Angular Material

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com

https://www.facebook.com/tutorialspointindia

https://twitter.com/tutorialspoint

About the Tutorial

Angular Material is a UI component library for Angular JS developers. Angular Material components help in constructing attractive, consistent, and functional web pages and web applications while adhering to modern web design principles like browser portability, device independence, and graceful degradation. It helps in creating faster, beautiful, and responsive websites. It is inspired by the Google Material Design.

Audience

This tutorial is meant for professionals who aspire to learn the basics of Angular Material and how to use it to create faster, beautiful, and responsive websites. This tutorial explains the fundamental concepts of Angular Material.

Prerequisites

Before proceeding with this tutorial, you should have a basic understanding of Angular JS, HTML, CSS, JavaScript, Document Object Model (DOM), and any text editor. In addition, it will help if you know how web-based applications work.

Execute Angular Material Online

For most of the examples given in this tutorial, you will find a **Try-it** option. Use this option to execute your Angular Material programs then and there and enjoy your learning.

Try the following example using the **Try-it** option available at the top right corner of the following sample code box –

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script type="text/javascript">
         angular.module('firstApplication', ['ngMaterial']);
      </script>
   </head>
   <body ng-app="firstApplication" ng-cloak>
      <md-toolbar class="md-warn">
         <div class="md-toolbar-tools">
            <h2 class="md-flex">HTML 5</h2>
         </div>
      </md-toolbar>
      <md-content flex layout-padding>
         <p>HTML5 is the next major revision of the HTML standard superseding HTML 
4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting 
content on the World Wide Web.</p>
         <p>HTML5 is a cooperation between the World Wide Web Consortium (W3C) and 
the Web Hypertext Application Technology Working Group (WHATWG).</p>
         <p>The new standard incorporates features like video playback and drag-and-
drop that have been previously dependent on third-party browser plug-ins such as 
Adobe Flash, Microsoft Silverlight, and Google Gears.</p>
      </md-content>
   </body>
</html>
```

Copyright & Disclaimer

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial i

Audience i

Prerequisites i

Execute Angular Material Online i

Copyright & Disclaimer ii

Table of Contents iii

1. ANGULAR MATERIAL – OVERVIEW 1

2. ANGULAR MATERIAL – ENVIRONMENT SETUP 2

How to Use Angular Material? 2

Local Installation 2

CDN Based Version 4

3. ANGULAR MATERIAL – AUTOCOMPLETE 6

Attributes 6

4. ANGULAR MATERIAL – BOTTOM SHEET 12

5. ANGULAR MATERIAL – CARDS 15

6. ANGULAR MATERIAL – WIDGETS 20

Angular Material – Buttons 23

Angular Material – CheckBoxes 29

Angular Material – Content 32

Angular Material – DatePicker 33

Angular Material – Dialogs 37

Angular Material – Divider 41

Angular Material – List 44

Angular Material – Menu 47

Angular Material - Menu Bar 50

Angular Material – Progress Bars 53

Angular Material – Radio Buttons 60

Angular Material – Selects 63

Angular Material – Fab Toolbars 66

Angular Material – Sliders 69

Angular Material – Tabs 72

Angular Material – Toolbars 76

Angular Material – Tooltips 80

Angular Material – Chips 83

Angular Material – Contact Chips 87

7. ANGULAR MATERIAL – LAYOUTS 92

Layout Directive 92

Flex Directive 95

8. ANGULAR MATERIAL – INPUTS 98

9. ANGULAR MATERIAL – ICONS 102

10. ANGULAR MATERIAL – GRIDS 107

Attributes 107

11. ANGULAR MATERIAL – SIDENAV 111

12. ANGULAR MATERIAL – FAB SPEED DIAL 114

Attributes 114

13. ANGULAR MATERIAL – SUBHEADERS 119

14. ANGULAR MATERIAL – SWIPE 122

15. ANGULAR MATERIAL – SWITCHES 125

16. ANGULAR MATERIAL – THEME 129

17. ANGULAR MATERIAL – TOASTS 133

18. ANGULAR MATERIAL – TYPOGRAPHY 135

19. ANGULAR MATERIAL – VIRTUAL REPEAT 138

Attributes **138**

md-virtual-repeat **138**

20. ANGULAR MATERIAL – WHITEFRAME 143

1. Angular Material – Overview

Angular Material is a UI component library for Angular JS developers. Angular Material's reusable UI components help in constructing attractive, consistent, and functional web pages and web applications while adhering to modern web design principles like browser portability, device independence, and graceful degradation.

Following are a few salient features of Angular Material:

- In-built responsive designing.
- Standard CSS with minimal footprint.
- Includes new versions of common user interface controls such as buttons, check boxes, and text fields which are adapted to follow Material Design concepts.
- Includes enhanced and specialized features like cards, toolbar, speed dial, side nav, swipe, and so on.
- Cross-browser, and can be used to create reusable web components.

Responsive Design

- Angular Material has in-built responsive designing so that the website created using Angular Material will redesign itself as per the device size.
- Angular Material classes are created in such a way that the website can fit any screen size.
- The websites created using Angular Material are fully compatible with PC, tablets, and mobile devices.

Extensible

- Angular Material is by design very minimal and flat.
- It is designed considering the fact that it is much easier to add new CSS rules than to overwrite existing CSS rules.
- It supports shadows and bold colors.
- The colors and shades remain uniform across various platforms and devices.

And most important of all, Angular Material is absolutely free to use.

2. Angular Material – Environment Setup

How to Use Angular Material?

There are two ways to use Angular Material:

- **Local Installation** - You can download the Angular Material libraries using npm, jspm, or bower on your local machine and include it in your HTML code.

- **CDN Based Version** - You can include the angular-material.min.css and angular-material js files into your HTML code directly from the Content Delivery Network (CDN).

Local Installation

Befor we use the following npm command, we require the NodeJS installation on our system. To get information about node JS, click here and open the NodeJS command line interface. We will use the following command to install Angular Material libraries.

