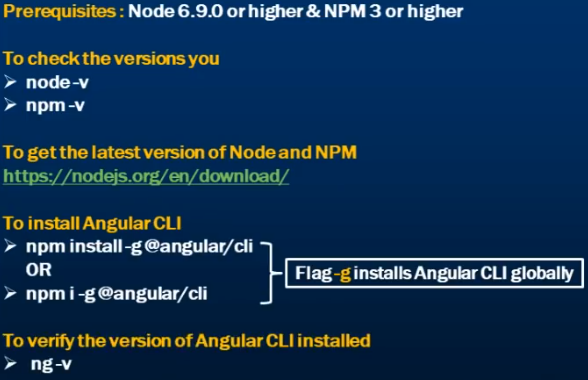
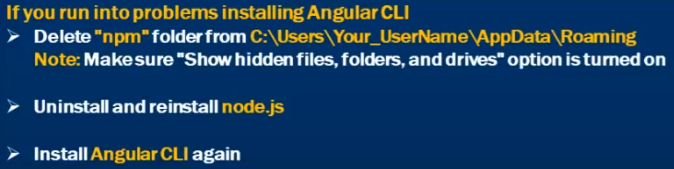
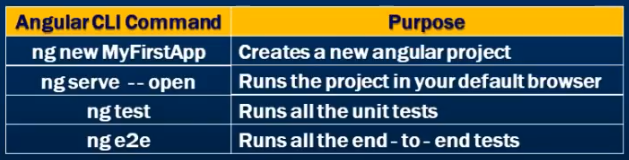
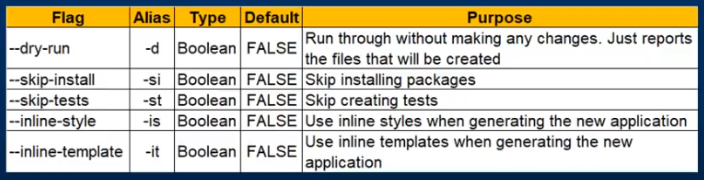


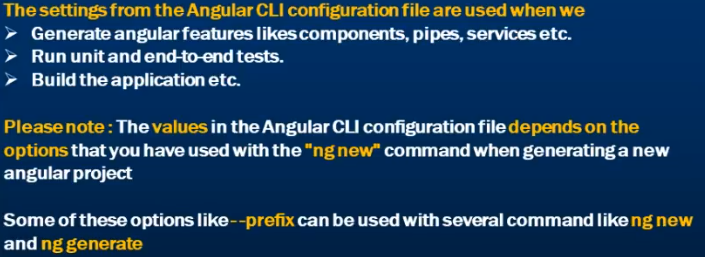
Installing Angular CLI:



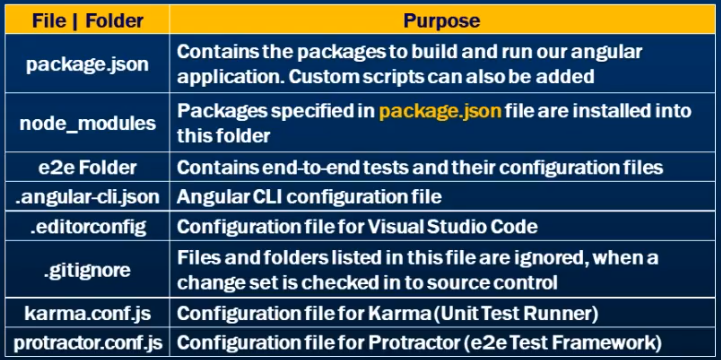


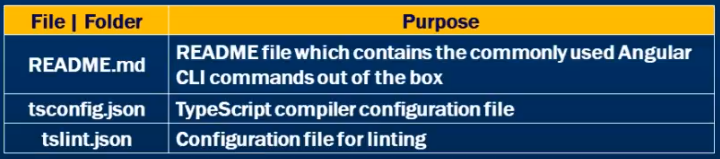




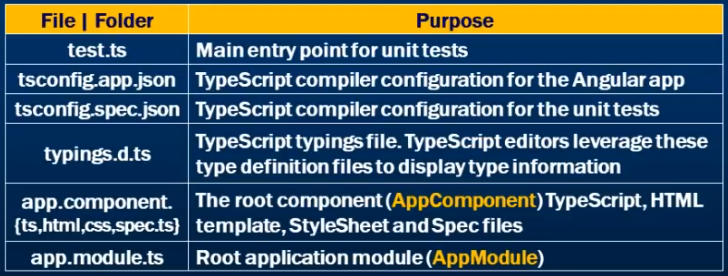
CLI configuration:

CLI Project Structure:



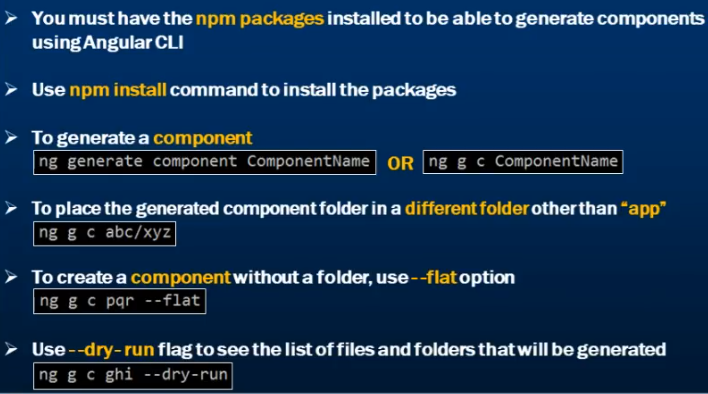


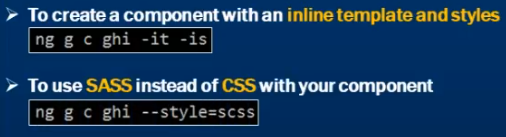




Polyfills: As we know Angular is built on the latest standards of web platform. Targeting such wide range of browsers is challenging because not all browsers support all features of modern browsers. This can be compensated by using Polyfills scripts as they implement missing features in Javascript.

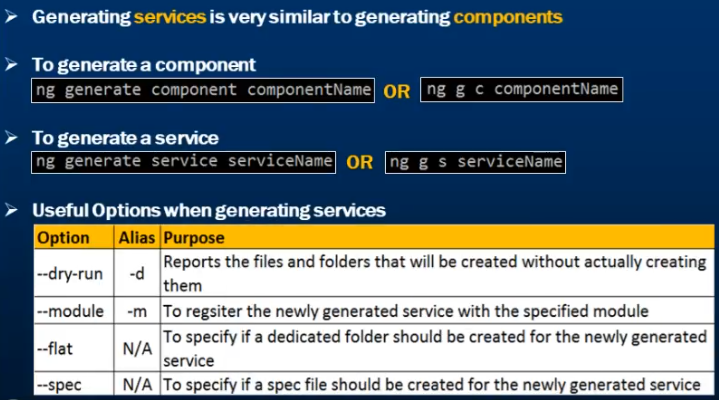
This Polyfills allow us to use an API regardless of whether a specific browser supports it or not.

