STYLE GUIDE

**COMPONENTS**

* **Do** define one thing, such as a service or component, per file.
* **Consider** limiting files to 400 lines of code.

FUNCTIONS

* no more than 75 lines.

General Naming Guidelines

* **Do** use consistent names for all symbols.
* The recommended pattern is feature.type.ts.

File Names

* **Do** use dashes to separate words in the descriptive name.
* **Do** use dots to separate the descriptive name from the type.
* **Do** use upper camel case for class names.
* **Do** match the name of the symbol to the name of the file.
* **Do** append the symbol name with the conventional suffix (such as [Component](https://angular.io/api/core/Component),[Directive](https://angular.io/api/core/Directive), Module, [Pipe](https://angular.io/api/core/Pipe), or Service) for a thing of that type.
* **Do** give the filename the conventional suffix (such as .component.ts, .directive.ts,.module.ts, .pipe.ts, or .service.ts) for a file of that type.

Directives

* **Do** Use lower camel case for naming the selectors of directives (like ngFor)

Classes

* Upper Camel Case for clases

Constants

* **Do** declare variables with const if their values should not change during the application lifetime.
* **Consider** spelling const variables in lower camel case.

Interfaces:

* **Do** name an interface using upper camel case.
* **Consider** naming an interface without an I prefix.
* **Consider** using a class instead of an interface.

### Properties and methods

* **Do** use lower camel case to name properties and methods.
* **Avoid** prefixing private properties and methods with an underscore.