```
npm install angular-material
```

The above command will generate the following output –

```
angular-animate@1.5.2 node_modules\angular-animate


angular-aria@1.5.2 node_modules\angular-aria


angular-messages@1.5.2 node_modules\angular-messages


angular@1.5.2 node_modules\angular


angular-material@1.0.6 node_modules\angular-material
```

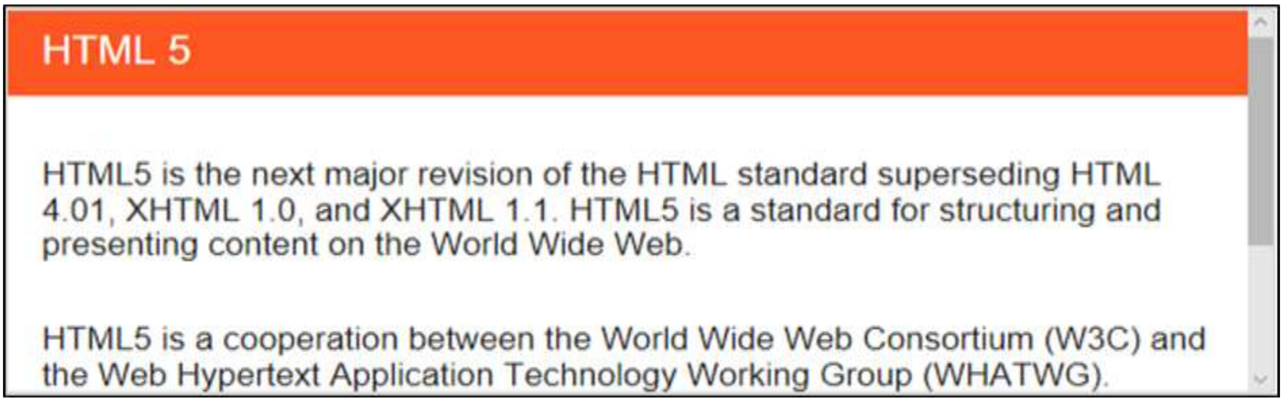
npm will download the files under **node_modules > angular-material** folder. Include the files as explained in the following example –

Example

Now you can include the *css* and *js* file in your HTML file as follows –

