

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

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [6. Angular Form](#) / [\[Thực hành\] Tạo ứng dụng todo](#)

[Thực hành] Tạo ứng dụng todo

Mục tiêu

Nắm bắt được về các thao tác với form trong Angular.

Mô tả

Xây dựng ứng dụng todo để quản lý các công việc cần làm.

Hướng dẫn

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



Tại cửa sổ tiếp theo gõ chữ **y** và ấn enter



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>
```

Ứng dụng đã được tạo thành công

Bước 2: Mở dự án vừa tạo bằng Webstorm

Bước 3: Mở cửa sổ terminal trong Webstorm và gõ lệnh **npm i bootstrap -s** để tải và cài đặt bootstrap về dự án.

Bước 4: Trong file src/styles.css thêm đoạn code sau để import thư viện bootstrap vào trong dự án:

```
@import "~bootstrap/dist/css/bootstrap.css";
```

Bước 5: Mở cửa sổ terminal và gõ lệnh **ng g c todo** để tạo component mới.

Bước 6: Tạo một interface có tên là Todo bằng câu lệnh **ng g i todo** và sau đó thêm các thuộc tính dưới đây vào cho interface này:

```
export interface Todo {
  id?: number;
  content?: string;
  complete?: boolean;
}
```

Bước 7: Trong file todo.component.ts tiến hành khởi tạo một mảng các đối tượng todo như sau:

```
todos: Todo[] = [];
```

Sau đó khởi tạo một đối tượng FormControl có tên là content:

```
content = new FormControl();
```

Bước 8: Trong file todo.component.html thêm đoạn code như sau:

```
<h2 class="text-center">
  Todo App
</h2>
<div class="col-md-8 offset-md-2">
  <input type="text" class="form-control" [formControl]="content" (change)="change()">
  <ul>
    <li (click)="toggleTodo(i)"
      [class.complete]="todo.complete"
      *ngFor="let todo of todos; index as i">
      {{todo.content}}
    </li>
  </ul>
```

```
</div>
```

Bước 9: Đặt thêm stylesheet sau vào file style của component(todo.component.css):

```
.complete {
  text-decoration: line-through;
}
```

Bước 10: Thêm đoạn code sau vào file todo.component.ts:

```
import {Component, OnInit} from '@angular/core';
import {Todo} from '../todo';
import {FormControl, FormGroup, Validators} from '@angular/forms';

let _id = 1;

@Component({
  selector: 'app-todo',
  templateUrl: './todo.component.html',
  styleUrls: ['./todo.component.css']
})
export class TodoComponent implements OnInit {
  todos: Todo[] = [];
  content = new FormControl();

  constructor() {}

  ngOnInit() {}

  toggleTodo(i: number) {
    this.todos[i].complete = !this.todos[i].complete;
  }

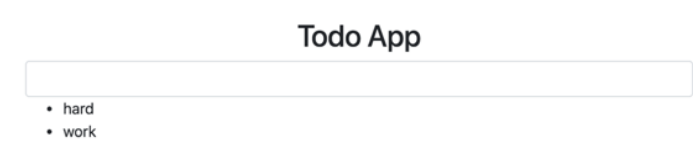
  change() {
    const value = this.content.value;
    if (value) {
      const todo: Todo = {
        id: _id++,
        content: value,
        complete: false
      };
      this.todos.push(todo);
      this.content.reset();
    }
  }
}
```

Bước 11: Thêm vào app.component.html:

```
<app-todo></app-todo>
```

Bước 12: Kiểm tra kết quả có giống với hình sau:

Lưu ý: khi click vào phần tử ở list bên dưới, phần tử đó sẽ bị gạch ngang để biểu thị đã hoàn thành, có thể click nhiều lần để toggle trạng thái



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

Hướng dẫn tiếp theo:

Nhường dần 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	▶ Comments (0)

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

