# Commandes Angular

## Créer un projet

*ng new my-app*

**Explication :**

*ng new* permet de créer un projet Angular, avec tous les fichiers et dépendances nécessaires pour une application de base. (Cela peut prendre un peu de temps)

Si l’argument *my-app* a été donné, il créera alors un dossier *my-app* dans lequel les sources du projet seront placées.

Un prompt vous sera ouvert dans lequel Angular vous proposer de choisir le type à utiliser pour les fichiers de styles (css, less, scss, sass).

Il est possible d’ajouter un paramètre à cette commande, nous allons utiliser *–routing*, qui permet d’avoir une meilleure organisation des fichiers, pour les routes.

## Lancer un serveur

Afin de tester notre application il faut utiliser la commande : *ng serve –open* dans votre invite de commande, tout en étant dans le dossier de votre application Angular.

## Générer un component (un nouveau composant angular)

Pour générer un component on utilise la commande : *ng generate component xxxxxx* dans votre invite de commande, où ‘xxxxxx’ correspond au nom du component que vous souhaitez créer.

## Ajouter Bootstrap

Pour l’ajouter il faut utiliser la commande : *npm install bootstrap@next*

(Documentation de Bootstrap 5 : <https://getbootstrap.com/docs/5.0/getting-started/introduction/>)

### Intégration dans le projet :

* Une fois la commande passée, Boostrap est ajoutée comme dépendance à votre projet, mais il n’est pour le moment pas utilisable !
* Vous pouvez constater que le *package.json* a été modifiée et qu’un dossier *bootstrap* a été ajouté à votre *nodes\_modules*
* Pour intégrer boostrap comme bibliothèque scss utilisable, il faut l’ajouter comme import
* Afin d’optimiser notre code scss, je recommande de centraliser les import de bibliothèques de style dans le fichier *styles.scss*, c’est donc dans ce dossier qu’il faut ajouter boostrap, via la ligne de code :

*@import "node\_modules/bootstrap/scss/bootstrap";*

* Pour se laisser l’opportunité de pouvoir surcharger les variables ou classes de bootstrap, je recommande de vous créer un dossier « *styles*» à la racine du dossier « *src*», dans lequel vous ajouterez un fichier scss, du nom de votre choix, qui regroupera vos classes personnalisées pour votre application.

De plus, vous devez importer ce fichier dans le *styles.scss* (situé à la racine du dossier « *src* ») APRES l’import de boostrap, afin justement de pouvoir surcharger les variables ou classe de boostrap.

(PS : lorsque vous intégrez bootstrap à votre application, le *package.json* est mis à jour, pensez à faire un *npm install* lorsque que vous récupérez un projet qui vient d’intégrer boostrap)

## Création de services

Pour générer un service on utilise la commande : *ng generate service xxxxxx* dans votre invite de commande, où ‘xxxxxx’ correspond au nom du service que vous souhaitez créer.

## Ajouter sprintf

// Installer la dépendance sprintf au projet

npm install sprintf-js --save

// Installer la version .ts

npm i --save\_dev @types/sprintf-js