

input[range] DIRECTIVE IN ANGULARJS

- **input [range]** is one of the AngularJS input directive in **module ng**.
- AngularJS directive **input [range]** is used to create an HTML range input with validation and transformation.
- The model for the range input should be a **Number**.

Some following values for Angular:

- As the element value must reflect the current model value, a range input will set the ngModel expression to the value that the browser to be set for the input element.
 - For ex, `<input type="range" ng-model="model.value">`
- If sets the value for `model.value=null`, the browser will set the input to '50' as well as angular will set the model to 50.
- It means the model for range will directly set to 50 after ngmodel has been initialized.
- It does not affect the changes to the model value.
- This directive executes at priority level 0.

Syntax for input [range] directive in AngularJS:

```
<input type="range"  
      ng-model="string"  
      [name="string"]  
      [min="string"]  
      [max="string"]  
      [step="string"]  
      [ng-change = "string"] >  
      [ng-checked = "expression"] >
```

Parameter values:

Parameters	Type	Description
ngModel	string	Defines angular expression to data-bind to.
name (optional)	string	Name of the form under which the control is available
min(optional)	string	To set the min validation to ensure that the entered is greater than min.



max(optional)	string	To set the max validation to ensure that the value entered is shorter than max.
step(optional)	string	To set the step validation to ensure that the value entered is matches the step.
ngChange (optional)	string	An expression of Angular to be executed when input changes due to user interaction with the input element.
ngChecked (optional)	expression	Checked attribute will be set on the element when the expression is truthy.

Sample coding for input [range] directive in AngularJS

```
<!DOCTYPE html>
<html>
  <head>
    <title>Wikitechy AngularJS Tutorial</title>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/angular.min.js">
    </script>
  </head>
  <body>
    <form ng-app="myApp" name="Form" ng-controller="rangeCtrl">
      <h3>input[range] directive example in AngularJS</h3>
      <h4>Range input with html min and max Attribute</h4>
      Model as range: <input type="range" name="range" ng-model="value"
min="{{min}}" max="{{max}}"> <br/>

      <h4>Range input with ng-min and ng-max Attribute</h4>
      Model as range: <input type="range" name="range" ng-model="value"
ng-min="min" ng-max="max"> <br/> <br/>
```



```
Model as number: <input type="number" ng-  
model="value"><br/><br/>  
Min: <input type="number" ng-model="min"> <br/><br/>  
Max: <input type="number" ng-model="max"> <br/>  
<p> value = {{value}}</p>  
<p>Form.range.$valid = {{Form.range.$valid}}</p>  
<p> Form.range.$error = {{Form.range.$error}}</p>  
</form>  
<script>  
var app=angular.module('myApp', []);  
app.controller('rangeCtrl', function($scope) {  
    $scope.value = 75;  
    $scope.min = 5;  
    $scope.max = 100;  
});  
</script>  
</body>  
</html>
```

Code Explanation input [range] directive in AngularJS:



```
<!DOCTYPE html>
<html>
  <head>
    <title>Wikitechy AngularJS Tutorial</title>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/
      1.5.6/angular.min.js"></script>
  </head>
  <body>
    <form ng-app="myApp" name="Form" ng-controller="rangeCtrl">
      <h3>input[range] directive example in AngularJS</h3>
      <h4>Range input with html min and max Attribute</h4>
      Model as range: <input type="range" name="range" ng-model="value"
        min="{{min}}" max="{{max}}" ><br/>

      <h4>Range input with ng-min and ng-max Attribute</h4>
      Model as range: <input type="range" name="range" ng-model="value"
        ng-min="min" ng-max="max" ><br/><br/>

      Model as number: <input type="number" ng-model="value"><br/><br/>
      Min: <input type="number" ng-model="min"><br/><br/>

      Max: <input type="number" ng-model="max"><br/>
      <p> value = {{value}}</p>
      <p>Form.range.$valid = {{Form.range.$valid}}</p>
      <p> Form.range.$error = {{Form.range.$error}}</p>
    </form>
    <script>
      var app=angular.module('myApp', []);
      app.controller('rangeCtrl', function($scope) {
        $scope.value = 75;
        $scope.min = 5;
        $scope.max = 100;
      });
    </script>
  </body>
</html>
```