```
<html lang="en" >
   <head>
      <link rel="stylesheet" href="/node_modules/angular-material/angular-material.css">
      <script src="/node_modules/angular-material/angular.js"></script>
      <script src="/node_modules/angular-material/angular-animate.js"></script>
      <script src="/node_modules/angular-material/angular-aria.js"></script>
      <script src="/node_modules/angular-material/angular-messages.js"></script>
      <script src="/node_modules/angular-material/angular-material.js"></script>
      <script type="text/javascript">
         angular.module('firstApplication', ['ngMaterial']);
      </script>
   </head>
   <body ng-app="firstApplication" ng-cloak>
      <md-toolbar class="md-warn">
         <div class="md-toolbar-tools">
            <h2 class="md-flex">HTML 5</h2>
         </div>
      </md-toolbar>
      <md-content flex layout-padding>
         <p>HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.</p>
         <p>HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).</p>
         <p>The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.</p>
      </md-content>
   </body>
</html>
```

The above program will generate the following result –



HTML 5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

CDN Based Version

You can include the **angular-material.css** and **angular-material.js** files into your HTML code directly from the Content Delivery Network (CDN). Google CDN provides content for the latest version.

We are using the Google CDN version of the library throughout this tutorial.

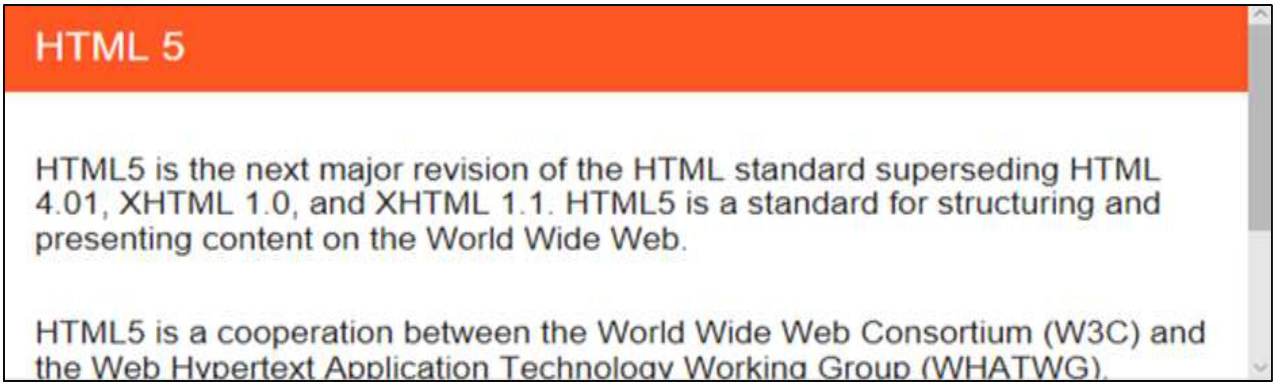
Example

Now let us rewrite the above example using **angular-material.min.css** and angular-**material.min.js** from Google CDN.

```
<html lang="en" >

   <head>

      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">

      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>

      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>

      <script type="text/javascript">

         angular.module('firstApplication', ['ngMaterial']);

      </script>

   </head>

   <body ng-app="firstApplication" ng-cloak>

      <md-toolbar class="md-warn">

         <div class="md-toolbar-tools">

            <h2 class="md-flex">HTML 5</h2>

         </div>

      </md-toolbar>

      <md-content flex layout-padding>

         <p>HTML5 is the next major revision of the HTML standard superseding HTML
4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting
content on the World Wide Web.</p>
```

```
         <p>HTML5 is a cooperation between the World Wide Web Consortium (W3C) and
the Web Hypertext Application Technology Working Group (WHATWG).</p>

         <p>The new standard incorporates features like video playback and drag-and-
drop that have been previously dependent on third-party browser plug-ins such as
Adobe Flash, Microsoft Silverlight, and Google Gears.</p>

