



# **DIRECTIVES IN ANGULARIS**

- AngularJS introduces some extended new attributes in HTML called Directives.
- AngularJS enhance your application by providing extra functionality by builtin directives.
- AngularJS supports us to define our **own Custom directives**.
- The new custom directives will be created by **.directive** function.
- We can invoke directives by using:
  - Invoke as Element Name.
  - Invoke as Attribute Name.
  - Invoke as Class Name.
  - Invoke as Comment.
- We should define the directive name in **camelCase** (directiveName) and case sensitive, we should invoke the directive in **dash-delimited** (directive-name) format.

#### **Matching Directives in AngularJS:**

- Before create an AngularJS directive we should know when to use the directive and how to use the directive and Angular compiler.
- HTML Element matches the AngularJS directives by defining the directives in the part of the HTML element.
- The <input> tag matches the ng-model directive by





# AngularJS Step By Step Tutorials



#### <input ng-model="name">

The <input> tag matches the ng-model directive by

The <my-own-directive> element matches the my-own-directive directive:

<my-own-directive > </ my-own-directive>

# **Matching Directives in AngularJS:**

- The AngularJS matches which element or attributes matched to which directive.
- AngularJS has case-sensitive and camelCase name for their directives.
- The normalization process is as follows:
  - o Remove the front part of the element or attributes if x- or datapresents.
  - o Convert the delimited (-, :, \_) name to camelCase directive (Example: directive-name as directiveName).









# **Syntax for Create Custom Directive in AngularJS:**

```
var app = angular.module("myApp", []);
app.directive("myOwnDirective", function() {
   return {
     template: "This is my Own Directive"
  };
});
```

# **Syntax for Invoke Custom Directive in AngularJS:**

```
< my-own-directive > </ my-own-directive >
```







# **Sample code for Create Custom directive in AngularJS:**

```
<!DOCTYPE html>
<html>
   <head>
      <title>Wikitechy AngularJS Tutorials</title>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/</pre>
1.4.8/angular.min.js"> </script>
   </head>
   <body>
      <h1> Custom Directives in AngularJS </h1>
      <div ng-app="myApp">
         <my-own-directive></my-own-directive>
      </div>
      <script>
        var app = angular.module("myApp", []);
         app.directive("myOwnDirective", function() {
            return {
              template: "This is my Own Directive"
           };
        });
      </script>
   </body>
</html>
```

#### Data:

• Set of data has been used in our AngularJS Application.

```
template: "This is my Own Directive"
```









#### HTML:

Viewable HTML contents in AngularJS Application.

```
<div ng-app="myApp">
  <my-own-directive> </my-own-directive>
</div>
```

# Logic:

Logic for declaring a Custom Directive in AngularJS application.

```
var app = angular.module("myApp", []);
app.directive("myOwnDirective", function() {
   return {
      template: "This is my Own Directive"
  };
});
```







### **Code Explanation for Create Custom Directive in AngularJS:**

```
<!DOCTYPE html>
<html>
 <html >
    <head>
      <title>Wikitechy Agular JS Tutorials</title>
      <script src="https://ajax.googleapis.</pre>
        com/ajax/libs/angularjs/1.5.6/angular.min.js"></script>
    </head>
    <body>
      <h1>Custom Directives in AngularJS</h1>
      <div ng-app="myApp">
        <my-own-directive></my-own-directive> ----- (1)
      </div>
      <script>
        var app = angular.module("myApp", []);
     app.directive("myOwnDirective", function() {
          return {
     Template : "This is my Own Directive"
          };
        });
      </script>
    </body>
  </html>
```

- 1. Invoke custom directive by Element Name as <my-own-directive> (dashdelimited).
- 2. Create a Custom directive by using .directive function as myOwnDirective (camelCase).
- 3. The **template** to bind the Element content by the custom directive.

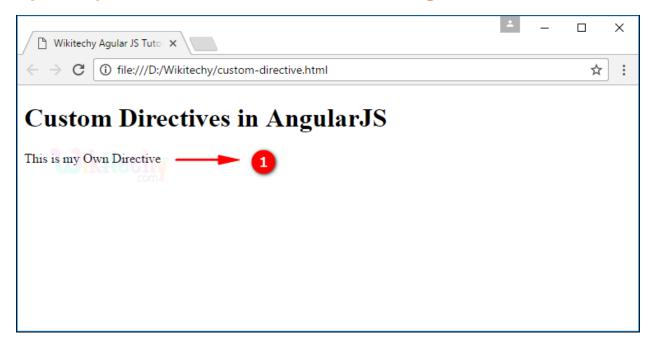








### **Sample Output for Create Custom Directive in AngularJS:**



1. The output shows custom directive template content displayed.

