

First Things First



Deborah Kurata

CONSULTANT | SPEAKER | AUTHOR | MVP | GDE

@deborahkurata | blogs.msmvps.com/deborahk/





Module Overview



Selecting a Language


Selecting an Editor

Setting up an Angular Application

About Modules



JavaScript Language Specification

A yellow square containing the letters 'JS' in a large, bold, black sans-serif font.

ECMAScript (ES)

ES 3

ES 5

ES 2015 (formerly known as ES 6)

- Must be transpiled



Selecting a Language

ES 5

- Runs in the browser
- No compile required

ES 2015

- Lots of new features (classes, let, arrow, etc.)

TypeScript

- Superset of JavaScript
- Strong typing
- Great IDE tooling

Dart

- No JavaScript



What Is TypeScript?



Open source language

Superset of JavaScript

Transpiles to plain JavaScript

Strongly typed

- TypeScript type definition files (*.d.ts)

Class-based object-orientation



Learning More



TypeScript Playground

<http://www.typescriptlang.org/Playground/>

Pluralsight Courses

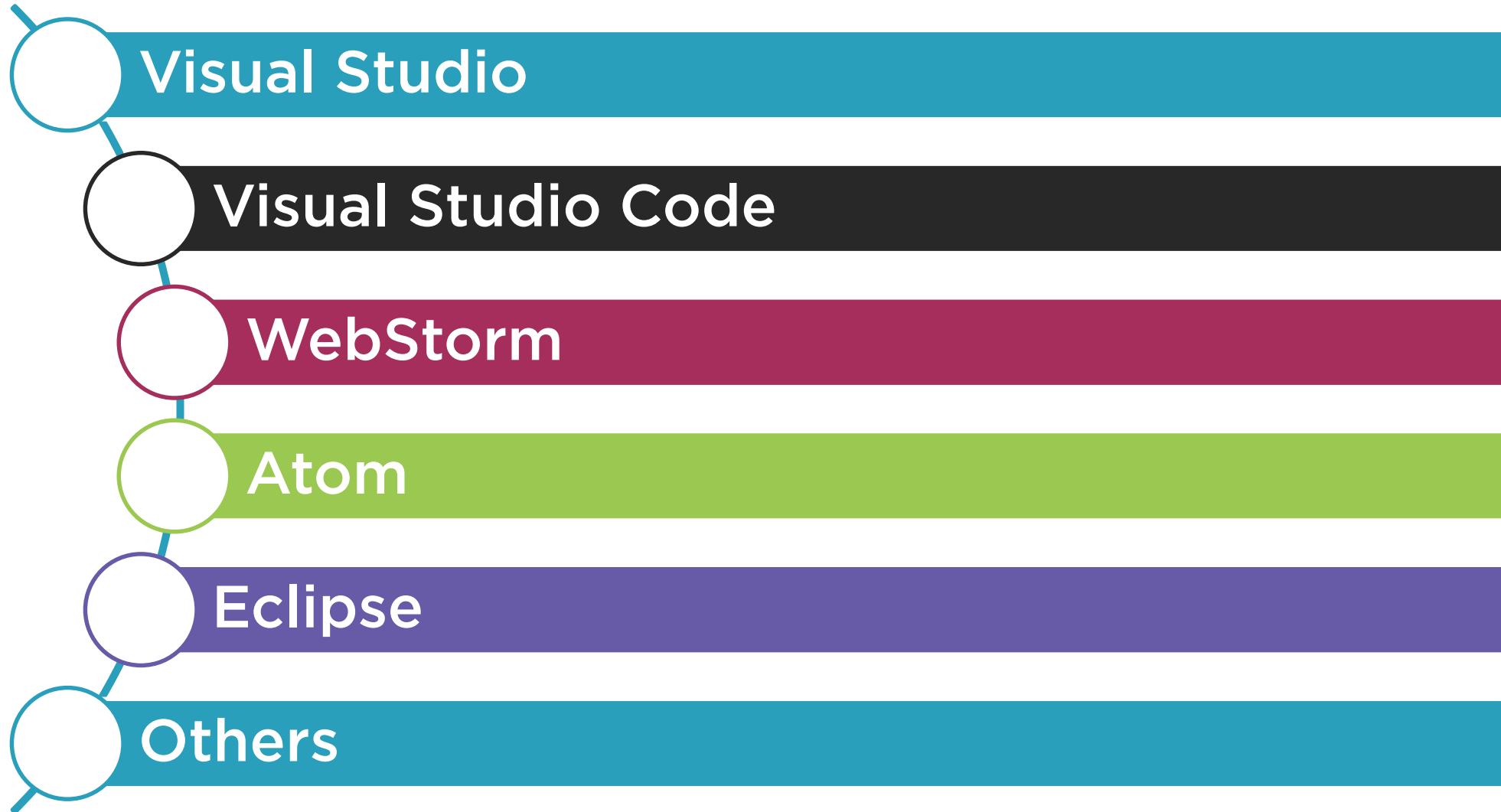
"TypeScript Fundamentals"

