# Scope IN ANGULARJS

- The **scope** is an object that refers to the application model.
- It is used to specified the binding part (expression) between the HTML and the controller.
- It plays a role of **linking** controller with the views.
- It is accessible for both the view and the controller.
- Scopes are organized in **hierarchical** structure which imitate the DOM structure of the applications.
- Scopes can watch binding(expression) and propagate events.
- Scopes can be nested to limit access to the properties of application component, while giving access to shared model properties.
- In AngularJS, the nested scopes are either **"child scope"** or **"isolate scope".**
- The **"child scope"** is inheriting properties from the base or parent scope.
- An **"isolate scope"** does not prototypically (**"child scope"**) inherit from its parent scope.

## Syntax for scope in Angular JS:

```
myApp.controller('myController', function($scope) {
$scope.myVariable = "value";
});
```

## Sample coding for scope in AngularJS:

```
<!DOCTYPE html>
<html>
	<head>
		<title>Wikitechy AngularJS Tutorials</title>
	</head>
		<script src="https://ajax.googleapis.com/ajax/libs/
		angularjs/1.5.6/angular.min.js"></script>
	<body>
		<h2> Wikitechy scope in AngularJS</h2>
		<div ng-app="myApp" ng-controller="scopeCtrl">
				{{ msg }}
		</div>
		<script>
			var app= angular.module('myApp', [ ]);
			app.controller('scopeCtrl', function($scope) {
			$scope.msg = "Welcome To AngularJS Tutorials";
			});
		</script>
	</body>
</html>
```

**Data:**

- The **msg** data been defined for our AngularJS Application.

```
msg = "Welcome To AngularJS Tutorials";
```

**Logic:**

- Controller logic for the AngularJS application

```
app.controller('scopeCtrl', function($scope)
    {
      $scope.msg = "Welcome To AngularJS Tutorials";
    });
```

**HTML:**

- Viewable HTML contents in AngularJS Application.

```
<div ng-app="myApp" ng-controller="scopeCtrl">
                        {{ msg }}
</div>
```

## Code Explanation for scope in AngularJS:

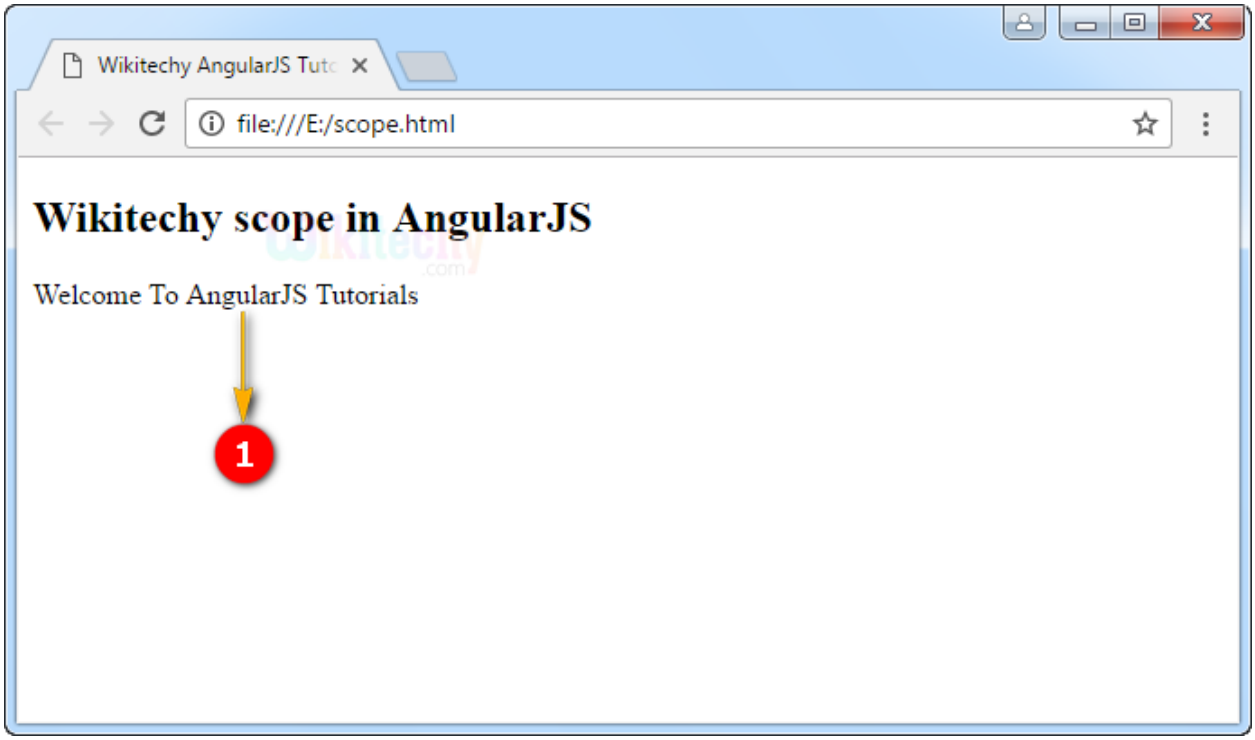
```
<!DOCTYPE html>
<html>
    <head>
        <title>Wikitechy AngularJS Tutorials</title>
    </head>
        <script
 1 ←src="https://ajax.googleapis.com/ajax/libs/
        angularjs/1.5.6/angular.min.js">
        </script>
    <body>
        <h2> Wikitechy scope in AngularJS</h2>
                2                    3
        <div ng-app="myApp" ng-controller="scopeCtrl">
 4 ←{{msg}}                 5                  7
        </div>
        <script>
            var app = angular.module('myApp', []);
 6 ←app.controller('scopeCtrl', function($scope) {
                $scope.msg = "Welcome To AngularJS Tutorials";
            });
        </script>
    </body>                 8
</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 ** ** tag. It's also used to define a ** ** tag as a root element.
3. The **ng-controller="scopeCtrl"** is an AngularJS directive. It is used to define a controller name as **"scopeCtrl".**
4. **{{ msg }}** is a used to bind the controller value by using the scope object.

5. **angular.module** function is used to create a module. Here has passed an empty array to it.
6. Here we have declared a controller **scopeCtrl** 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.msg** as **"Welcome To AngularJS Tutorials"** which are to be used to display the **{{ msg }}** value in the HTML** ** element.

## Sample Output for scope in AngularJS :




1. The output displays a message of a string **{{ msg }}** by using a $scope object