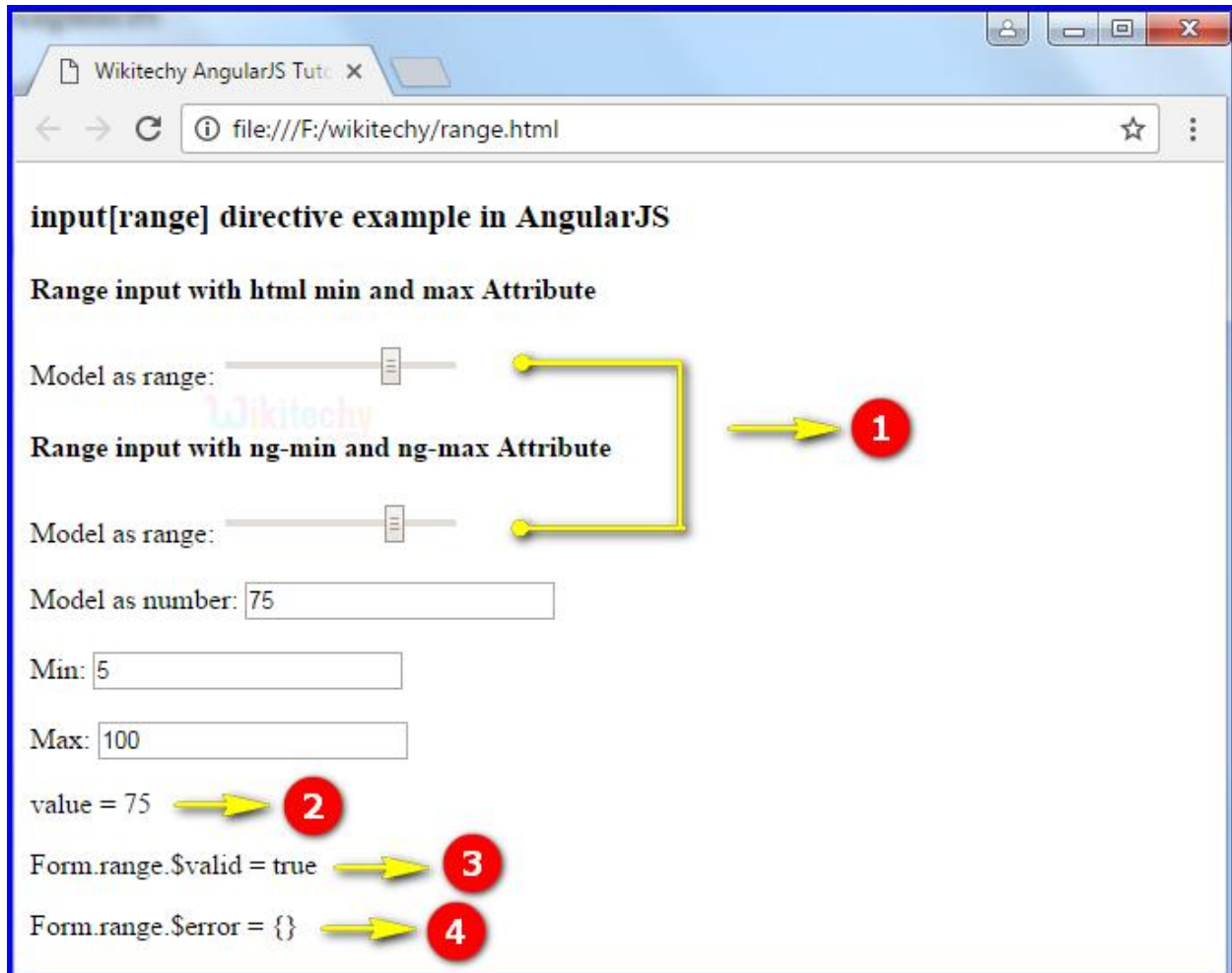
1. The **ng-app** specifies the root element ("**myApp**") to define AngularJS application.
2. ng-controller specifies the application controller in AngularJS the controller value is given as "**rangeCtrl**".



3. **"range"** is declare the **type** value of the <input> tag.
4. **min** parameter is used to declare the min value in expression.({{min}})
5. **max** parameter is used to declare the max value in expression.({{max}})
6. **"range"** is declare the **type** value of the <input> tag.
7. ng-min is used to declare the min value in AngularJS.("min")
8. ng-max is used to declare the max value in AngularJS("max")
9. **"number"** is declare the **type** value of the <input> tag.
- 10.The **ng-model** bind an input field value to AngularJS application variable ("min").
- 11.The **ng-model** bind an input field value to AngularJS application variable ("max").
- 12.**Form.range.\$valid** to checks the correct range format or not. If the user set the range in the input field then the output will be displays as true otherwise false.
- 13.**Form.range.\$error** to check whether the valid range format or not .If the range specified in error it throw an exception (Like "required"="true) otherwise it is an empty curly braces ({ }).
- 14.The values are given as value, min and max in the scope object.



Sample Output input [range] directive in AngularJS:



input[range] directive example in AngularJS

Range input with html min and max Attribute

Model as range:

Range input with ng-min and ng-max Attribute

Model as range:

Model as number:

Min:

Max:

value = 75

Form.range.\$valid = true

Form.range.\$error = {}

1. If the user set the range in the input field.
2. The output displays as **value=75**.
3. The output displays true because it is consider as a valid range value.
4. The output displays empty curly braces it means does not thrown any error.

Browser Support for input [range] directive in AngularJS:

Browser	Support
Chrome	Yes
Internet explorer (9.0)	No
edge	Yes
Firefox	Yes
Safari	Yes

Tips and Notes

- input [range] does not set the min and max attributes, because it is not compatible with ngMax, ngMin and ngStep.
- It means the browser will not automatically adjust the input value , and will always assume min=0, max=100 and step=1

