

# Create new Directives

In addition to all the built-in AngularJS directives, you can create your own directives.

New directives are created by using the `.directive` function.

To invoke the new directive, make an HTML element with the same tag name as the new directive.

When naming a directive, you must use a camel case name, `w3TestDirective`, but when invoking it, you must use - separated name, `w3-test-directive`:

## Example

```
<body ng-app="myApp">

<w3-test-directive></w3-test-directive>

<script>
var app = angular.module("myApp", []);
app.directive("w3TestDirective", function() {
    return {
        template : "<h1>Made by a directive!</h1>"
    };
});
</script>

</body>
```

You can invoke a directive by using:

- Element name
- Attribute
- Class
- Comment

The examples below will all produce the same result:

Element name

```
<w3-test-directive></w3-test-directive>
```

Attribute

```
<div w3-test-directive></div>
```

Class

```
<div class="w3-test-directive"></div>
```

Comment

```
<!-- directive: w3-test-directive -->
```

# Restrictions

You can restrict your directives to only be invoked by some of the methods.

## Example

By adding a `restrict` property with the value `"A"`, the directive can only be invoked by attributes:

```
var app = angular.module("myApp", []);
app.directive("w3TestDirective", function() {
    return {
        restrict : "A",
        template : "<h1>Made by a directive!</h1>"
    };
});
```

The legal restrict values are:

- `E` for Element name
- `A` for Attribute
- `C` for Class
- `M` for Comment

By default the value is `EA`, meaning that both Element names and attribute names can invoke the directive.