      </md-content>

   </body>

</html>
```

The above program will generate the following result –



HTML 5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

3. Angular Material – Autocomplete

The **md-autocomplete**, an Angular Directive, is used as a special input control with an inbuilt dropdown to show all possible matches to a custom query. This control acts as a real-time suggestion box as soon as the user types in the input area. **<md-autocomplete>** can be used to provide search results from local or remote data sources. md-autocomplete caches results when performing a query. After first call, it uses the cached results to eliminate unnecessary server requests or lookup logic and it can be disabled.

Attributes

The following table lists out the parameters and description of the different attributes of **md-autocomplete**.

S.No.	Parameter & Description
1	*** md-items** An expression in the format of item in items to iterate over matches for your search.
2	**md-selected-item-change** An expression to be run each time a new item is selected.
3	**md-search-text-change** An expression to be run each time the search text updates.
4	**md-search-text** A model to bind the search query text to.
5	**md-selected-item** A model to bind the selected item to.
6	**md-item-text** An expression that will convert your object to a single string.
7	**placeholder** Placeholder text that will be forwarded to the input.
8	**md-no-cache** Disables the internal caching that happens in autocomplete.

S.No.	Parameter & Description
9	**ng-disabled** Determines whether or not to disable the input field.
10	**md-min-length** Specifies the minimum length of text before autocomplete will make suggestions.
11	**md-delay** Specifies the amount of time (in milliseconds) to wait before looking for results.
12	**md-autofocus** If true, will immediately focus the input element.
13	**md-autoselect** If true, the first item will be selected by default.
14	**md-menu-class** This will be applied to the dropdown menu for styling.
15	**md-floating-label** This will add a floating label to autocomplete and wrap it in the md-input-container.
16	**md-input-name** The name attribute given to the input element to be used with the FormController.
17	**md-input-id** An ID to be added to the input element.
18	**md-input-minlength** The minimum length for the input's value for validation.
19	**md-input-maxlength** The maximum length for the input's value for validation.
20	**md-select-on-match** When set as true, autocomplete will automatically select the exact item if the search text is an exact match.

Example

The following example shows the use of the **md-autocomplete** directive and also the use of autocomplete.

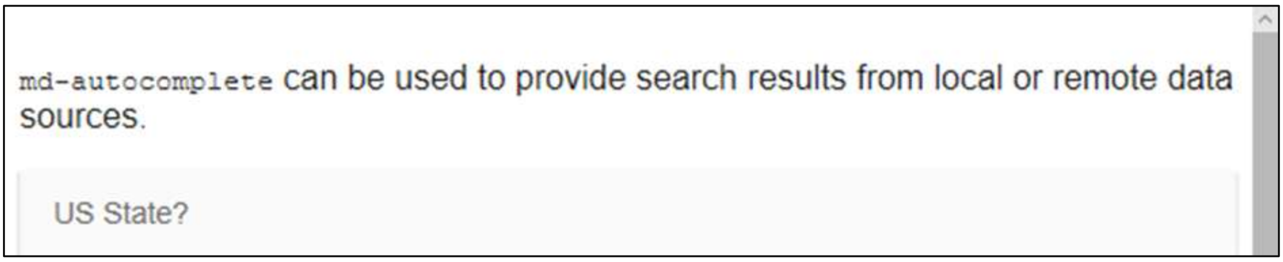
am_autocomplete.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <script language="javascript">
        angular
           .module('firstApplication', ['ngMaterial'])
           .controller('autoCompleteController', autoCompleteController);

        function autoCompleteController ($timeout, $q, $log) {
           var self = this;
           self.simulateQuery = false;
           self.isDisabled    = false;
           // list of states to be displayed
           self.states        = loadStates();
           self.querySearch   = querySearch;
           self.selectedItemChange = selectedItemChange;
           self.searchTextChange   = searchTextChange;
           self.newState = newState;
           function newState(state) {
              alert("This functionality is yet to be implemented!");
           }
           function querySearch (query) {
              var results = query ? self.states.filter( createFilterFor(query) ) :
self.states, deferred;
              if (self.simulateQuery) {
                 deferred = $q.defer();
                 $timeout(function () {
                       deferred.resolve( results );
                    },
                        Math.random() * 1000, false);
                 return deferred.promise;
              } else {
                 return results;
              }
           }
           function searchTextChange(text) {
              $log.info('Text changed to ' + text);
           }
           function selectedItemChange(item) {
              $log.info('Item changed to ' + JSON.stringify(item));
           }
           //build list of states as map of key-value pairs
           function loadStates() {
              var allStates = 'Alabama, Alaska, Arizona, Arkansas, California,
Colorado, Connecticut, Delaware,\
                 Florida, Georgia, Hawaii, Idaho, Illinois, Indiana, Iowa, Kansas,
Kentucky, Louisiana,\
                 Maine, Maryland, Massachusetts, Michigan, Minnesota, Mississippi,
Missouri, Montana,\
                 Nebraska, Nevada, New Hampshire, New Jersey, New Mexico, New York,
North Carolina,\
                 North Dakota, Ohio, Oklahoma, Oregon, Pennsylvania, Rhode Island,
South Carolina,\
                 South Dakota, Tennessee, Texas, Utah, Vermont, Virginia,
Washington, West Virginia,\
                 Wisconsin, Wyoming';
              return allStates.split(/, +/g).map( function (state) {
                 return {
                    value: state.toLowerCase(),
                    display: state
                 };
              });
           }
           //filter function for search query
           function createFilterFor(query) {
              var lowercaseQuery = angular.lowercase(query);
              return function filterFn(state) {
                 return (state.value.indexOf(lowercaseQuery) === 0);
              };
           }
        }
       </script>
   </head>
   <body ng-app="firstApplication" ng-cloak>
      <div ng-controller="autoCompleteController as ctrl" layout="column" ng-cloak>
         <md-content class="md-padding">
            <form ng-submit="$event.preventDefault()">
               <p><code>md-autocomplete</code> can be used to provide search results
from local or remote data sources.</p>
               <md-autocomplete
                  ng-disabled="ctrl.isDisabled"
                  md-no-cache="ctrl.noCache"
                  md-selected-item="ctrl.selectedItem"
                  md-search-text-change="ctrl.searchTextChange(ctrl.searchText)"
                  md-search-text="ctrl.searchText"
                  md-selected-item-change="ctrl.selectedItemChange(item)"
                  md-items="item in ctrl.querySearch(ctrl.searchText)"
                  md-item-text="item.display"
                  md-min-length="0"
                  placeholder="US State?">
                  <md-item-template>
                     <span md-highlight-text="ctrl.searchText" md-highlight-
flags="^i">{{item.display}}</span>
                  </md-item-template>
                  <md-not-found>
```

```
                     No states matching "{{ctrl.searchText}}" were found.
                     <a ng-click="ctrl.newState(ctrl.searchText)">Create a new 
one!</a>
                  </md-not-found>
               </md-autocomplete>
               <br/>
               <md-checkbox ng-model="ctrl.simulateQuery">Show progress for 
results?</md-checkbox>
               <md-checkbox ng-model="ctrl.noCache">Disable caching?</md-checkbox>
               <md-checkbox ng-model="ctrl.isDisabled">Disable?</md-checkbox>
               <p><code>md-autocomplete</code> caches results when performing a 
query.  After first call, it uses the cached results to eliminate unnecessary server 
requests or lookup logic and it can be disabled.</p>
            </form>
         </md-content>
      </div>
   </body>
</html>
```

