

ng-class-odd DIRECTIVE IN ANGULARJS

- **ng-class-odd** directive is used to dynamically set one or more CSS classes to an HTML element, but will take effect only on every odd appearance of the HTML element.
- The **ng-class-odd** directive will only work, when the **ng-repeat** directive is existing in the same HTML element.
- All HTML elements supported the **ng-class-odd** directive. But the **ng-class-odd** directive is perfect for styling items in **list** or rows in a **table**.

Syntax for ng-class-odd directive in AngularJS:

```
<element ng-repeat="x in data" ng-class-odd = "expression" > </element>
```

Parameter value for ng-class-odd directive in AngularJS:

Value	Description
expression	It is used to define an expression that returns one or more class names.

Sample coding for ng-class-odd directive in AngularJS:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Wikitechy AngularJS Tutorials</title>
    <script src="https://ajax.googleapis.com/ajax/libs/
      angularjs/1.5.6/angular.min.js">
    </script>
    <style>
      .striped {
        background-color:yellow;
        width:100px;
      }
    </style>
  </head>
  <body ng-app="myApp">
    <h2>Wikitechy ng-class-odd directive in AngularJs</h2>
    <ul ng-controller="ngclassoddCtrl">
      <li ng-repeat="x in technology" ng-class-odd=" 'striped' ">
        {{x.tut}} </li>
    </ul>
    <script>
      var app = angular.module( "myApp" , []);
      app.controller("ngclassoddCtrl", function($scope) {
        $scope.technology = [
          {"tut":"HTML"},
          {"tut":"AngularJS"},
          {"tut":"CSS"},
          {"tut":"C"},
          {"tut":"JAVA"},
          {"tut":"PHP"}, ];
      });
    </script>
  </body>
</html>
```

Code Explanation for ng-class-odd directive in AngularJS:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Wikitechy AngularJS Tutorials</title>
    1 ← <script src="https://ajax.googleapis.com/ajax/libs/
        angularjs/1.5.6/angular.min.js">
    </script>
    <style>
      .striped {
        background-color:yellow;
        width:100px;
      }
    </style>
  </head>
  <body ng-app="myApp">
    <h2>Wikitechy ng-class-odd directive in AngularJs</h2>
    <ul ng-controller="ngclassoddCtrl">
      <li ng-repeat="x in technology" ng-class-odd="'striped'">
        6 ← {{x.tut}}
      </li>
    </ul>
    <script>
      8 ← var app = angular.module("myApp", []);
      app.controller("ngclassoddCtrl", function($scope) {
        $scope.technology = [
          {"tut":"HTML"},
          {"tut":"AngularJS"},
          {"tut":"CSS"},
          {"tut":"C"},
          {"tut":"JAVA"},
          {"tut":"PHP"},
        ];
      });
    </script>
  </body>
</html>
```

1. AngularJS is distributed as a JavaScript file, and can be added to a HTML page with a `<script>` tag.

2. `. striped {background-color: yellow; width:100 px}` is used to define a CSS class as **striped** which is used to set the `` element's even item background color as **yellow** with 100 pixels' width.
3. The AngularJS application is defined by `ng-app="myApp"`. The application runs inside the `<body>` tag. It's also used to define a `<body>` tag as a root element.
4. The `ng-controller="ngclassoddCtrl"` is an AngularJS directive. It is used to define a controller name as `"ngclassoddCtrl"`.
5. `<li ng-repeat="x in technology" ng-class-odd=" 'striped' ">`, here the `` elements defined `ng-repeat` and `ng-class-odd` directives. Because the `ng-class-odd` directive only work with `ng-repeat` directive. The `ng-class-odd` directive is used to dynamically set striped CSS classes to an `` element, but will take effect only on every odd appearance of the `` element.
6. The `{{x.tut}}` is used to bind the `tut` value from a `technology` array.
7. The `angular.module` function is used to create a Module.
8. Here we have declared a controller `ngclassoddCtrl` module using `apps.controller()` function. The value of the controller modules is stored in scope object. In AngularJS, `$scope` is passed as first argument to `apps.controller` during its constructor definition.
9. Here we have set the value of `$scope.technology=[{"tut":"HTML"}, {"tut":"AngularJS"}, {"tut":"CSS"}, {"tut":"C"}, {"tut":"JAVA"}, {"tut":"PHP"},];`];

Sample Output for ng-class-odd directive in AngularJS:



1. The output displays the technology array items. And every odd item in technology array is displayed with yellow color background by using the **ng-class-odd** directive.