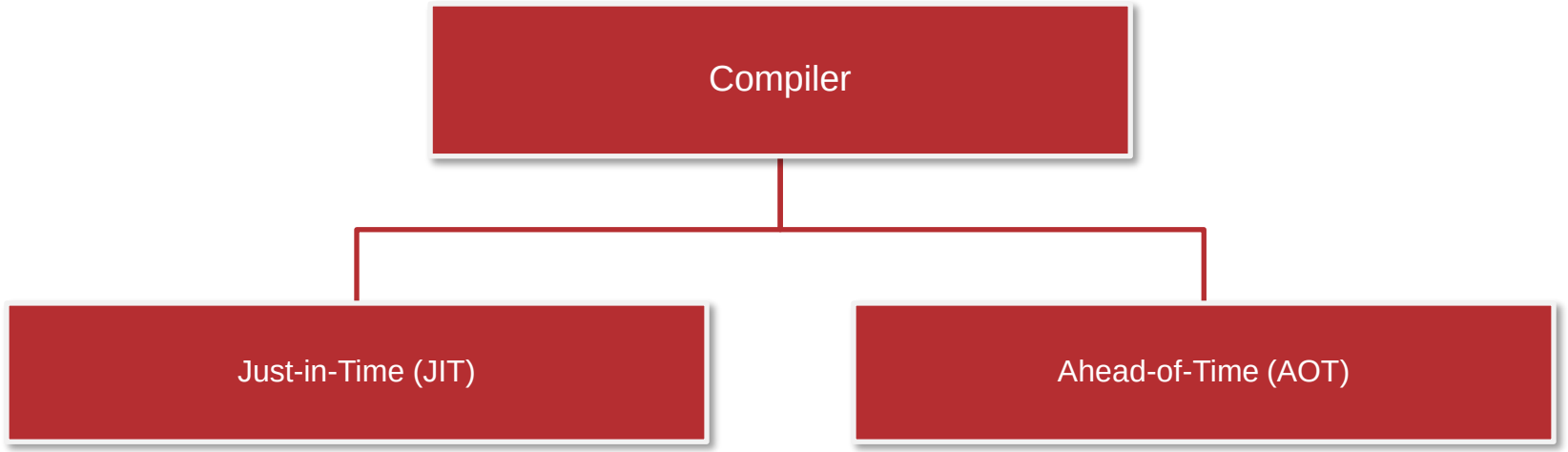
ANGULAR INTERVIEW QUESTIONS

1

What are different types of compilation in Angular ?