Result

Verify the result.



md-autocomplete can be used to provide search results from local or remote data sources.

US State?

4. Angular Material – Bottom Sheet

The **$mdBottomSheet**, an Angular Service, is used to open a bottom sheet over the application and provides a simple promise API.

<table>
<tr><th>S.NO.</th><th>Method & Description</th></tr>
<tr><td>1</td><td>

$mdBottomSheet.show(options);

Show a bottom sheet with the specified options.

<table>
<tr><th>S.NO.</th><th>Parameter & Description</th></tr>
<tr><td>1</td><td>

*** options**

An options object, with the following properties:

templateUrl - {string=}: The url of an html template file that will be used as the content of the bottom sheet. Restrictions: the template must have an outer **md-bottom-sheet** element.

template - {string=}: Same as templateUrl, except this is an actual template string.

scope - {object=}: The scope to link the template / controller to. If none is specified, it will create a new child scope. This scope will be destroyed when the bottom sheet is removed unless preserveScope is set to true.

preserveScope - {boolean=}: It dictates whether to preserve the scope when the element is removed. By default, it is false.

controller - {string=}: The controller to associate with this bottom sheet.

locals - {string=}: An object containing key/value pairs. The keys will be used as names of values to inject into the controller. For example, **locals: {three: 3}** will inject three into the controller with the value of 3.

