

[CGA-JAVA-WFDA] Web Front-end Development Angular 2.1

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [8. Modules & Dependency Injection](#) / [\[Thực hành\] Xây dựng ứng dụng quản lý sản phẩm](#)

[Thực hành] Xây dựng ứng dụng quản lý sản phẩm

Mục tiêu

Sử dụng module trong Angular

Mô tả

Sử dụng ứng dụng quản lý sản phẩm đã được xây dựng ở bài Service và Router tại [đây](#) và tạo thêm các module để quản lý các component.

Yêu cầu

Ứng dụng quản lý sản phẩm phải được xây dựng như sau:

Tạo thêm một interface Category để quản lý danh mục các sản phẩm

Tạo các component được sử dụng để thêm mới, cập nhật và xoá một Category

Tạo CategoryService để thực hiện thao tác CRUD một Category

Xây dựng ProductModule và CategoryModule để quản lý các component

Hướng dẫn

Bước 1: Sử dụng lệnh ng g m product --routing để tạo ProductModule và ProductRoutingModule

Trong file product.module.ts khai báo các component ProductListComponent, ProductCreateComponent, ProductUpdateComponent và ProductDeleteComponent vào trong declaration và import `ReactiveFormsModule` đồng thời xoá ở bên app.module.ts

app.module.ts

```
import {BrowserModule} from '@angular/platform-browser';
import {NgModule} from '@angular/core';

import {AppRoutingModule} from './app-routing.module';
import {AppComponent} from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule {
}
```

product.module.ts

```
import {NgModule} from '@angular/core';
import {CommonModule} from '@angular/common';

import {ProductRoutingModule} from './product-routing.module';
import {ProductListComponent} from './product-list/product-list.component';
import {ProductCreateComponent} from './product-create/product-
create.component';
import {ProductEditComponent} from './product-edit/product-edit.component';
import {ProductDeleteComponent} from './product-delete/product-
delete.component';
import {ReactiveFormsModule} from '@angular/forms';

@NgModule({
  declarations: [
    ProductListComponent,
    ProductCreateComponent,
    ProductEditComponent,
    ProductDeleteComponent
  ],
  imports: [
    CommonModule,
    ProductRoutingModule,
    ReactiveFormsModule
  ]
})
export class ProductModule {
}
```

Bước 2: Cấu hình lại routes trong file app-routing.module.ts thành như sau:

```
import {NgModule} from '@angular/core';
import {RouterModule, Routes} from '@angular/router';

const routes: Routes = [
  {
    path: 'product',
    loadChildren: () => import('./product/product.module').then(module =>
module.ProductModule)
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
```

```
export class AppRoutingModuleModule {  
}
```

Trong file product-routing.module.ts, cấu hình lại thành như sau:

```
import {NgModule} from '@angular/core';  
import {RouterModule, Routes} from '@angular/router';  
import {ProductListComponent} from '../product-list/product-list.component';  
import {ProductCreateComponent} from '../product-create/product-  
create.component';  
import {ProductEditComponent} from '../product-edit/product-edit.component';  
import {ProductDeleteComponent} from '../product-delete/product-  
delete.component';  
  
const routes: Routes = [  
  {  
    path: 'list',  
    component: ProductListComponent  
  }, {  
    path: 'create',  
    component: ProductCreateComponent  
  }, {  
    path: 'edit/:id',  
    component: ProductEditComponent  
  }, {  
    path: 'delete/:id',  
    component: ProductDeleteComponent  
  }  
];  
  
@NgModule({  
  imports: [RouterModule.forChild(routes)],  
  exports: [RouterModule]  
})  
export class ProductRoutingModule {  
}
```

Chạy thử chương trình và kiểm tra lại kết quả.

Bước 3: Tạo một interface có tên là Category bằng câu lệnh ng g i model/category với 2 thuộc tính là id và name

```
export interface Category {  
  id?: number;  
  name?: string;  
}
```

Bước 4: Sử dụng lệnh ng g s service/category để tạo CategoryService trong thư mục service và khai báo danh sách các danh mục sản phẩm cùng với các phương thức CRUD cơ bản trong service này

```
import { Injectable } from '@angular/core';
import {Category} from '../model/category';

@Injectable({
  providedIn: 'root'
})
export class CategoryService {
  categories: Category[] = [{
    id: 1,
    name: 'iPhone'
  }, {
    id: 2,
    name: 'Samsung',
  }, {
    id: 3,
    name: 'LG',
  }
];

  constructor() {}

  getAll() {
    return this.categories;
  }

  saveCategory(category) {
    this.categories.push(category);
  }

  findById(id: number) {
    return this.categories.find(category => category.id === id);
  }

  updateCategory(id: number, category: Category) {
    for (let i = 0; i < this.categories.length; i++) {
      if (this.categories[i].id === id) {
        this.categories[i] = category;
      }
    }
  }

  deleteCategory(id: number) {
    this.categories = this.categories.filter(category => {
      return category.id !== id;
    });
  }
}
```

```
}
}
```

