Directives In Angularjs

What are directives?

1. Directives are reusable components.
2. We can teach the new syntax to the html

Ex: <custom-name></custom-name>

1. They will give you a common functionality on the html
2. To achieve the validations
3. To build the plugins or widgets.

**Building the directives.**

1. Create a separate module with the name as ‘components’



1. Create a directive with the name customName.
2. Directives will always return an object.
3. If the directive name is in camel case “customName” in the html it will be represented as

custom-name.

customName🡺custom-name

ngClick 🡺ng-click

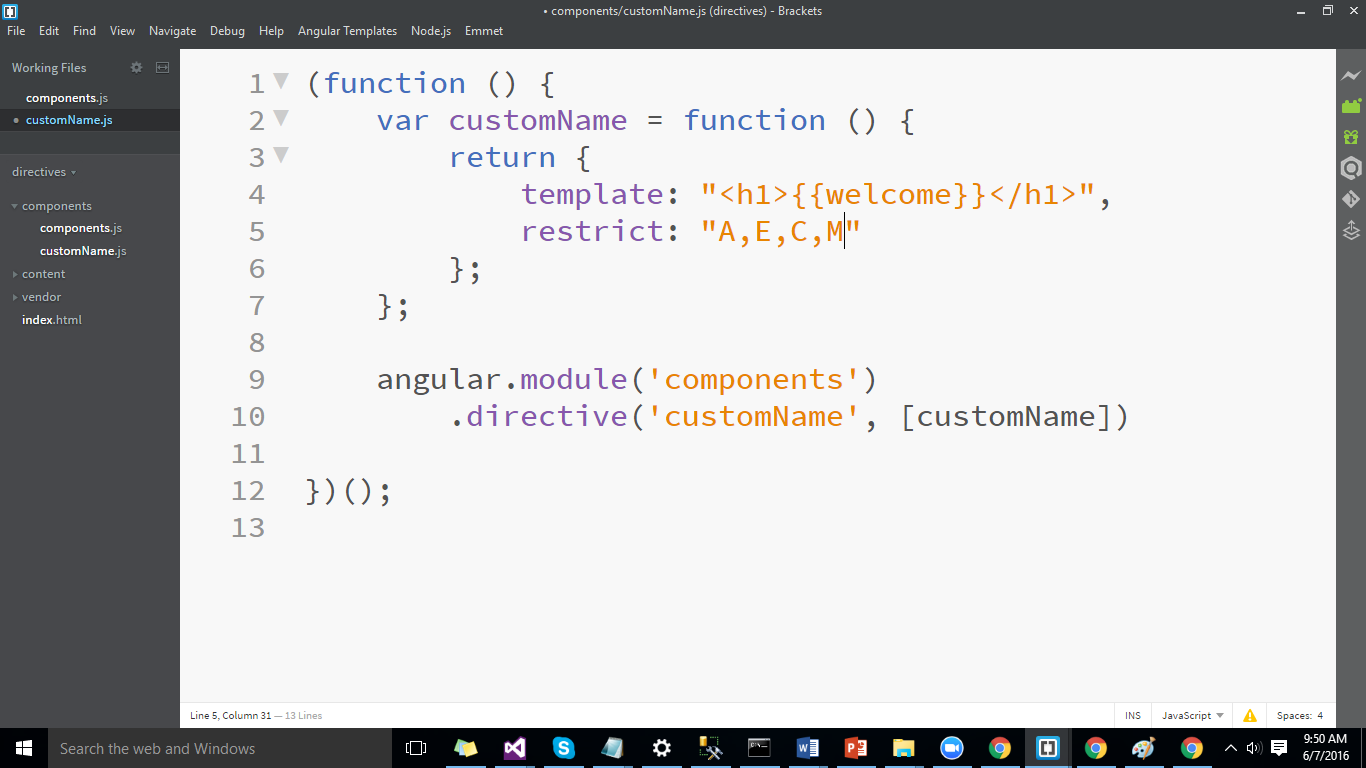
uiView🡺ui-view

ngModel🡺ng-model

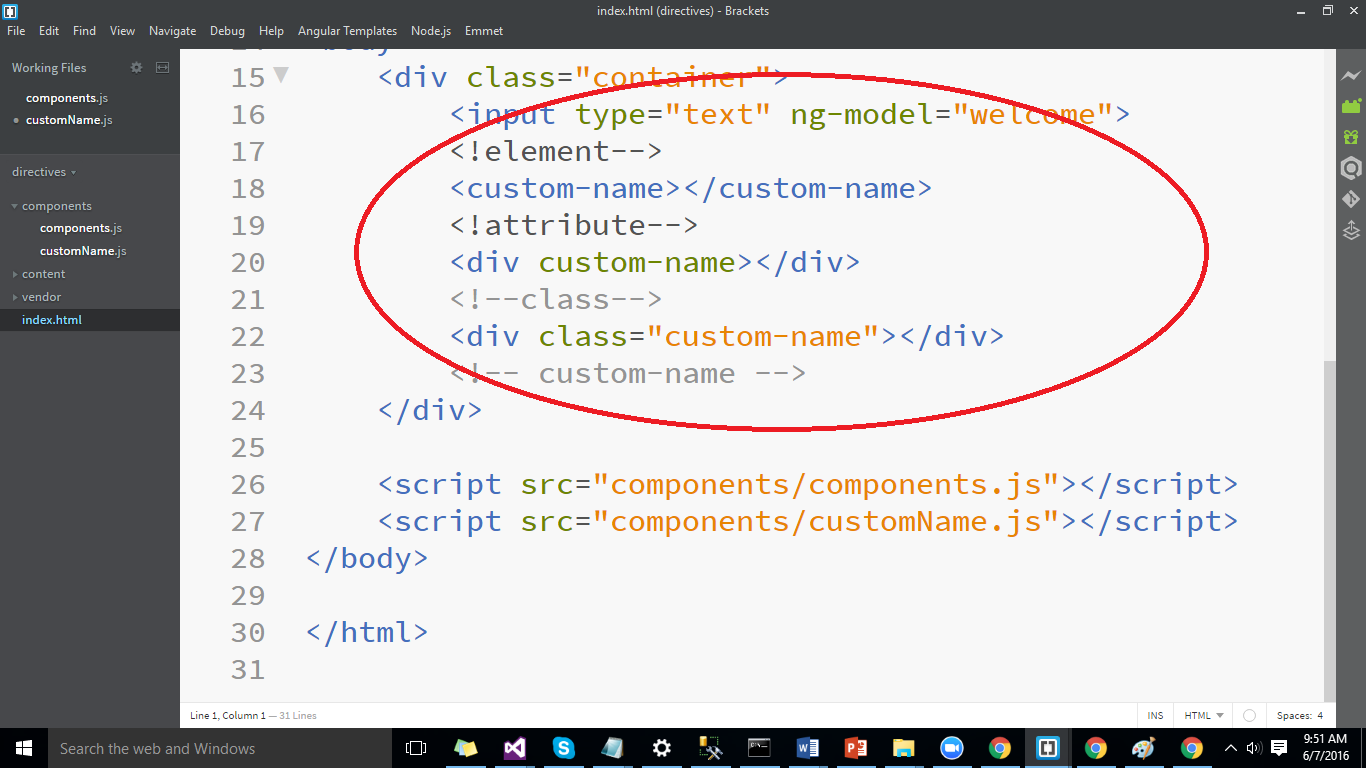
CUSTOMNAME🡺c-u-s-t-o-m-n-a-m-e

Properties of the Directives

1. Template🡺 purpose of the template is to the render the html string. It can have models, expressoins.
2. Complete Directive can be seen below.



In the html the directive will be represented as below.