clickOutsideToClose - {boolean=}: It dictates whether the user can click outside the bottom sheet to close it. By default, it is true.

escapeToClose - {boolean=}: It dictates whether the user can press escape to close the bottom sheet. By default, it is true.

resolve - {object=}: Similar to locals, except that it takes promises as values and the bottom sheet will not open until the promises resolve.

controllerAs - {string=}: An alias to assign the controller to on the scope.

parent - {element=}: The element to append the bottom sheet to. The parent may be a function, string, object, or null. Defaults to appending to

</td></tr>
</table>

</td></tr>
</table>

<table>
<tr><th>S.NO.</th><th>Method & Description</th></tr>
<tr><td></td><td>
<table>
<tr><td></td><td>the body of the root element (or the root element) of the application. e.g. angular.element(document.getElementById('content')) or "#content".
disableParentScroll - {boolean=}: Whether to disable scrolling while the bottom sheet is open. Default true.</td></tr>
<tr><th>S.NO.</th><th>Returns & Description</th></tr>
<tr><td>1</td><td>**promise**
A promise that can be resolved with $mdBottomSheet.hide() or rejected with $mdBottomSheet.cancel().</td></tr>
</table>
</td></tr>
</table>

Example

The following example shows the use of the **$mdBottomSheet** service and also the use of bottom sheet.

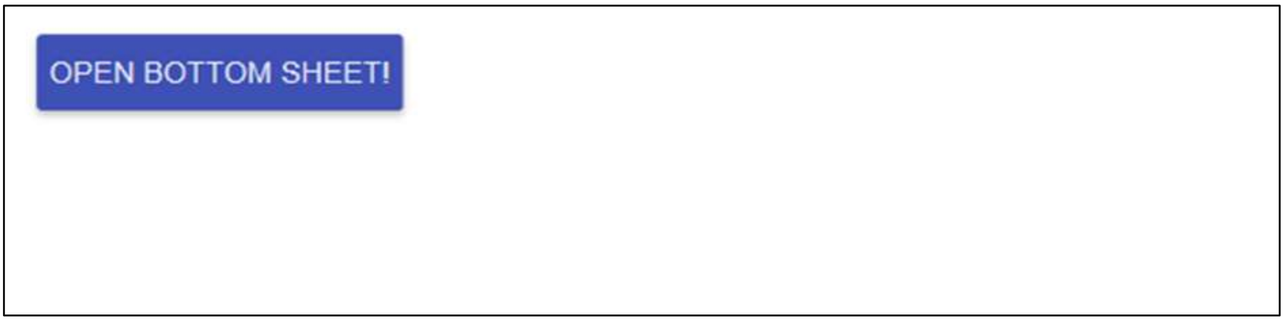
am_bottomsheet.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <script language="javascript">
        angular
           .module('firstApplication', ['ngMaterial'])
           .controller('bottomSheetController', bottomSheetController);
```