Bước 5: Sử dụng câu lệnh ng g m category --routing để tạo CategoryModule và CategoryRoutingModule

Bước 6: Thêm cấu hình routes cho file app.routing.module.ts

```
import {NgModule} from '@angular/core';
import {RouterModule, Routes} from '@angular/router';

const routes: Routes = [
  {
    path: 'product',
    loadChildren: () => import('./product/product.module').then(module =>
module.ProductModule)
  },
  {
    path: 'category',
    loadChildren: () => import('./category/category.module').then(module =>
module.CategoryModule)
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {
}
```

Bước 7: Gõ lệnh ng g c category/category-list để tạo CategoryListComponent

Trong file category-list.component.ts cấu hình như sau để lấy ra danh sách danh mục sản phẩm từ CategoryService:

```
import {Component, OnInit} from '@angular/core';
import {Category} from '../..../model/category';
import {CategoryService} from '../..../service/category.service';

@Component({
  selector: 'app-category-list',
  templateUrl: './category-list.component.html',
  styleUrls: ['./category-list.component.css']
})
export class CategoryListComponent implements OnInit {
  categories: Category[] = [];

  constructor(private categoryService: CategoryService) {
  }

  ngOnInit() {
    this.getAll();
  }

  getAll() {
    this.categories = this.categoryService.getAll();
  }
}
```

Chỉnh sửa lại file template của CategoryListComponent thành như sau:

category-list.component.html

```
<h1>Danh sách danh mục sản phẩm</h1>
<table>
  <tr>
    <td>STT</td>
    <td>Tên</td>
  </tr>
  <tr *ngFor="let category of categories; index as i">
    <td>{{i + 1}}</td>
    <td>{{category.name}}</td>
  </tr>
</table>
```

Thêm cấu hình routes trong file category-routing.module.ts:



```
import {NgModule} from '@angular/core';
import {RouterModule, Routes} from '@angular/router';
import {CategoryListComponent} from './category-list/category-
list.component';

const routes: Routes = [
  {
    path: 'list',
    component: CategoryListComponent
  },
];

@NgModule({
  imports: [RouterModule.forChild(routes)],
  exports: [RouterModule]
})
export class CategoryRoutingModule {
}
```

Trong file app.component.html thêm đoạn html như sau:

```
<a routerLink="/product/list">Danh sách sản phẩm</a>|<a
routerLink="/category/list">Danh sách danh mục sản phẩm</a>
<router-outlet></router-outlet>
```

Bước 8: Tạo file CategoryCreateComponent bằng câu lệnh ng g c category/category-create và cấu hình như sau:

Trong file category-create.component.ts khai báo một categoryForm:



```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {CategoryService} from '../..../service/category.service';

@Component({
  selector: 'app-category-create',
  templateUrl: './category-create.component.html',
  styleUrls: ['./category-create.component.css']
})
export class CategoryCreateComponent implements OnInit {
  categoryForm: FormGroup = new FormGroup({
    id: new FormControl(),
    name: new FormControl(),
  });

  constructor(private categoryService: CategoryService) {
  }

  ngOnInit() {
  }

  submit() {
    const category = this.categoryForm.value;
    this.categoryService.saveCategory(category);
    this.categoryForm.reset();
  }
}
```

category-create.component.html:

```
<h1>Tạo danh mục mới</h1>
<form [formGroup]="categoryForm" (ngSubmit)="submit()">
  <table>
    <tr>
      <td>Mã danh mục:</td>
      <td><input type="number" formControlName="id"></td>
    </tr>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td><button>Tạo mới</button></td>
    </tr>
  </table>
```


</form>

Khai báo đường dẫn đến category-create component và chỉnh sửa lại file category-list.component.html

category-routing.module.ts

```
const routes: Routes = [
  {
    path: 'list',
    component: CategoryListComponent
  },
  {
    path: 'create',
    component: CategoryCreateComponent
  },
];
```

category-list.component.html

```
<h1>Danh sách danh mục sản phẩm</h1>
<a routerLink="/category/create">Tạo mới danh mục</a>
<table>
  <tr>
    <td>STT</td>
    <td>Tên</td>
  </tr>
  <tr *ngFor="let category of categories; index as i">
    <td>{{i + 1}}</td>
    <td>{{category.name}}</td>
  </tr>
</table>
```

