**Install the Angular CLI**

npm install -g @angular/cli

**Create a workspace and initial application**

ng new my-app

**Serve the application**

cd my-app

ng serve --open

**to change port change in package.json file**

"start": "ng serve --open --port 4201",

Steps to add new project in git

git init

git add .

git commit -m "added web socket library"

git remote add origin \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

git push -u origin master

**Create component**

ng generate component heroes

* You used the CLI to create a second HeroesComponent.
* You displayed the HeroesComponent by adding it to the AppComponent shell.
* You applied the UppercasePipe to format the name.
* You used two-way data binding with the [ngModel](https://angular.io/api/forms/NgModel) directive.
* You learned about the AppModule.
* You imported the [FormsModule](https://angular.io/api/forms/FormsModule) in the AppModule so that Angular would recognize and apply the [ngModel](https://angular.io/api/forms/NgModel)directive.
* You learned the importance of declaring components in the AppModule and appreciated that the CLI declared it for you.

ng generate component hero-detail

ng generate service hero

ng generate component [messages](https://angular.io/api/service-worker/SwPush#messages)

ng generate service [message](https://angular.io/api/common/http/HttpErrorResponse#message)

ng generate module app-routing --flat --module=app

--flat puts the file in src/app instead of its own folder.  
--module=app tells the CLI to register it in the [imports](https://angular.io/api/core/NgModule#imports) array of the AppModule.

RouterLink directive that turns user clicks into router navigations. It's another of the public directives in the RouterModule.

ng generate component dashboard