

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng thư viện bên thứ 3 (3rd Party Libraries).
- ✓ Cài đặt và sử dụng express, nodejs kết hợp typescript.

NỘI DUNG

Bài 1: 3rd Party Libraries

- *product.model.ts*

```
import { IsNotEmpty, IsNumber, IsPositive } from 'class-validator';

export class Product {
  @IsNotEmpty()
  title: string;
  @IsNumber()
  @IsPositive()
  price: number;

  constructor(t: string, p: number) {
    this.title = t;
    this.price = p;
  }

  getInformation() {
    return [this.title, `${this.price}`];
  }
}
```

- *app.ts*

```
import 'reflect-metadata';
import { plainToClass } from 'class-transformer';
import { validate } from 'class-validator';

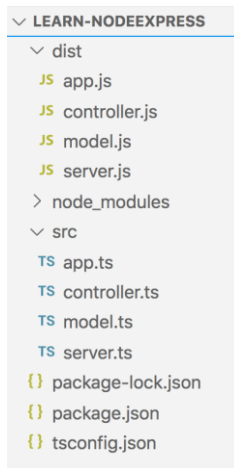
import { Product } from './product.model';

const products = [
  { title: 'A Carpet', price: 29.99 },
  { title: 'A Book', price: 10.99 }
];

const newProd = new Product('', -5.99);
validate(newProd).then(errors => {
  if (errors.length > 0) {
    console.log('VALIDATION ERRORS!');
    console.log(errors);
  } else {
    console.log(newProd.getInformation());
  }
});
```

Bài 2: Express, nodejs, typescript

- Tạo project có cấu trúc sau:



- Cài đặt express cho project.
- Code các tập tin
model.ts

```
export class User {
  public username: String;
  public name: String;
  public email: String;

  constructor(name: String, username: String, email: String) {
    this.name = name;
    this.username = username;
    this.email = email;
  }

  getUsername() {
    return this.username;
  }
  getName() {
    return this.name;
  }
}
```

controller.ts

```
import { Request, Response } from 'express';
import { User } from './model';

export let users = (req: Request, res: Response) => {
  let users = [
    new User('James Coonce', 'jcoonce', 'james@none.com'),
    new User('Jim Coonce', 'jimcoonce', 'jim@none.com'),
    new User('Norman', 'jcoonce', 'norman@none.com'),
  ];
  res.json(users);
};

export let create = (req: Request, res: Response) => {
  res.json(req.body);
};
```

app.ts

```
import express, { Request, Response, NextFunction } from 'express';
import bodyParser from 'body-parser';
import { json } from 'body-parser';
import * as controller from './controller';

const app = express();

app.use(json());
app.use(bodyParser.urlencoded({ extended: true }));

const router = express.Router();
app.use('/route', router);
router.get('/', (request, response) => { //route/index
  response.send('Hello world Route with Nodemon 1!');
});
router.get('/users', controller.users);
router.post('/users/create', controller.create);
```

Bài 3:

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 ---