Generate Components:



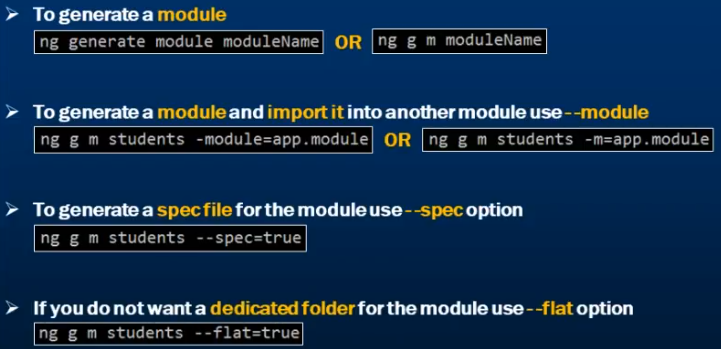
If you don’t want to spec file to be generated use --spec=false

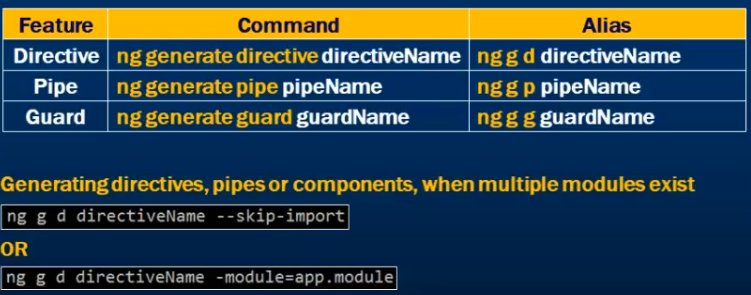
Generate Services:

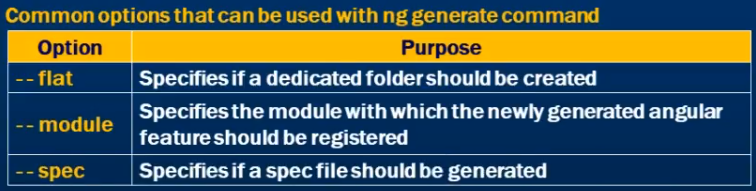


Note: Register service with Module: ng g s serviceName -m=app.module

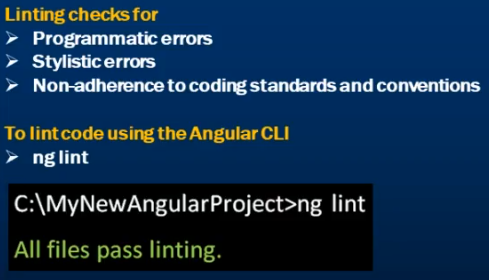
Generate Modules:

