Command :

|  |  |  |
| --- | --- | --- |
| **Command** | **Alias** | **Description** |
| add |  | It is used to add support for an external library to your project. |
| build | b | It compiles an Angular app into an output directory named dist/ at the given output path. Must be executed from within a workspace directory. |
| config |  | It retrieves or sets Angular configuration values in the angular.json file for the workspace. |
| doc | d | It opens the official Angular documentation (angular.io) in a browser, and searches for a given keyword. |
| e2e | e | It builds and serves an Angular app, then runs end-to-end tests using Protractor. |
| generate | g | It generates and/or modifies files based on a schematic. |
| help |  | It provides a list of available commands and their short descriptions. |
| lint | l | It is used to run linting tools on Angular app code in a given project folder. |
| new | n | It creates a new workspace and an initial Angular app. |
| run |  | It runs an Architect target with an optional custom builder configuration defined in your project. |
| serve | s | It builds and serves your app, rebuilding on file changes. |
| test | t | It runs unit tests in a project. |
| update |  | It updates your application and its dependencies. See https://update.angular.io/ |
| version | v | It utputs Angular CLI version. |
| xi18n |  | It extracts i18n messages from source code. |

* Bootstrap – angular kahase start
* Package.json – only contain references of dependencies and devdependencies.

Devdep are required at development and and only dep are required at frontend.

-insatll all dependencies from npm install

* Package.lock.json – detailed information related to packages we have installed and which is in package.json file
* Editorconfig – vs code related, not angular related
* Gitignore – contain all file and folders which arte not to push in git commit.
* Angular.json – angularcli configuration file, it provide project specific configuration defaults for build and development provided by the angular cli.
* Karma.conf.js – related to the testing, test project -karma needs to know about project this done via confi.file
* Read.me – helping purpose contains commands with details
* Tsconfig.spec.json – testing environment
* Tsconfig.app.json – app specific and just extends tsconfig.json

-typeScript compiler configuration

* Node\_modules – NPM-node package manager, which installs packages locally into a project specifically into the node\_modules folder

-from there package code can be included into a project

-project is built using node and that the node\_modules folder contains dependencies only related to the build

It is library and when we create angular project at that time we need many library like css,angular,form,routing,testing

-npm install -rebuild whole folder

* Assets- in that extra images
* Index.html – It is load at first
* Main.js – is run at first and boostrap-html k ander puri angular file load karna
* Src -app – component – jisko hum reuse kar sakte he and pra developing isme hota he
* App-root is custom tag and component file me likha hota he selector me
* Npm start and ng serve are same

Npm start call or run ng serve

* Building is the process of compiling our project
* Environment – environment.prod.ts is use for production or at production and environment.ts use at development
* Angular uses components to build web pages and uses modules to bundle different components in to packages.
* Unit testing: Just as an atom is the smallest independent unit of matter, a unit test is the atom of tests—that is to say, a unit test is written for individual units or components of an application.
* Integration testing: Here, we test the integration of two or more units. Integration tests usually require external resources like network and database. An integration test does not test a complete workflow, nor does it test an isolated unit.
* E2E testing: In E2E testing, you test the workflow of your application from start to finish. In other words, you test for actions that real users can perform on your app to check that desired outputs are obtained.
* End-to-end Testing - End-to-end testing is performed for all testing processes at end.
* Unit Testing-Unit testing is performed for testing processes first.
* BrowserModule – The browser module is imported from @angular/platform-browser and it is used when you want to run your application in a browser.
* CommonModule – The common module is imported from @angular/common and it is used when you want to use directives - NgIf, NgFor and so on.

-Directives are classes that add additional behavior to elements in your Angular applications. Use Angular's built-in directives to manage forms, lists, styles, and what users see.

* FormsModule – The forms module is imported from @angular/forms and it is used when you build template driven forms.
* declarations: The components, directives, and pipes that belong to this NgModule.
* exports: The subset of declarations that should be visible and usable in the component templates of other NgModules.
* imports: Other modules whose exported classes are needed by component templates declared in this NgModule.
* bootstrap: The main application view, called the root component, which hosts all other application views. Only the root NgModule should set the bootstrap property.

* JIT and AOT are two ways to compile code in an Angular project. We use JIT in development mode while AOT is for production mode.
* We can easily implement features and debug in JIT mode since we have map file while AOT does not. However, The big benefit when we use AOT for production are reducing bundle size for faster rendering.