"AngularJS with TypeScript"

"Using ES6 with TypeScript"



TypeScript Editors



Visual Studio Code

The image shows the Visual Studio Code website and a screenshot of the Visual Studio Code editor interface. The website header includes the Visual Studio Code logo, navigation links (Docs, Updates, Blog, Extensions, FAQ), a search bar, and a green 'Download' button. A banner below the header announces 'Version 1.4 is now available! Read about the new features and fixes in July.' The main content area features the text 'Code editing. Redefined.' and 'Free. Open source. Runs everywhere.' Below this, there are download links for Windows, Linux (deb, rpm), and OS X, each with a corresponding icon and a 'Zip archive' link. The right side of the image is a screenshot of the Visual Studio Code editor. The editor window shows a file named 'www.ts' with the following code:

```
1 import app from './app';
2 import debugModule = require('debug');
3 import http = require('http');
4
5 const debug = debugModule('node-express-typescript:server');
6
7 // Get port from environment and store in Express.
8 const port = normalizePort(process.env.PORT || '3000');
9 app.set('port', port);
10
11 // create express server
12 const server = express();
13 server.listen(port);
14 server.on('error', (err) => {
15   if (err.syscall !== 'listen') {
16     throw err;
17   }
18
19   const bind = typeof port === 'string'
20     ? ` ${port} `
21     : ` ${port} `;
22
23   // Note: localhost is not available during development
24   throw new Error(`listen() ${err.message} on ${bind}`);
25 });
26
27 function normalizePort(val: any): number|string|boolean {
28   let port = parseInt(val, 10);
29
30   if (isNaN(port)) {
31     // invalid port
32     return false;
33   }
34
35   if (port < 0) {
36     // negative port
37     return false;
38   }
39
40   if (port > 0) {
41     return port;
42   }
43
44   // default: localhost
45   return 'localhost';
46 }
```

The editor interface also shows the 'EXTENSIONS' sidebar on the left, listing various extensions like C#, Python, Debugger for Chrome, C/C++, Go, and ESLint, each with an 'Install' button. The status bar at the bottom indicates the current file is 'www.ts', line 9, column 21, with 2 spaces, UTF-8 encoding, LF line endings, and TypeScript language.

