Understanding Directives

In this lecture, we are going to start learning about Directives.

You can think of them as **Custom Attributes for transforming content**.

Browsers have various attributes for affecting the behaviour of an element. For example, the target attribute on an anchor element can change how a link is opened in the browser.

Browsers defined dozens of attributes to alter an elements behaviour. In some cases, we may want to create **custom attributes instead of creating an entire components.** Angular gives us the power through a feature called **Directives**.

**We can apply Directives to native HTML elements or Custom components.**

**They’re more flexible than regular attributes.**

I provide a link to an official list of directives defined by Angular: <https://v17.angular.io/api?type=directive>

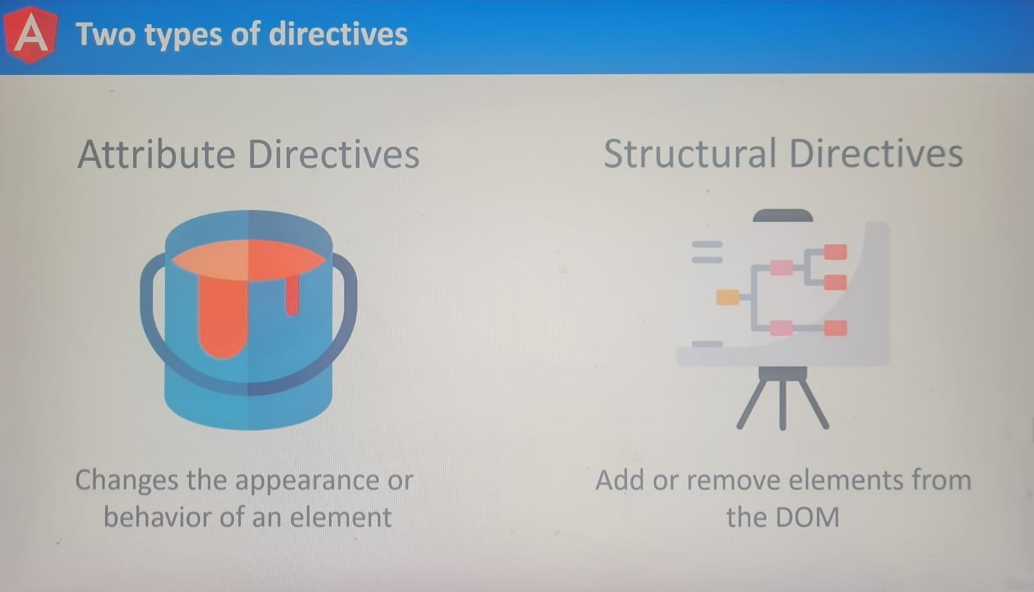
<https://angular.dev/api?type=directive>

We are going to put our focus on the **directives** defined under the **common package.**

The **other directives** will be **explored in other sections of the course**.

As we discussed before, the **common package** is **imported through the BrowserModule**.

We don’t have to take additional steps to start using these directives.



There are two types of **Directives**:

1. **Attribute Directives**
2. **Structural Directives**

Both types of directives have different syntax rules and behaviours.

**Attribute Directives – focus on changing the appearance or behaviour of an element.**

For example, if we want to dynamically add styles to an element, this type of directive would be considered an attribute directive.

**Structural Directives – can add or remove elements in the document. They’re focused on changing the layout of the DOM.**

If this sees confusing, that’s perfectly all right.