Angular offers two ways to compile your application



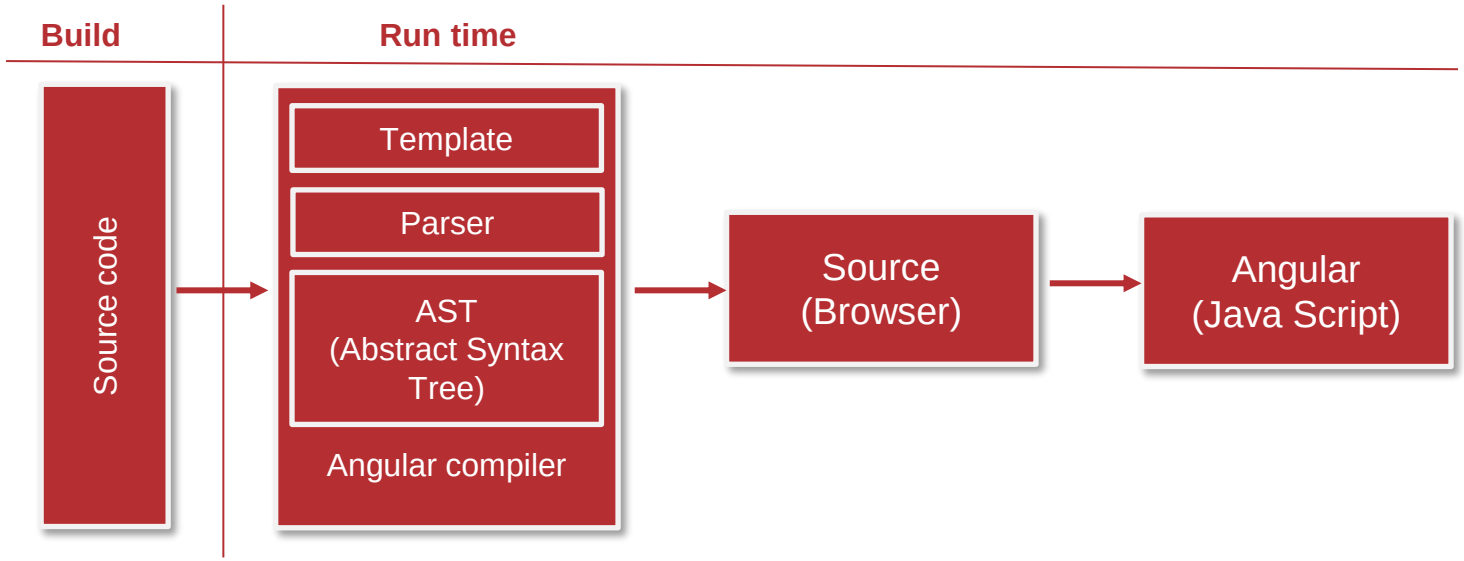
2

What is JIT ?

The just-in-time compiler converts the typescript and HTML code during runtime; rather than compiling the entire code at once, it does it for each component or section of the program as it is called and processed by the browser's interpreter. So, in a sense, the program is JIT-compiled as it is running in the browser. Below are the command which we use.

ng build
ng serve

Note: This was the default until Angular 8



3

What are the advantage of JIT ?

The benefits of JIT are listed below.

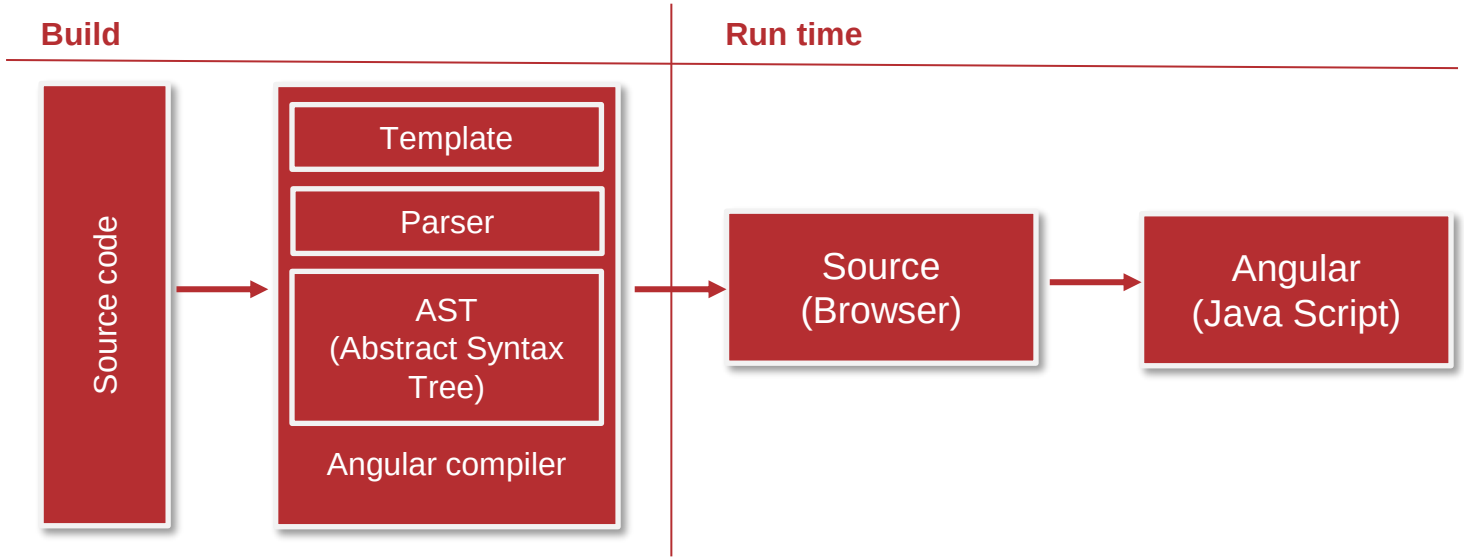
Less memory	JIT compilers need less memory usage.
Start time	JIT compilers run after a program starts.
Code optimization	Code optimization can be done while the code is running.
Reduce fault	Any page faults can be reduced.

