

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

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [7. Service và Router](#) / [\[Thực hành\] Ứng dụng nghe nhạc](#)

[Thực hành] Ứng dụng nghe nhạc

Mục tiêu

Nắm bắt được về các thao tác với service và router trong Angular.

Mô tả

Xây dựng ứng dụng nghe nhạc


Hướng dẫn

Bước 1: Mở cửa sổ terminal và gõ lệnh ng new angular-product-management để tạo mới một dự án Angular

Command Prompt

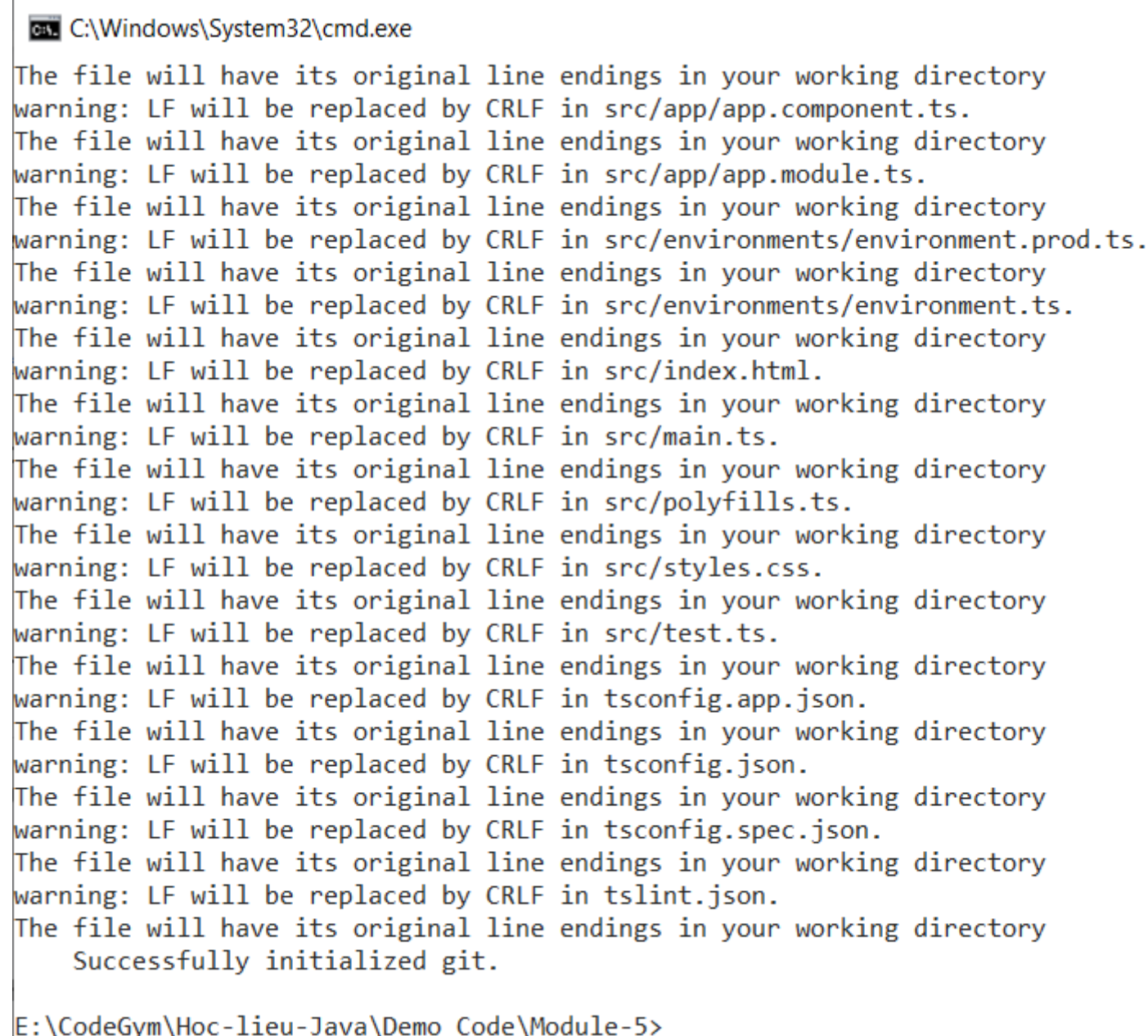
```
E:\CodeGym\Hoc-lieu-Java\Demo Code\Module-5>ng new angular-playlist-app
```

Tại cửa sổ tiếp theo gõ chữ y và ấn enter để tạo routing cho dự án

 npm

```
E:\CodeGym\Hoc-lieu-Java\Demo Code\Module-5>ng new angular-playlist-app
? Would you like to add Angular routing? (y/N) y
```

Tiếp theo chọn CSS và ấn enter và chờ ứng dụng được khởi tạo



```
C:\Windows\System32\cmd.exe

The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/app/app.component.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/app/app.module.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/environments/environment.prod.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/environments/environment.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/index.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/main.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/polyfills.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/styles.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/test.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.app.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.spec.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tslint.json.
The file will have its original line endings in your working directory
Successfully initialized git.

E:\CodeGym\Hoc-lieu-Java\Demo Code\Module-5>
```

