Angular-8

Angular CLI Part-2

Presentation BY RAJ PRUDHVI

Contact us:https://training.uplatz.com
Email: info@uplatz.com
Phone: +44 7836 212635

Uplatz

Angular cli

Part-2

- Angular CLI generate Component
- Angular CLI generate Service
- Angular CLI generate class , interface
- Angular CLI ng serve options

Uplatz

Angular cli generate component

ng generate component ComponentName [or] ng g c ComponentName

Example : ng g c uplatz

When we execute above command , several things happen

- ✓ A folder with name uplatz is created
- ✓ The component files (Component class, View template, CSS file and the spec file) are created and placed inside the folder “ uplatz “
- ✓ The root module file (app.module.ts) is also updated with our new component i.e the required import statement to import the uplatz component from the component file is included and the component is also declared in the declarations array of the @NgModule() decorator

Uplatz

Angular cli generate component

- By default a separate folder is created for every new component that we generate, and the component files (.ts , .css , .html & .spec) are placed in this folder.
- This newly created folder is placed in the app folder by default.
- If you want the newly created folder to be placed in a different folder other than the app folder, simply include the folder name in the ng generate command
- To create a component without a folder, use **--flat** option with the ng generate command
- A flat component is a component that is created with --flat option. This component does not have it's own folder. By default the flat component files are placed in the "app" folder. If you want to place them in a different folder instead, specify the folder name along with the ng generate command.

Angular cli generate component

- Just like how we can use the **--dry-run** flag with “ **ng new** " command, we can also use it with ng generate command.
- The **--dry-run** flag reports the files and folders that will be generated, without actually generating them. Once you are happy with what it is going to generate, you can remove the **--dry-run** flag and execute the command.
- If you want an inline template and styles instead of an external template and style sheet, use -it flag for inline template and -is flag for inline styles. Along the same lines, if you do not want a spec file use --skip-tests.

Uplatz

Angular cli generate service

- Generating services is similar to generating components.
- To generate a service we use the following command

ng generate service serviceName or ng g s serviceName

For example, to generate a student service we use the following command.

ng generate service student

The above command generates the service and the spec file. What it does not do is register the service. Remember for us to be able to use the service, we must register the service.

Angular cli generate service

We can also use the **--dry-run** flag or it's alias -d to see what Angular CLI generates. Notice in the following command we are using -d option, so Angular CLI simply report the files it is going to generate

ng g s student -d

The above command generates the service and the spec file. If you do not want the spec file, simply use **--skip-tests**

ng g s student -d --skip-tests

Angular cli generate service

When generating a component, Angular CLI by default creates a folder for the component and places all the component files in that folder. A service on the other hand will not have it's own folder. If you want a folder of it's own for a service that the Angular CLI is generating, set **--flat** option to false as shown below.

ng g s student -d –skip-tests --flat=false

Uplatz

Angular cli generate class

So far we have discussed generating angular features like components, services, directives etc. We can use the Angular CLI to generate TypeScript features as well. In this session we will discuss generating TypeScript features like classes, interfaces using the Angular CLI.

To generate a class use

ng generate class className or ng g cl className

Angular cli generate class

For example, to generate an employee class use

ng g cl employee

The above command places the employee class directly in the “ **app "** folder. Instead if you want the employee class in a different folder, simply prefix the name of the folder. The command below creates a folder with name "**employee**" and then creates the "**employee**" class in it.

ng g cl employee/employee

Angular cli generate interface

To generate an interface use

ng generate interface interfaceName or ng g i interfaceName

To generate an interface named uplatz

ng g i uplatz

Uplatz

Angular cli ng serve options

We can use ng serve command to build and run angular application in development mode
To see the list of all options that we can use with "**ng serve**" command use --help option

ng serve --help

The following command, builds and launches the application in your default browser.

ng serve --open

Angular cli ng serve options

Option	Alias	Default	Purpose
--watch	-w	true	Run build when files change
--live-reload	-lr	true	Whether to reload the page on change
--open	-o	false	Opens the url in default browser
--port	-p	4200	The port on which the server is listening

Thank You

Contact us:https://training.uplatz.com

Email: info@uplatz.com

Phone: +44 7836 212635

Uplatz