4

What is AOT ?

AOT is a sort of compilation that compiles your program at build time. From Angular 9, this is the default starting. The kind of compilation (JIT or AOT) used when you execute the ng build (build only), or ng serve (build and serve locally) CLI commands depends on the value of the aot property in your build configuration as provided in angular.json. Aot is often set to true. Below are the command which we use.

ng build --aot
ng serve --aot



5

What are the advantage of AOT ?

The benefits of AOT are listed below.

Faster rendering	A pre-compiled version of the program is downloaded by the browser. so that it can render the application immediately without building the application.
Fewer asynchronous requests	It eliminates separate ajax queries by embedding external HTML templates and CSS style sheets within the application's JavaScript.
Smaller Angular framework download size	Does not call for the Angular compiler to be downloaded. As a result, the application payload is significantly decreased.
Detect template errors earlier	Identifies and communicates template binding issues during the actual construction process.
Better security	It turns HTML components and templates into JavaScript. There won't be any injection attacks as a result.

6

What are the compilation phase of AOT ?

There are three phases of AOT compilation

Code analysis

In this phase, the TypeScript compiler and *AOT collector* create a representation of the source. The collector does not attempt to interpret the metadata it collects. It represents the metadata as best it can and records errors when it detects a metadata syntax violation.

Code generation

In this phase, the compiler's **StaticReflector** interprets the metadata collected in phase 1, performs additional validation of the metadata, and throws an error if it detects a metadata restriction violation..

Template type checking

In this optional phase, the Angular template compiler uses the TypeScript compiler to validate the binding expressions in templates. You can enable this phase explicitly by setting the **strictTemplates** configuration option.

7

What are the ways to control AOT compilation ?

You can control your app compilation in two ways

Template compiler option

By providing template compiler option in the **tsconfig.json** file.

```
{
   "compilerOptions": {
     "experimentalDecorators": true,
                ...
   },
   "angularCompilerOptions": {
     "fullTemplateTypeCheck": true,
     "preserveWhitespaces": true,
                ...
   }
}
```

Meta data with decorators

By configuring Angular metadata with decorators. We use metadata.json file for this. The metadata.json file can be viewed as an abstract syntax tree (AST) representation of the overall structure of a decorator's metadata. The AOT collector emits metadata information in metadata.json files, one per.d.ts file, after scanning the metadata stored in the Angular decorators during the analysis phase.

What is package.json ?

When you build a new Angular application, the package.json file will appear among the newly created files and directories. The package.json file, which is found at the project root, includes details about your web application. The file's name, package, indicates that its primary use is to include information on the npm packages that have been installed for the project.

The primary package.json contains important package metadata, including the following:

Project name	The name of the project.
Version	The version number of the project.
Description	A brief description of the project.
Author	The author of the project.
License	The license under which the project is released.
Dependencies	A list of packages that are required by the project. These packages are installed when the project is set up and are used by the project at runtime.
Scripts	A list of scripts that can be run using npm. These scripts can perform tasks such as building the project, running tests, and starting a development server.

9

What is angular.json ?

An Angular app's angular.json file serves as the Angular CLI's configuration file. It keeps records of the project's dependencies, build and test setups, and other parameters. In addition to managing the various environments and configuration profiles for your project, this file lets you manage the build and runtime parameters for your Angular application. It is simpler to manage the configuration of your Angular application because the angular.json file offers a central area for declaring these parameters.

Architect	Locate the architect section in the file.
Target Environment	To configure the target environment, locate the "build" or "serve" option.
Configuration	You can define a new environment by adding a new key to the "configurations" section and giving it a descriptive name, like "production" or "staging.".
Another configuration	Add any necessary environment-specific configuration parameters, like baseHref or outputPath.

What is angular cli ?

The Angular CLI is a command-line interface tool that you use to initialize, develop, scaffold, and maintain Angular applications directly from a command shell.
With the help of Angular CLI, you can do below things:

Create Project	Create the project from scratch.
Scaffolding	Scaffold components, services or directives.
Brief/Lint	Brief or lint the code.
Serve application	Serve the Angular application efficiently.
Run unit test	It run unit tests and end-to-end tests.