<https://code.visualstudio.com/>



Learning More



Visual Studio Code Site

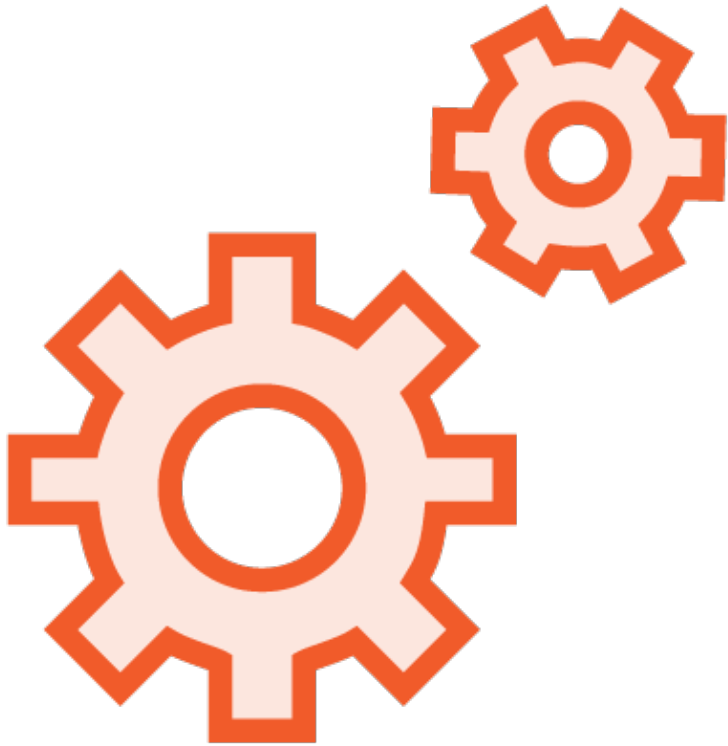
<https://code.visualstudio.com/>

Pluralsight Course

"Visual Studio Code"



Setting up Our Environment



npm

Set up the Angular application



npm



Node Package Manager

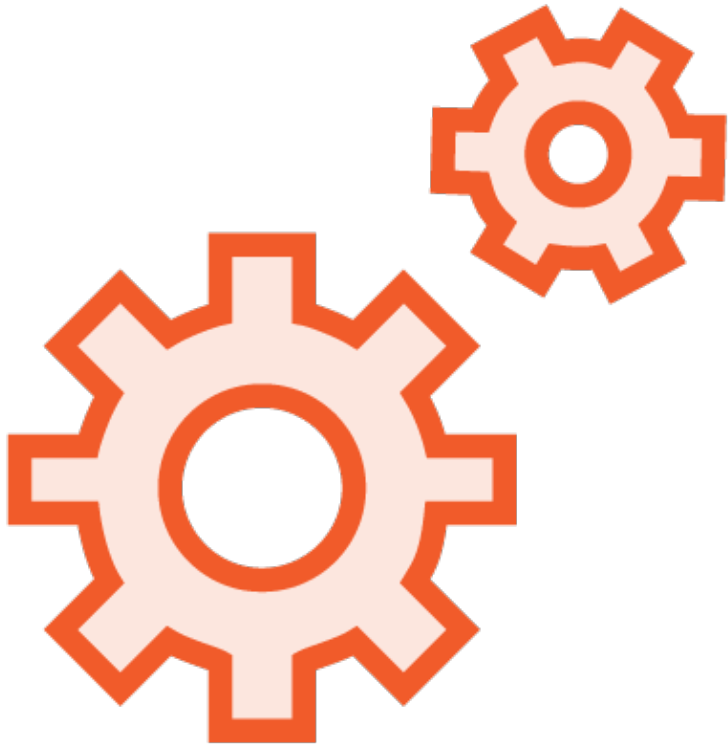
Command line utility

Installs libraries, packages, and applications

<https://www.npmjs.com/>



Setting up an Angular Application



1. **Create an application folder**
2. **Add package definition and configuration files**
3. **Install the packages**
4. **Create the app's Angular Module**
5. **Create the main.ts file**
6. **Create the host Web page (index.html)**