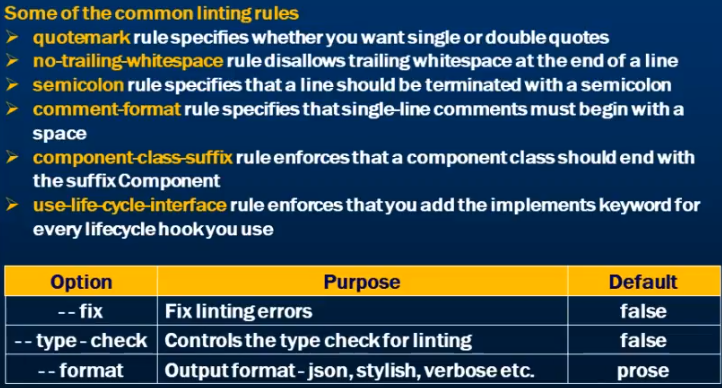


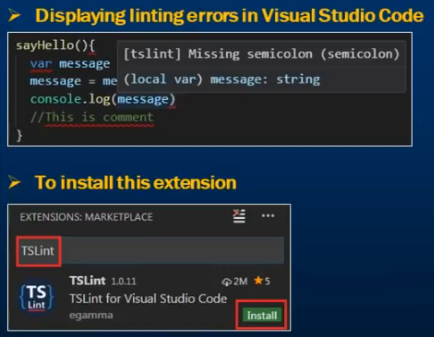






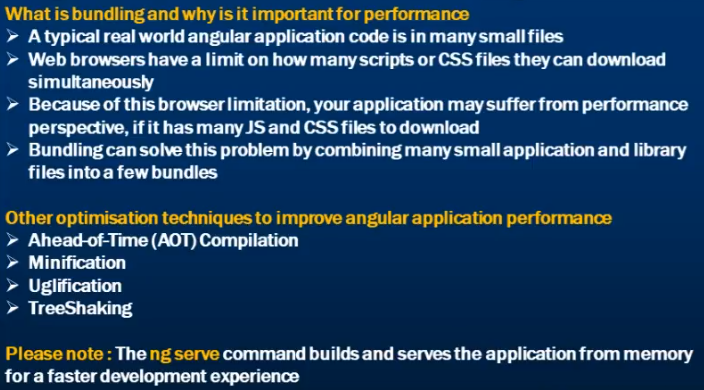
Linting:  


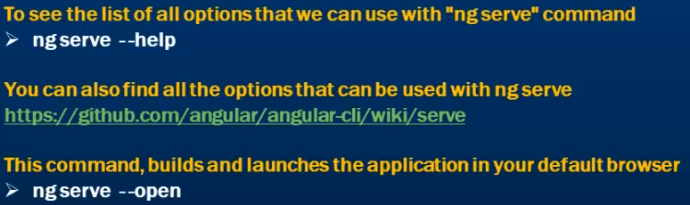


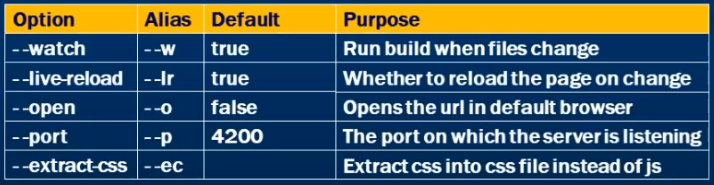


Routing:  

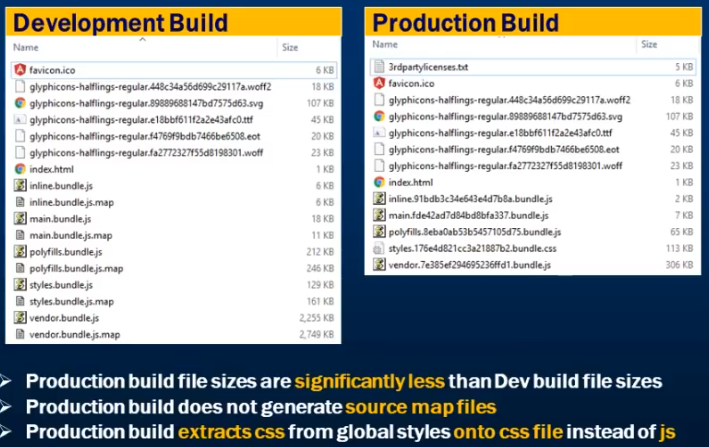

Running Angular App locally:  





