



C1- Code & Go

Angular2

Angular2 ToDoList

Introduction to Angular2



Angular2

Foreword

Today, you are going to discover Angular2.

You will be asked to do a **to-do list** like the one you did yesterday, but using Angular2 this time.

Each exercise will build up on the code of the preceding exercise, and integrate new notions.

Please read thoroughly each exercise, there is everything you need to understand the outcome expected from you.

Ask yourself the right questions and most importantly **think**.

You will create a repository named **MEAN_Pool_Day_04/**

There are several ways to install Angular2:

- Manually installing Angular 2 with the guidance of the quickstart documentation
- Installing Angular 2 by cloning the Quickstart repo
- Installing Angular 2 with the Angular-Cli

The whole Angular2 project will be submitted in your repository (no need for a separate exercise folder each time).



Exercise 01

2 pts

Create a new Angular2 project. You must create an empty home page for your site : **index.html**.

This page must be accessible with the route:
GET /

For testing purposes, you can use **http-server** or whatever suits you.



Exercise 02

4 pts

Now create a new component **todoComponent** in the file named `todocomponent.ts`

In this component, you will add two variables:

- A list of your Todo: **todos**.
- A string **newTodo**.

You will also create a new view, **todos.html**

This view will contains some elements:

- A title **To-do-List first name, name**
- A form containing:
 - An input field that will correspond to `newTodo` in the `todoComponent`
 - A button **ADD**



Exercise 03

3 pts

You will add on your view the todo list.

All those todos will need a checkbox which code for their status. The status will be false by default and true when checked.

Also, you'll need to add a delete button for each one of your todo. Whenever you click on it, it will delete the associated todo.

Once this is done, you will now give some methods to your components:

- **addTodo** : A method that adds a todo to the list.
- **deleteTodo** : Same as above but to delete one todo.



ngModel, ngFor



Exercise 04

2 pts

You will add a new button in your view: **deleteSelected**.

Whenever you will click on that button, it will call the `deleteSelected` method of your component:
This method will delete all the checked todos.



Exercise 05

2 pts

For safety, add some unit test for your application.

Using Karma, you must create 3 tests for your main controller:

- A test that checks that the `todoList` is set and empty at the start
- A test to check that the function used to add a task works
- And a test to check that the function used to remove a task works



Exercise 06

3 pts

You may have seen that your application does not save its data (when you reload your page, all the tasks you added are gone).

To modify this behavior, you'll have to use a local storage, and save the elements of todoList in it.



Watch local storage in Angular2



Exercise 07

2 pts

Now that our data are saved, we need something to be able to sort through it.

Add an input field in which, when you type anything, it filters the task of that todoList.

Example: if you type "one" in your search field, you will only see the tasks that have "one" in their name.



Exercise 08

2 pts

Change your todoList so that tasks are ordered by status with the tasks that are not completed coming first.