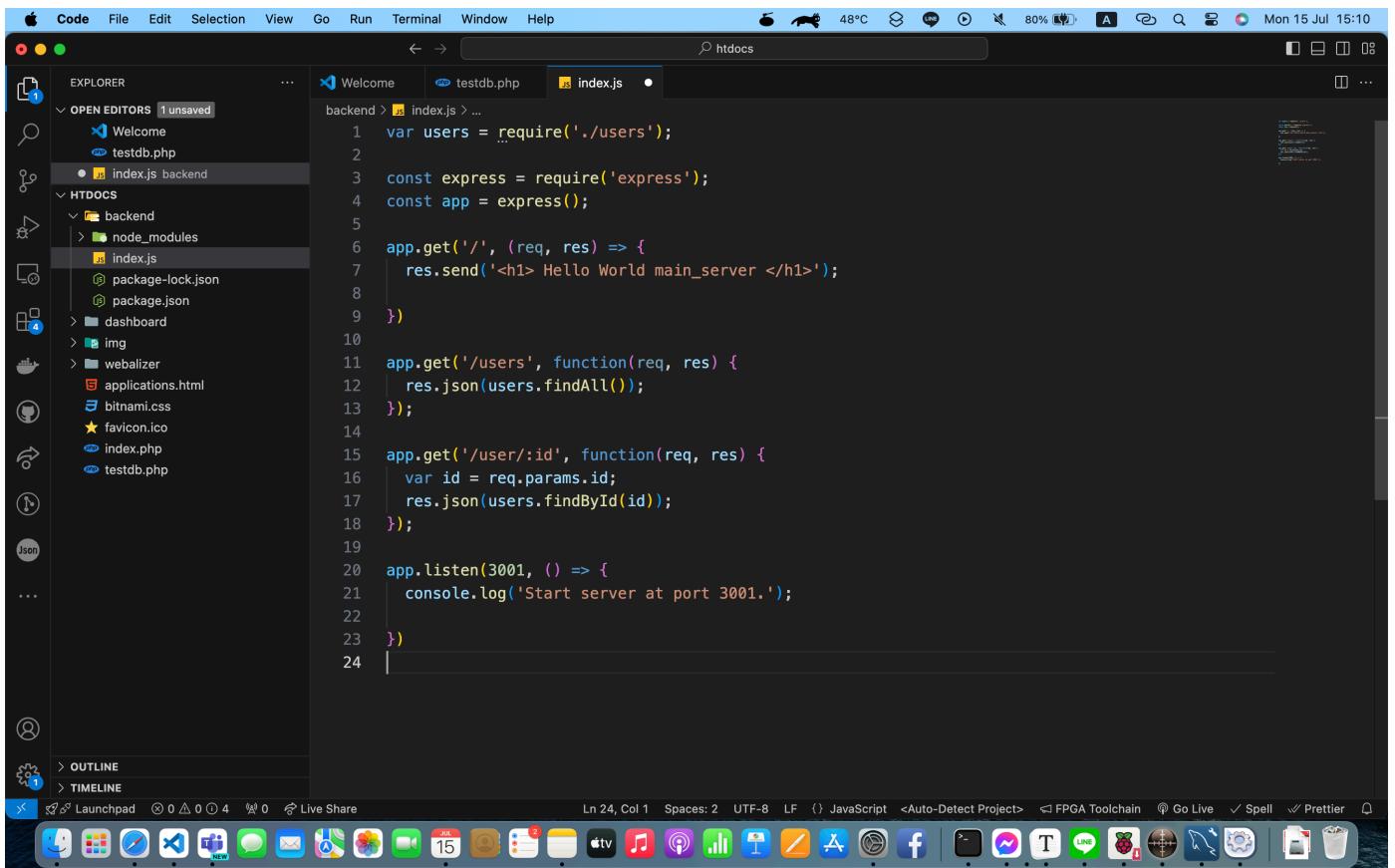
app.get('/', (req, res) => {
  res.send('Hello World')
})

app.listen(3001, () => {
  console.log('Start API Server at port 3001.')
})
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % npm install -g nodemon
added 29 packages in 4s
4 packages are looking for funding
  run `npm fund` for details
(base) jiraphat@Jiraphats-MacBook-Air-2 backend % nodemon index.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
Start API Server at port 3001.
```



A screenshot of a web browser window. The address bar shows "localhost:3001". The page content includes a "Favorites" section with tiles for Apple, iCloud, Yahoo!, Google, Wikipedia, Facebook, LinkedIn, The Weather Channel, Yelp, TripAdvisor, and Navy Cadence. Below this is a "Frequently Visited" section with tiles for YouTube, ChatGPT, Internet Speed Test, Canva, Instagram, and Google Translate.

ปรับปรุง index.js



```
var users = require('../users');
const express = require('express');
const app = express();