```
         function bottomSheetController ($scope, $mdBottomSheet) {
            $scope.openBottomSheet = function() {
               $mdBottomSheet.show({
                  template: '<md-bottom-sheet>Learn <b>Angular Material</b> @ 
TutorialsPoint.com!</md-bottom-sheet>'
               });
            };
         }
        </script>
    </head>
    <body ng-app="firstApplication">
       <div ng-controller="bottomSheetController as ctrl" layout="column">
          <md-content class="md-padding">
             <form ng-submit="$event.preventDefault()">
                <md-button class="md-raised md-primary" ng-click="openBottomSheet()">
                   Open Bottom Sheet!
                </md-button>
             </form>
          </md-content>
       </div>
    </body>
</html>
```

Result

Verify the result.



5. Angular Material – Cards

The **md-card**, an Angular Directive, is a container directive and is used to draw cards in the angularjs application. The following table lists down the angular directives and classes used in md-card.

S.NO.	Directive/Class & Description
1	**<md-card-header>** Header for the card, holds avatar, text and squared image.
2	**<md-card-avatar>** Card avatar.
3	**md-user-avatar** Class for user image.
4	**<md-icon>** Icon directive.
5	**<md-card-header-text>** Contains elements for the card description.
6	**md-title** Class for the card title.
7	**md-subhead** Class for the card sub header.
8	**<img>** Image for the card.
9	**<md-card-title>** Card content title.

10	**<md-card-title-text>** Card Title text container.
11	**md-headline** Class for the card content title.
12	**md-subhead** Class for the card content sub header.
13	**<md-card-title-media>** Squared image within the title.
14	**md-media-sm** Class for small image.
15	**md-media-md** Class for medium image.
16	**md-media-lg** Class for large image.
17	**<md-card-content>** Card content.
18	**md-media-xl** Class for extra large image.
19	**<md-card-actions>** Card actions.

20	**<md-card-icon-actions>** Icon actions.

Example

The following example shows the use of md-card directive and also the use of card classes.

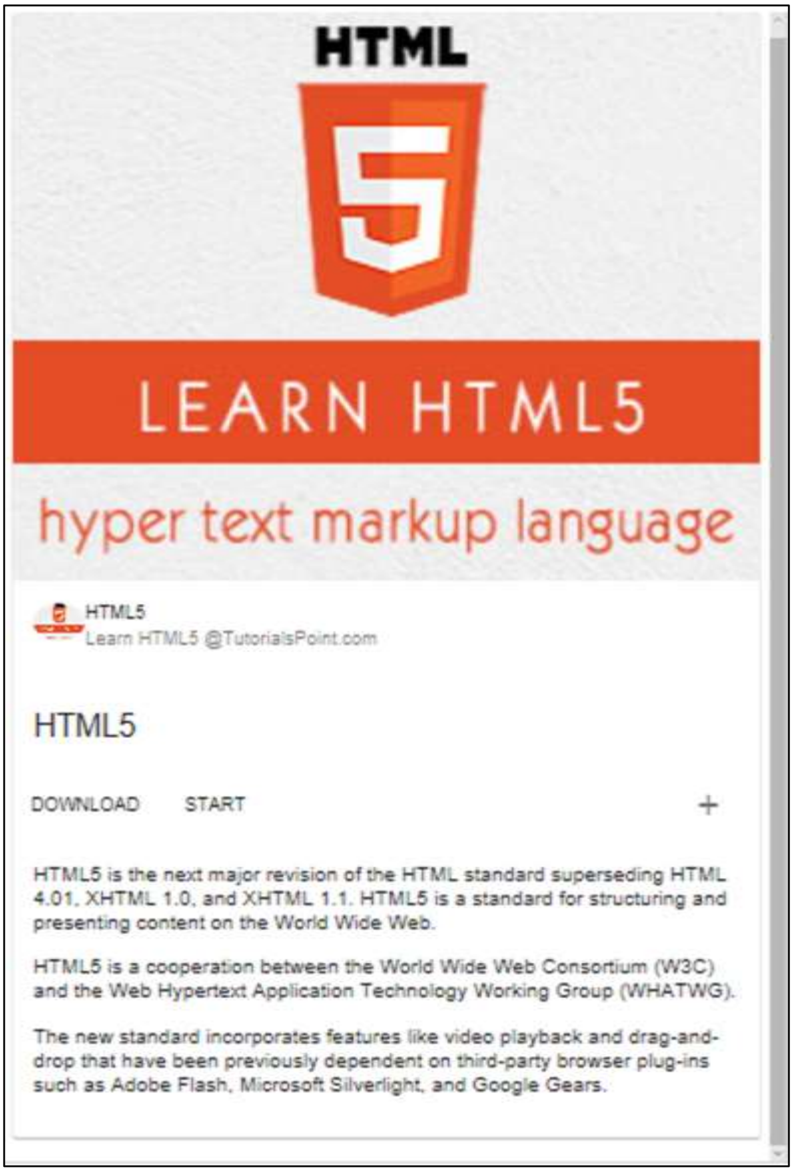
am_cards.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('cardController', cardController);

         function cardController ($scope) {
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <md-card>
         <img ng-src="html5-mini-logo.jpg" class="md-card-image" alt="Learn HTML5">
         <md-card-header>
```

```
            <md-card-avatar>
               <img class="md-user-avatar" src="html5-mini-logo.jpg">
            </md-card-avatar>
            <md-card-header-text>
               <span class="md-title">HTML5</span>
               <span class="md-subhead">Learn HTML5 @TutorialsPoint.com</span>
            </md-card-header-text>
         </md-card-header>
         <md-card-title>
            <md-card-title-text>
               <span class="md-headline">HTML5</span>
            </md-card-title-text>
         </md-card-title>
         <md-card-actions layout="row" layout-align="start center">
            <md-button>Download</md-button>
            <md-button>Start</md-button>
            <md-card-icon-actions>
               <md-button class="md-icon-button" aria-label="icon">
                  <md-icon class="material-icons">add</md-icon>
               </md-button>
            </md-card-icon-actions>
         </md-card-actions>
         <md-card-content>
            <p>HTML5 is the next major revision of the HTML standard superseding
HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and
presenting content on the World Wide Web.</p>
            <p>HTML5 is a cooperation between the World Wide Web Consortium (W3C)
and the Web Hypertext Application Technology Working Group (WHATWG).</p>
            </p>The new standard incorporates features like video playback and drag-
and-drop that have been previously dependent on third-party browser plug-ins such as
Adobe Flash, Microsoft Silverlight, and Google Gears.</p>
         </md-card-content>
      </md-card>
   </body>
</html>
```