Setting up an Angular Application



Manually perform each step

Download the results of these steps

<https://github.com/angular/quickstart>

Angular CLI

<https://github.com/angular/angular-cli>

Starter files

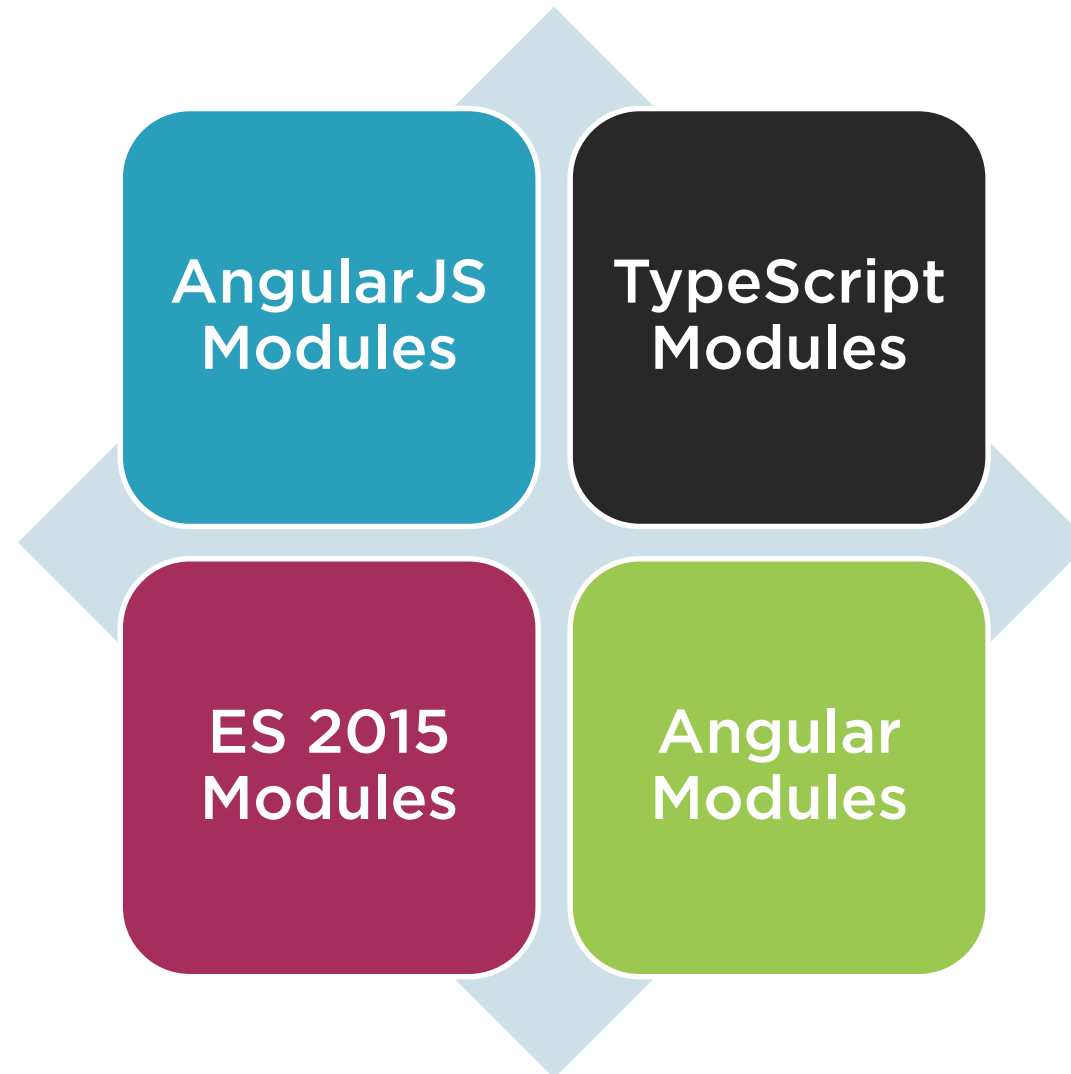
<https://github.com/DeborahK/Angular-GettingStarted>



