TUTORIAL COURSE

# Cài đặt

## Node

npm init

npx tsc –init

Generate a new file called config.json and tsconfig.json.

## ts-node

<https://www.npmjs.com/package/ts-node>

ts-node is a TypeScript execution engine and REPL for Node.js. Enabling you to directly execute TypeScript on Node.js without precompiling.

# Locally in your project.

npm install -D typescript

npm install -D ts-node

# Or globally with TypeScript.

npm install -g typescript

npm install -g ts-node

## Other

npm install -s helmet cors jsonwebtoken bcryptjs

npm i --save-dev nodemon ts-node-dev

npm i --save-dev @types/node @types/express

## Typeorm

npm install -g typeorm

npm install -s class-validator

# Cấu trúc thư mục

Graphical user interface, application

Description automatically generated

* Config: chứa file cấu hình kết nối database. Trong đó: db.connect.ts sử dụng để kết nối đến DB. File gd.data.source.ts định nghĩa PostgresDataSource.

Graphical user interface, text, application

Description automatically generated

* Controller: điều hướng đến model tương ứng và trả về kết quả cho router

Text

Description automatically generated

* Entities: chema định nghĩa bảng các đối tượng. Trong đó, base.entity.ts là file chứa thuộc tính tất cả các bảng đều phải có.

Text

Description automatically generated

* Interfaces: chứa file định nghĩa app expressjs

Graphical user interface, text

Description automatically generated

* Middleware: chứa file lưu các middleware sử dụng trong app này.

A screenshot of a computer

Description automatically generated with low confidence

* Model: file xử lý dữ liệu và trả lại controller theo thông tin request.

Graphical user interface, website

Description automatically generated

* Router: lưu router điều hướng đến controller

Graphical user interface, text

Description automatically generated

* App.ts: file định nghĩa các thành phần của ứng dụng và kết nối đến database.

import express, { Express } from 'express'

import Middleware from './middleware/app.middleware'

import Routes from './router/index'

import dotenv from 'dotenv'

dotenv.config()

const port = process.env.PORT || 3000

import { dbConnect } from './config/db.connect'

class App {

    public app: Express = express()

    constructor() {

        this.app = express()

        Middleware.init(this)

        Routes.init(this)

        dbConnect()

    }

}

const app = new App().app

app.listen(port, () => {

    console.log(`⚡️[server] is listenning on port ${port}`);

});

# Sử dụng const, let, var

## Const

* Sử dụng để gán các giá trị là hằng số: const PI: number = 3.14
* Sử dụng để gán giá trị không đổi: const indicatorValue: number = 7
* Sử dụng để gán giá trị obj, array, function: chỉ được phép thay đổi thuộc tính nằm trong nó. Không thay đổi đối tượng.

## Let

Biến có thể thay đổi trong hàm. Không thể sử dụng ngoài hàm. Khi kết thúc hàm thì biến đó sẽ được giải phóng khỏi bộ nhớ.

## Var

Biến có thể thay đổi, và giá trị có thể truyền ra ngoài hàm.

# Falsy value

Graphical user interface, table

Description automatically generated with medium confidence