QUICKSTART

TYPESCRIPT

A quick look at Angular basics

Angular applications are made of *components*. A *component* is the combination of an HTML template and a component class that controls a portion of the screen. Here is an example of a component that displays a simple string:

app/app.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'my-app',
  template: `<h1>Hello {{name}}</h1>`
})
export class AppComponent { name = 'Angular'; }
```

You can try this out without installing anything. Open the QuickStart example on Plunkr in another tab and follow along.

Every component begins with a `@Component` *decorator* function that takes a *metadata* object. The metadata object describes how the HTML template and component class work together.

The `selector` property tells Angular to display the component inside a custom `<my-app>` tag in the `index.html` .

index.html (inside <body>)

```
<my-app>Loading AppComponent content here ...</my-app>
```

The `template` property defines a message inside an `<h1>` header. The message starts with "Hello" and ends with `{{name}}` which is an Angular interpolation binding expression. At runtime, Angular replaces `{{name}}` with the value of the component's `name` property.

In the example, change the component class's `name` property from `'Angular'` to `'World'` and see what happens.

Interpolation binding is one of many Angular features you'll discover in this documentation.

A WORD ABOUT TYPESCRIPT

This example is written in TypeScript, a superset of JavaScript. Angular uses TypeScript because its types make it easy to support developer productivity with tooling. You can also write Angular code in JavaScript; this guide explains how.

Next step

To learn how to write a real application, your next step is to set up a local development environment and begin exploring with code. The **Developer Guide** shows you how.