Modules

Namespaces

Code Organization



ES 2015 Modules

Export

product.ts

```
export class Product{  
}
```

product.js

```
function Product() {  
}
```



Transpile

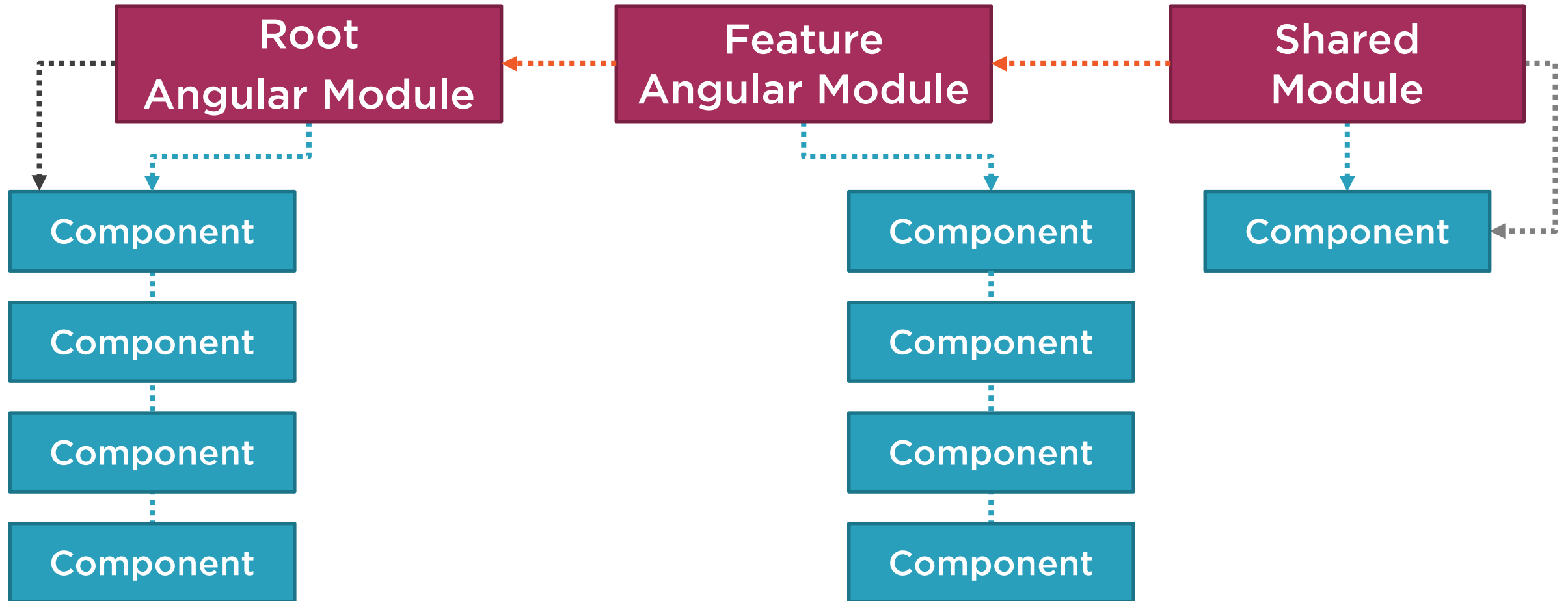
Import

product-list.ts

```
import { Product } from  
'./product'
```



Angular Modules



Modules

ES Modules

Code files that
import or export something

Organize our code files

Modularize our code

Promote code reuse

Angular Modules

Code files that
organize the application into cohesive
blocks of functionality

Organize our application

Modularize our application

Promote application boundaries





Web Browser

Web
Server

URL Request (www.mysite.com)