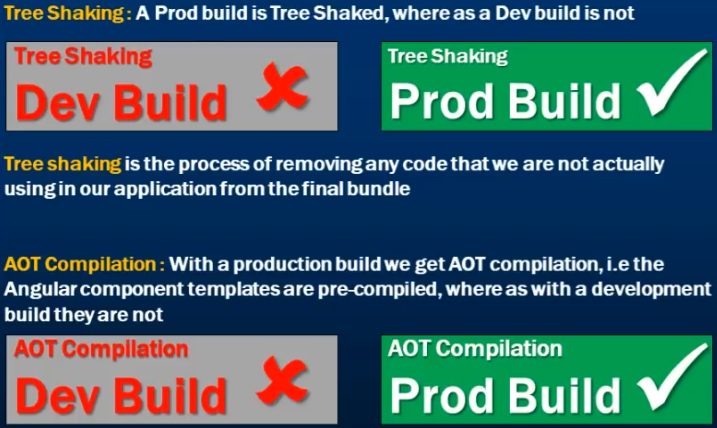


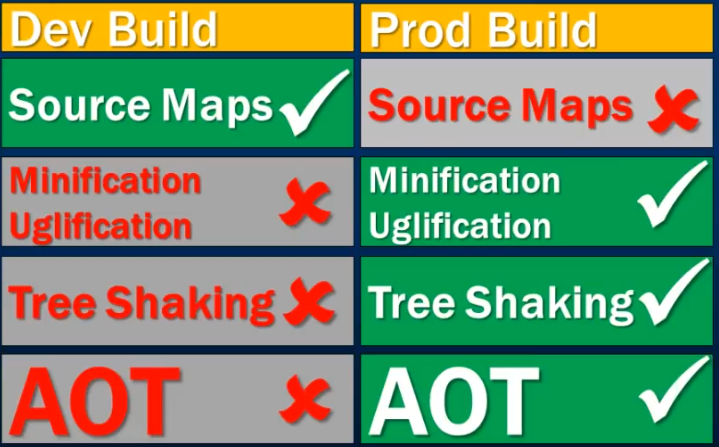


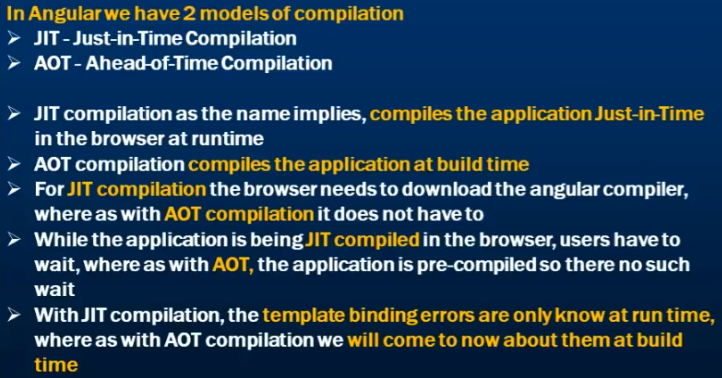
Dev Build VS PROD Build:

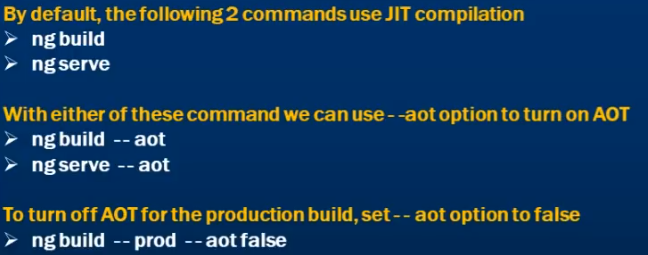




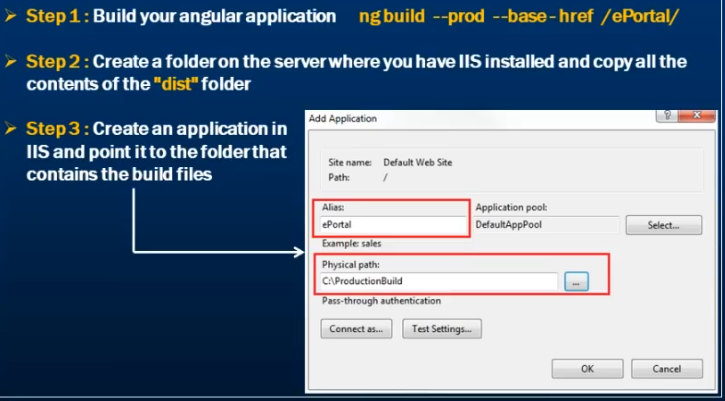








Deploy Angular app to IIS:





Angular Debugging:

1. Typescript: Take advantage of static typing
2. Debugger: Javascript debugger statement
3. Json Pipe: Inspect the data in HTML
4. Console Debugging: Angular built-in debugging tools
5. \*Augury plugin for Chrome: Visualize bugs with Augury
6. \*Angular Logger: Create your own logger