

MODULES IN ANGULARJS

- An AngularJS supports modular approach.
- The module is a container for the different parts like [services](#), [controllers](#), [filters](#), etc., of an application.
- The [angular.module](#) function is used to create a module in AngularJS.
- The [ng-controller](#) directive is used to add a controller to a module.
- AngularJS allows the user can use the module to add their own directives to their applications.
- AngularJS has a set of built-in [directives](#) which you can use to add functionality to your application.
- Global function should be avoided in JavaScript. Because it can be easily overwriting or destroyed by other scripts.
- User can load the AngularJS library JS file either in the [<head>](#) or at the start of the [<body>](#).

Syntax for creating a Module in AngularJS

```
var variablename= angular.module (moduleName, [ ]);
```



Sample coding for Module in AngularJS:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Wikitechy AngularJS Tutorials</title>
  </head>
  <body>
    <h2>Wikitechy Modules in AngularJS</h2>
    <script
      src="https://ajax.googleapis.com/ajax/libs/
      angularjs/1.4.8/angular.min.js" >
    </script>
    <div ng-app="myApp" ng-controller="moduleCtrl" >
      {{firstWord + " " + secondWord+ " " +thirdWord}}
    </div>
    <script>
      var app = angular.module ("myApp", [ ]);
      app.controller ("moduleCtrl", function($scope) {
        $scope.firstWord = "Welcome";
        $scope.secondWord = "To";
        $scope.thirdWord = "AngularJS Tutorials";
      });
    </script>
  </body>
</html>
```



Data:

- Collection of data has been defined using firstWord, secondWord and thirdWord for our AngularJS Application.

```
firstWord = "Welcome";  
secondWord = "To";  
thirdWord = "AngularJS Tutorials";
```

Logic:

- Controller logic for the AngularJS application.

```
apps.controller ("moduleCtrl", function($scope)  
{  
  $scope.firstWord = "Welcome";  
  $scope.secondWord = "To";  
  $scope.thirdWord = "AngularJS Tutorials";  
}
```

HTML:

- Viewable HTML contents in AngularJS Application.

```
<div ng-app="myApp" ng-controller="moduleCtrl">  
    {{firstWord + " " + secondWord + " " + thirdWord}}  
</div>
```

Code Explanation for Modules in AngularJS:

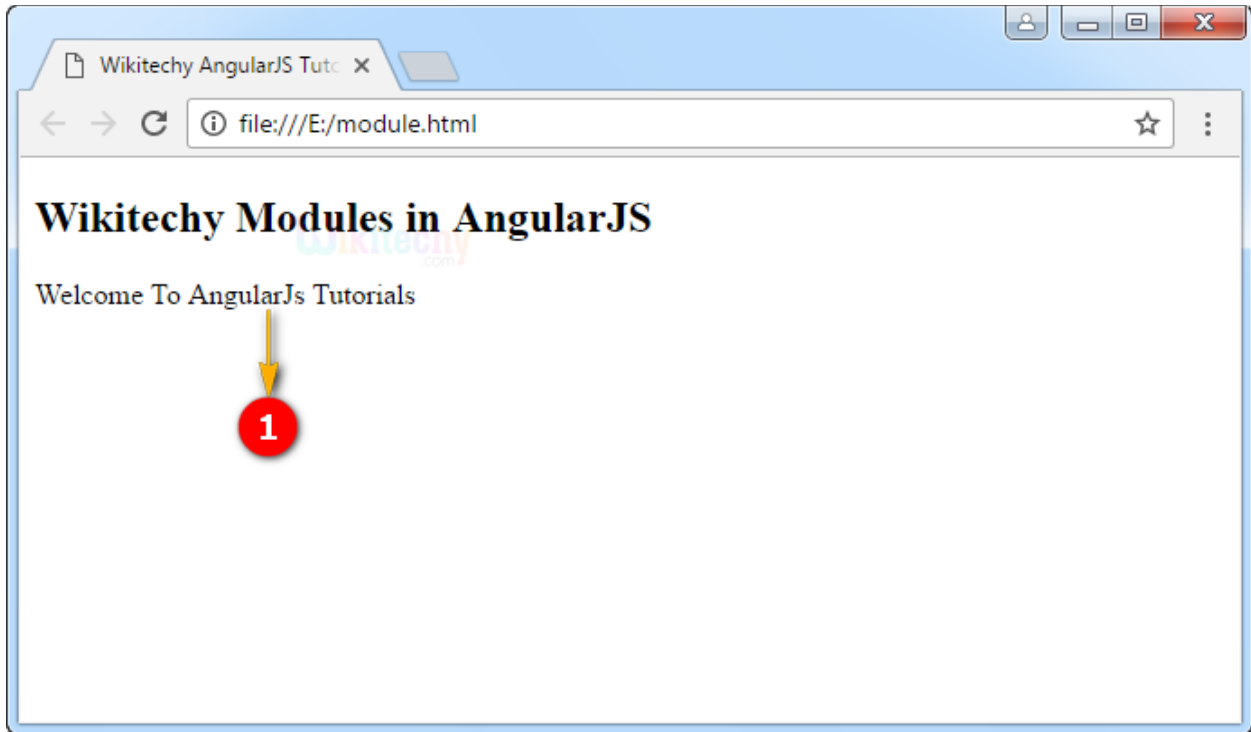
```
<!DOCTYPE html>
<html>
  <head>
    <title> Wikitechy AngularJS Tutorials</title>
  </head>
  <body>
    <h2>Wikitechy Modules in AngularJS</h2>
    <script
      1 ← src="https://ajax.googleapis.com/ajax/libs/
        angularjs/1.4.8/angular.min.js"
      2 ← </script>
    <div ng-app="myApp" ng-controller="moduleCtrl">
      4 ← {{ firstWord + " " + secondWord+ " "+thirdWord }}
    </div>
    <script>
      5 ← var app = angular.module("myApp", []);
      6 ← app.controller("moduleCtrl", function($scope)
      7 ← {
      8 ←   $scope.firstWord = "Welcome";
      $scope.secondWord = "To";
      $scope.thirdWord = "AngularJs Tutorials";
      });
    </script>
  </body>
</html>
```

1. AngularJS is distributed as a JavaScript file, and can be added to a HTML page with a [<script>](#) tag.
2. The AngularJS application is defined by `ng-app="myApp"`. The application runs inside the [<div>](#) tag. It's also used to define a [<div>](#) tag as a root element.

3. The [ng-controller](#)="**moduleCtrl**" is an AngularJS directive. It is used to assign a controller name as "**moduleCtrl**".
4. **{{firstWord+" "+secondWord+" "+thirdWord}}** is a string concatenation code of the **myApp** application for AngularJS. The **{{ }}** is used to bind the value of the application variable.
5. Here we create an module by using **angular.module** function. We have passed an empty array to it.
6. Here we have declared a controller **moduleCtrl** module using **apps.controller()** function.
7. The value of the controller modules is stored in scope object. In AngularJS, **\$scope** is passed as first argument to **apps.controller** during its constructor definition.
8. Here we have set the value of **\$scope.firstWord**, **\$scope.secondWord** and **\$scope.thirdWord** which are to be used to display the **{{firstWord+" "+secondWord+" "+thirdWord}}** values in the HTML [<div>](#) element.



Sample Output for Modules in AngularJS :



1. The output shows a concatenation of a three string `{{firstWord+" "+secondWord+ " "+ thirdWord}}`.