Vậy là dự án đã được tạo xong

Bước 2: Mở dự án vừa tạo bằng Webstorm

Bước 3: Mở cửa sổ terminal trong Webstorm và gõ lệnh `ng g i model/song` để tạo một interface mới trong thư mục model

Trong interface Song thêm đoạn code như sau để khai báo các thuộc tính:

```
export interface Song{
  id?: string;
  name?: string;
}
```

Bước 4: Gõ lệnh `ng g s service/song` để tạo SongService

Bước 5: Trong file `song.service.ts`, khai báo một danh sách bài hát:

```
playlist: Song[] = [
  {
    id: 'CX11yw6YL1w',
    name: 'Céline Dion - Ashes (from the Deadpool 2 Motion Picture Soundtrack)'
  },
  {
    id: 'HBYirj2c_jw',
    name: 'Deadpool 2 - Take on Me'
  },
  {
    id: 'N602ncUKvlg',
    name: 'Nelly - Just A Dream'
  },
  {
    id: 'uelHwf8o7_U',
    name: 'Eminem - Love The Way You Lie ft. Rihanna'
  },
  {
    id: 'WpYeekQkAdc',
    name: 'The Black Eyed Peas - Where Is The Love?'
  }
];
```

Bước 6: Khai báo phương thức `findSongById()` để tìm kiếm bài hát theo id đã có:

```
findSongById(id: string) {
  return this.playlist.find(item => item.id === id);
}
```

Code đầy đủ của `song-service.ts` như sau:

```
import {Injectable} from '@angular/core';
import {Song} from '../model/song';

@Injectable({
  providedIn: 'root'
})
export class SongService {
  playlist: Song[] = [
    {
      id: 'CX11yw6YL1w',
      name: 'Céline Dion - Ashes (from the Deadpool 2 Motion Picture Soundtrack)'
    },
    {
      id: 'HBYirj2c_jw',
      name: 'Deadpool 2 - Take on Me'
    },
    {
      id: 'N602ncUKvlg',
      name: 'Nelly - Just A Dream'
    },
    {
      id: 'ue1Hwf8o7_U',
      name: 'Eminem - Love The Way You Lie ft. Rihanna'
    },
    {
      id: 'WpYeekQkAdc',
      name: 'The Black Eyed Peas - Where Is The Love?'
    }
  ];

  constructor() {}

  findSongById(id: string) {
    return this.playlist.find(item => item.id === id);
  }
}
```

Bước 7: Sử dụng lệnh **ng g c youtube-playlist** để tạo YoutubePlaylistComponent:

Trong file youtube-playlist.component.ts, khai báo một đối tượng songService trong constructor như sau:

```
constructor(private songService: SongService) {}
```

Và khai báo một đối tượng playlist và một phương thức getAll() để lấy ra danh sách bài hát ở SongService vừa khai báo ở trên và gọi phương thức getAll() trong ngOnInit:

```
import {Component, OnInit} from '@angular/core';
import {SongService} from '../service/song.service';
import {Song} from '../model/song';

@Component({
  selector: 'app-youtube-playlist',
  templateUrl: './youtube-playlist.component.html',
  styleUrls: ['./youtube-playlist.component.css']
})
export class YoutubePlaylistComponent implements OnInit {
  playlist: Song[] = [];

  constructor(private songService: SongService) {
  }

  ngOnInit() {
    this.getAll();
  }

  getAll() {
    this.playlist = this.songService.playlist;
  }
}
```

Cấu hình routing cho YoutubePlaylistComponent:

```
const routes: Routes = [
  {
    path: 'youtube',
    component: YoutubePlaylistComponent
  }
];
```

Trong file youtube-playlist.component.html sử dụng đoạn template sau để hiển thị danh sách bài hát:

```
<h2>Youtube playlist</h2>
<ul>
  <li *ngFor="let song of playlist">
    {{song.name}}
  </li>
</ul>

<router-outlet></router-outlet>
```

Trong file app.component.html sử dụng cặp thẻ <router-outlet></router-outlet> để hiển thị nội dung của component tương ứng và thêm đoạn code như sau:

```
<a routerLink="/youtube">Youtube</a>
<router-outlet></router-outlet>
```

Bước 8: Tạo component YoutubePlayerComponent bằng câu lệnh sau: ng g c youtube-player