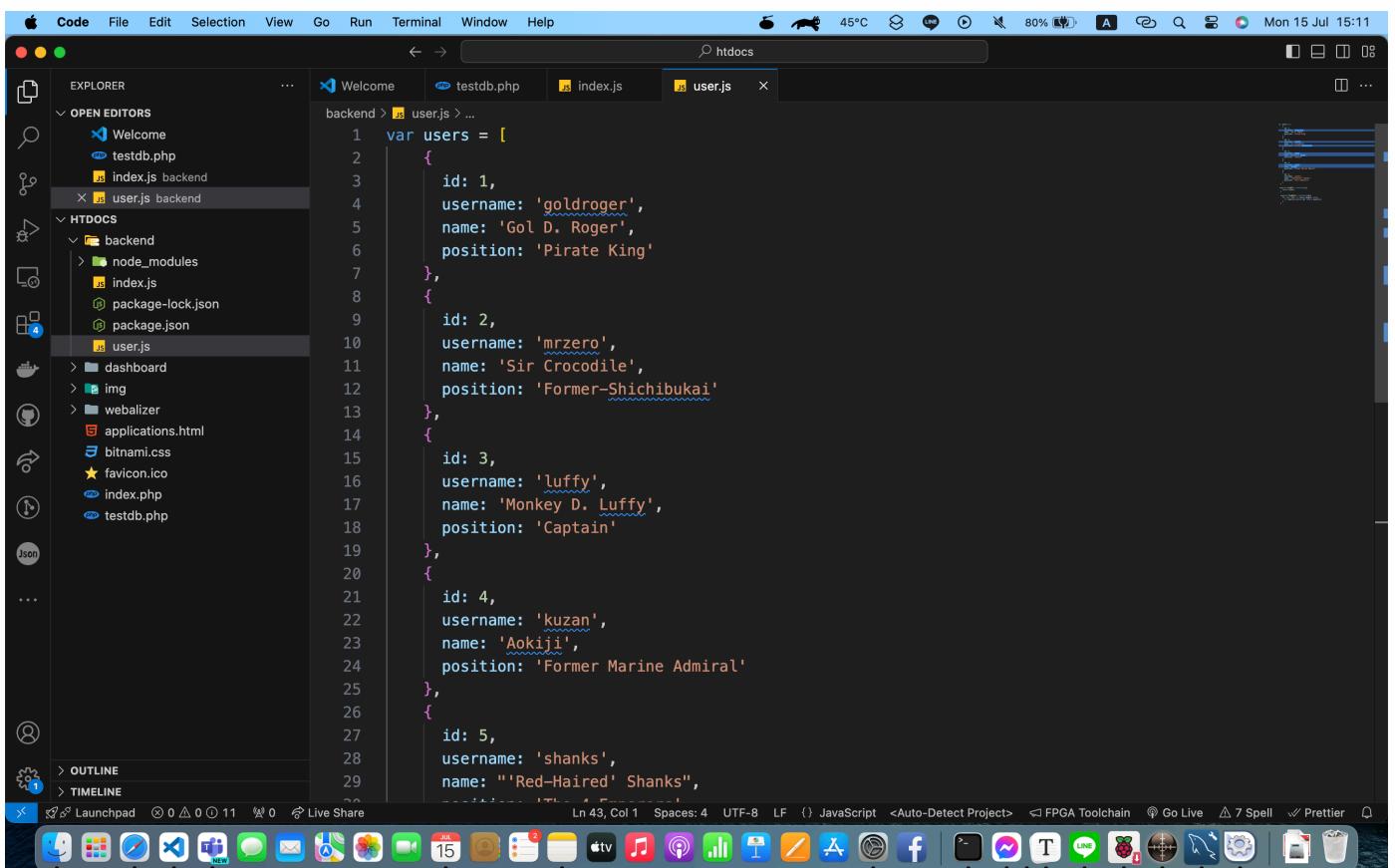
app.get('/', (req, res) => {
    res.send('<h1> Hello World main_server </h1>');
});

app.get('/users', function(req, res) {
    res.json(users.findAll());
});

app.get('/user/:id', function(req, res) {
    var id = req.params.id;
    res.json(users.findById(id));
});

app.listen(3001, () => {
    console.log('Start server at port 3001.');
});
```

สร้างไฟล์ users.js



```
var users = [
    {
        id: 1,
        username: 'goldroger',
        name: 'Gol D. Roger',
        position: 'Pirate King'
    },
    {
        id: 2,
        username: 'mrzero',
        name: 'Sir Crocodile',
        position: 'Former-Shichibukai'
    },
    {
        id: 3,
        username: 'luffy',
        name: 'Monkey D. Luffy',
        position: 'Captain'
    },
    {
        id: 4,
        username: 'kuzan',
        name: 'Aokiji',
        position: 'Former Marine Admiral'
    },
    {
        id: 5,
        username: 'shanks',
        name: "'Red-Haired' Shanks",
        position: 'Former Marine Admiral'
    }
];
```

ทดสอบเรียก API

<http://localhost:3001/users>



```
[{"id":1,"username":"goldroger","name":"Gol D. Roger","position":"Pirate King"}, {"id":2,"username":"mrzero","name":"Sir Crocodile","position":"Former-Shichibukai"}, {"id":3,"username":"luffy","name":"Monkey D. Luffy","position":"Captain"}, {"id":4,"username":"kuzan","name":"Aokiji","position":"Former Marine Admiral"}, {"id":5,"username":"shanks","name":"'Red-Haired' Shanks","position":"The 4 Emperors"}]
```

ແລະ <http://localhost:3001/user/5>



```
{"id":5,"username":"shanks","name":"'Red-Haired' Shanks","position":"The 4 Emperors"}
```