Bước 9: Tạo CategoryEditComponent bằng câu lệnh ng g c category/category-edit và cấu hình các file lần lượt như sau:

category-edit.component.ts

```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {ActivatedRoute, ParamMap} from '@angular/router';
import {CategoryService} from '../../service/category.service';

@Component({
  selector: 'app-category-edit',
  templateUrl: './category-edit.component.html',
  styleUrls: ['./category-edit.component.css']
})
export class CategoryEditComponent implements OnInit {
  categoryForm: FormGroup;
  id: number;

  constructor(private categoryService: CategoryService,
               private activatedRoute: ActivatedRoute) {
    this.activatedRoute.paramMap.subscribe((paramMap: ParamMap) => {
      this.id = +paramMap.get('id');
      const category = this.getCategory(this.id);
      this.categoryForm = new FormGroup({
        id: new FormControl(category.id),
        name: new FormControl(category.name),
      });
    });
  }

  ngOnInit() {
  }

  getCategory(id: number) {
    return this.categoryService.findById(id);
  }

  updateCategory(id: number) {
    const category = this.categoryForm.value;
    this.categoryService.updateCategory(id, category);
    alert('Cập nhật thành công');
  }
}
```

category-update.component.html

```
<h1>Cập nhật thông tin danh mục sản phẩm</h1>
<form [formGroup]="categoryForm" (ngSubmit)="updateCategory(id)">
  <table>
    <tr>
      <td>Mã danh mục:</td>
      <td><input type="number" formControlName="id"></td>
    </tr>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td><button>Cập nhật</button></td>
    </tr>
  </table>
</form>
```

category-routing.module.ts

```
const routes: Routes = [
  {
    path: 'list',
    component: CategoryListComponent
  },
  {
    path: 'create',
    component: CategoryCreateComponent
  },
  {
    path: 'edit/:id',
    component: CategoryEditComponent
  },
];
```

category-list.component.html:

```
<h1>Danh sách danh mục sản phẩm</h1>
<a routerLink="/category/create">Tạo mới danh mục</a>
<table>
  <tr>
    <td>STT</td>
    <td>Tên</td>
    <td></td>
  </tr>
  <tr *ngFor="let category of categories; index as i">
    <td>{{i + 1}}</td>
    <td>{{category.name}}</td>
    <td><a [routerLink]="['/category/edit',category.id]">Edit</a></td>
  </tr>
</table>
```

Bước 9: Tạo CategoryDeleteComponent bằng câu lệnh ng g c category/category-delete và cấu hình các file lần lượt như sau:

category-delete.component.ts

```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {CategoryService} from '../../service/category.service';
import {ActivatedRoute, ParamMap, Router} from '@angular/router';

@Component({
  selector: 'app-category-delete',
  templateUrl: './category-delete.component.html',
  styleUrls: ['./category-delete.component.css']
})
export class CategoryDeleteComponent implements OnInit {
  categoryForm: FormGroup;
  id: number;

  constructor(private categoryService: CategoryService,
               private router: Router,
               private activatedRoute: ActivatedRoute) {
    this.activatedRoute.paramMap.subscribe((paramMap: ParamMap) => {
      this.id = +paramMap.get('id');
      const category = this.getCategory(this.id);
      this.categoryForm = new FormGroup({
        id: new FormControl(category.id),
        name: new FormControl(category.name),
      });
    });
  }

  ngOnInit() {
  }

  getCategory(id: number) {
    return this.categoryService.findById(id);
  }

  deleteCategory(id: number) {
    this.categoryService.deleteCategory(id);
    this.router.navigate(['/category/list']);
  }
}
```

category-delete.component.html

```
<h1>Xoá danh mục sản phẩm</h1>
<form [formGroup]="categoryForm" (ngSubmit)="deleteCategory(id)">
  <table>
    <tr>
      <td>Mã danh mục:</td>
      <td><input type="number" formControlName="id"></td>
    </tr>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td><button>Xoá</button></td>
    </tr>
  </table>
</form>
```

category-routing.module.ts

```
const routes: Routes = [
  {
    path: 'list',
    component: CategoryListComponent
  },
  {
    path: 'create',
    component: CategoryCreateComponent
  },
  {
    path: 'edit/:id',
    component: CategoryEditComponent
  },
  {
    path: 'delete/:id',
    component: CategoryDeleteComponent
  },
];
```

category-list.component.html:

```
<h1>Danh sách danh mục sản phẩm</h1>
<a routerLink="/category/create">Tạo mới danh mục</a>
<table>
  <tr>
    <td>STT</td>
    <td>Tên</td>
    <td colspan="2"></td>
  </tr>
  <tr *ngFor="let category of categories; index as i">
    <td>{{i + 1}}</td>
    <td>{{category.name}}</td>
    <td><a [routerLink]="['/category/edit',category.id]">Edit</a></td>
    <td><a [routerLink]="['/category/delete',category.id]">Delete</a></td>
  </tr>
</table>
```

Checkout sang nhánh module để xem mã nguồn tham khảo: <https://github.com/codegym-vn/wfda-angular-product-management>

Hướng dẫn nộp bài:

up bài tập lên github

paste link github vào phần nộp bài

Submission status

Submission status	No attempt
Grading status	Not graded
Last modified	-
Submission comments	<div>▶ Comments (0)</div>

Add submission

You have not made a submission yet.

Previous Activity

Jump to...

Next Activity

CHƯƠNG TRÌNH

- Career
- Premium
- Accelerator

TÀI NGUYÊN

- Blog
- Tạp chí Lập trình
- AgileBreakfast

Follow Us



CodeGym@2018. All rights reserved.

You are logged in as Dương Văn Thanh Sơn (Log out)