Trong file youtube-player.component.ts sử dụng các phương thức như sau:

```
import { Component, OnInit } from '@angular/core';
import { SongService } from '../service/song.service';
import { DomSanitizer } from '@angular/platform-browser';
import { ActivatedRoute, ParamMap } from '@angular/router';

@Component({
  selector: 'app-youtube-player',
  templateUrl: './youtube-player.component.html',
  styleUrls: ['./youtube-player.component.css']
})
export class YoutubePlayerComponent implements OnInit {
  song: any;
  constructor(
    private songService: SongService,
    private activatedRoute: ActivatedRoute,
    private domSanitizer: DomSanitizer
  ) { }

  ngOnInit() {
    this.activatedRoute.paramMap.subscribe((paramMap: ParamMap) => {
      const id = paramMap.get('id');
      this.song = this.songService.findSongById(id);
    });
  }

  getSrc() {
    const url = 'https://www.youtube.com/embed/' + this.song.id;
    return this.domSanitizer.bypassSecurityTrustResourceUrl(url);
  }
}
```

Cập nhật lại file html của youtube-player.component.html:

```
<div *ngIf="song; else tpl">
  <h2>Song name: {{song.song}}</h2>
  <iframe width="560" height="315" [src]="getSrc()" frameborder="0" allow="autoplay; encrypted-
media" allowfullscreen></iframe>
</div>

<ng-template #tpl>
  <h2>Nothing to play</h2>
</ng-template>
```

Cập nhật lại file app-routing-module.ts, khai báo thêm routes cho youtube-player component:

```
import {NgModule} from '@angular/core';
import {RouterModule, Routes} from '@angular/router';
import {YoutubePlaylistComponent} from './youtube-playlist/youtube-playlist.component';
import {YoutubePlayerComponent} from './youtube-player/youtube-player.component';

const routes: Routes = [
  {
    path: 'youtube',
    component: YoutubePlaylistComponent,
    children: [{
      path: ':id',
      component: YoutubePlayerComponent
    }]
  }
];

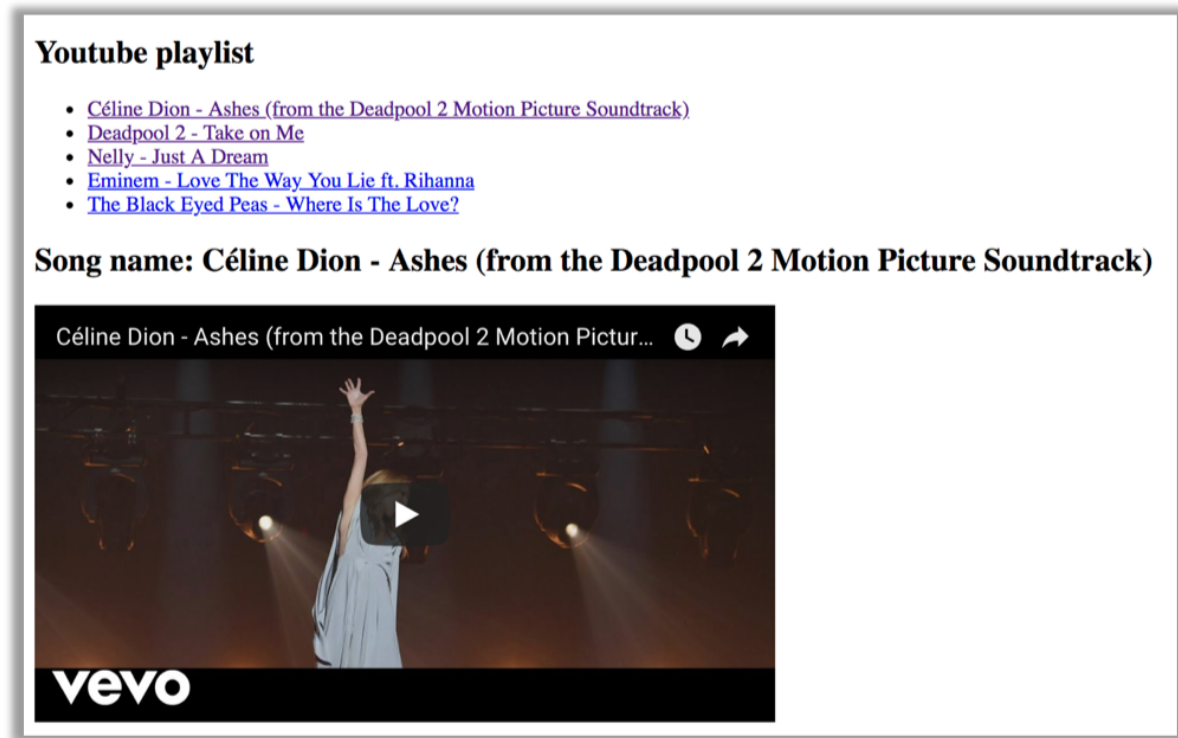
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {
}
```

Cập nhật lại file youtube-playlist.component.html, thêm thẻ a để routerLink đến hiển thị nội dung của youtube-player component:

```
<h2>Youtube playlist</h2>
<ul>
  <li *ngFor="let song of playlist">
    <a [routerLink]="['/youtube', song.id]">{{song.name}}</a>
  </li>
</ul>

<router-outlet></router-outlet>
```

Bước 10: kiểm tra kết quả



Checkout sang nhánh develop để xem mã nguồn tham khảo: <https://github.com/codegym-vn/wfda-angular-playlist-app>

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