Response

index.html

JavaScript



Summary



Selecting a Language

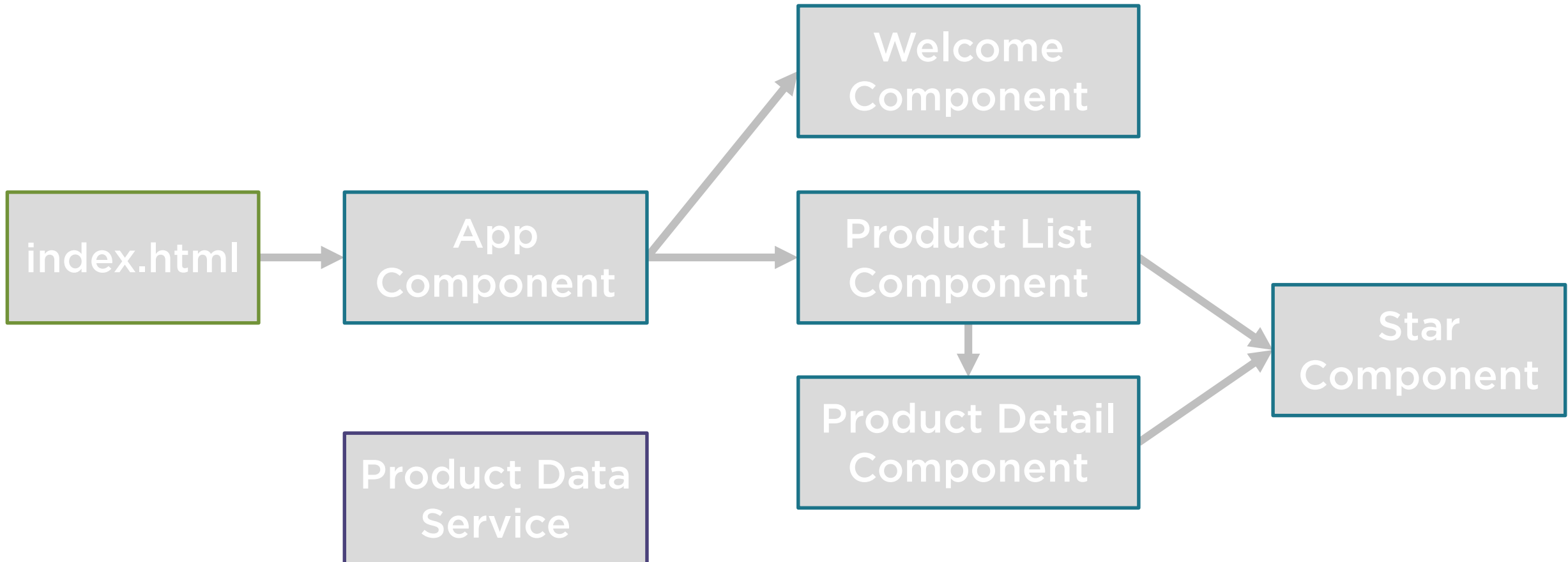
Selecting an Editor

Setting up an Angular Application

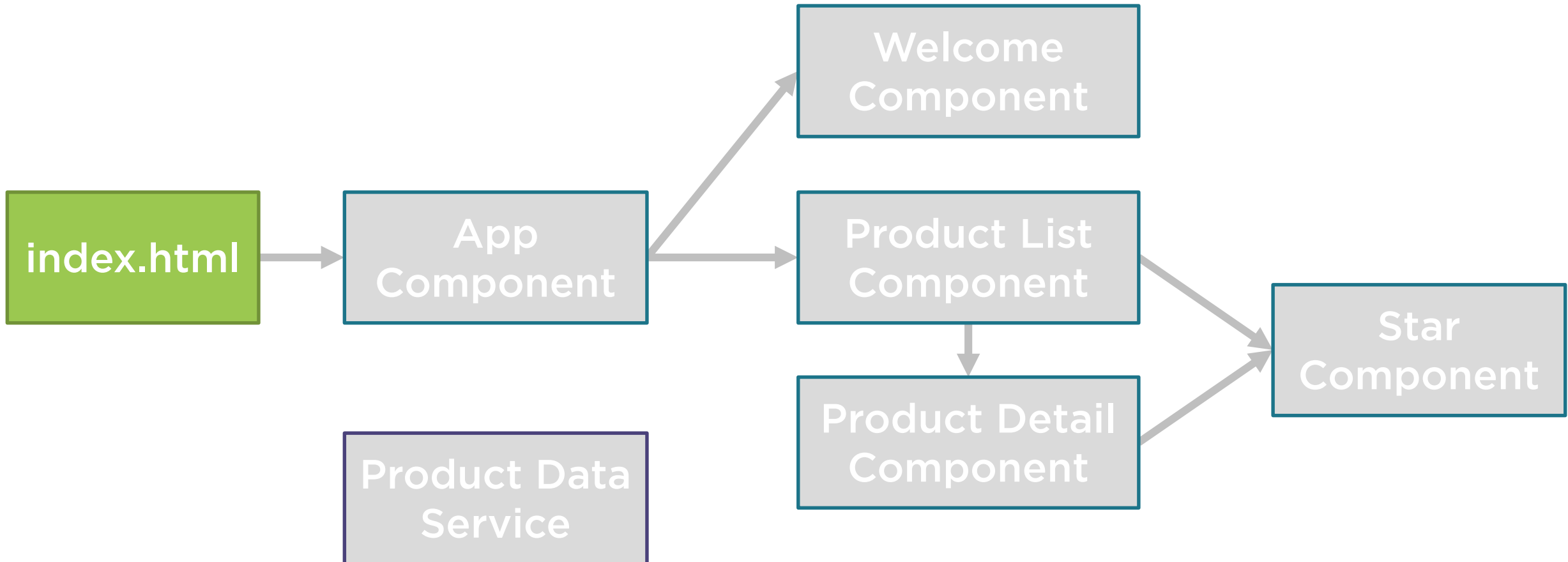
About Modules



Application Architecture



Application Architecture



Application Architecture