Link Function:

1. To bind the events
2. To read the attributes and act according to the attributes we will use the link function.
3. To convert existing jquery plugins into angular.
4. It will take three parameters

Scope 🡪 parent controller

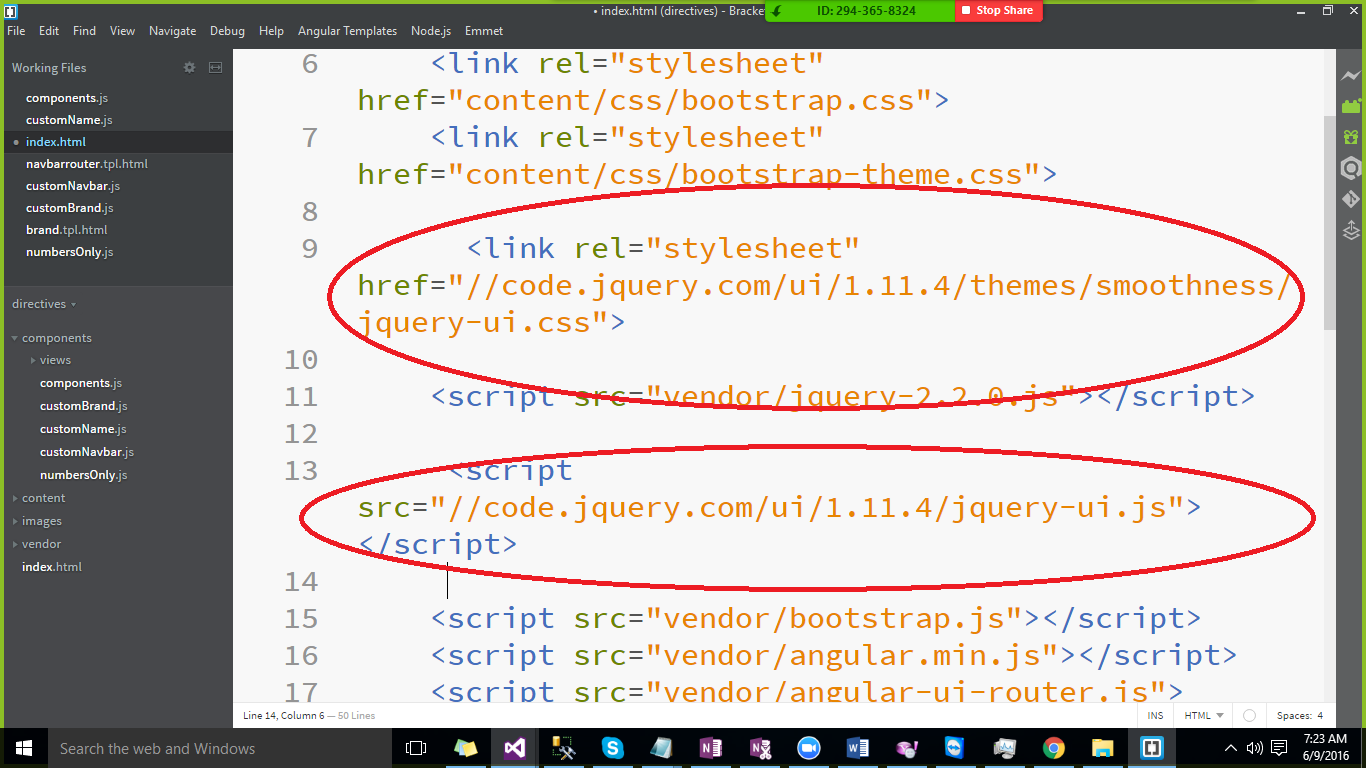
Element 🡪 the element on which directive is declared.

Attrs: attributes that are declared on the element.

Steps to convert Existing plugin into a directive:

Convert the jquery date picker to a directive.

1. Refer the required libraries.



1. Build the directive now.

Isolate scope in the directives

1. Isolated scope is an independent scope.
2. Scope lives only with the directives
3. Outside the scope cannot be accessible
4. If it wants to access the outer scope
5. Oneway @
6. Two way =