Result

Verify the result.



HTML5

Learn HTML5 @TutorialsPoint.com

HTML5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.

6. Angular Material – Widgets

Angular Material provides a rich library of UI widgets. This allows users to have advanced interaction capability with the application.

The following table lists down a few important *Widgets* with their description:

S.NO.	Widget & Description
1	**Buttons** The **md-button**, an Angular Directive, is a button directive having optional ink ripples (and are by default enabled). If **href** or **ng-href** attribute is provided, then this directive acts as an anchor element.
2	**CheckBoxes** The **md-checkbox**, an Angular Directive, is used as a checkbox control.
3	**Content** The **md-content**, an Angular Directive, is a container element and is used for scrollable content. The **layout-padding** attribute can be added to have padded content.
4	**DatePicker** The **md-datepicker**, an Angular Directive, is an input control to select a date. It also supports ngMessages for input validation.
5	**Dialogs** The **md-dialog**, an Angular Directive, is a container element and is used to display a dialog box. Its element **md-dialog-content** contains the content of the dialog and the **md-dialog-actions** is responsible for dialog actions. The **mdDialog**, an Angular Service, opens a dialog over the application to keep the users informed and help them make decisions.

6	**Divider** The **md-divider**, an Angular Directive, is a rule element and is used to display a thin lightweight rule to group and divide contents within lists and page layouts.
7	**List** The **md-list**, an Angular directive, is a container component that contains **md-list-item** elements as a children. The md-list-item directive is a container component for row items of md-list container. CSS classes **md-2-line** and **md-3-line** can be added to md-list-item to increase the height of row with 22px and 40px respectively.
8	**Menu** The **md-menu**, an Angular directive, is a component to display addition options within the context of action performed. The **md-menu** has two child elements. The first element is the **trigger element** and is used to open the menu. The second element is the **md-menu-content** to represent the content of the menu when the menu is opened. The md-menu-content usually carries the menu items as md-menu-item.
9	**Menu Bar** The **md-menu-bar** an Angular directive, is a container component to hold multiple menus. The menu bar helps to create an operating system provided menu effect.
10	**Progress Bars** The **md-progress-circular** and **md-progress-linear** are Angular progress directives, and are used to show the loading content message in application.
11	**Radio Buttons** The **md-radio-group** and **md-radio-button** Angular directives are used to show radio buttons in the applcation. The md-radio-group is the grouping container for md-radio-button elements.
12	**Selects** The **md-select**, an Angular directives is used to show Select box, bounded by ng-model.

13	**Fab Toolbars** The **md-fab-toolbar**, an Angular directive, is used to show a toolbar of elements or buttons for quick access to common actions.
14	**Sliders** The **md-slider**, an Angular directives is used to show a range component. It has two modes: **normal** - User can slide between wide range of values. Default. **discrete** - User can slide between selected values. To enable discrete mode use md-discrete and step attributes.
15	**Tabs** The **md-tabs** and **md-tab** Angular directives are used to show tabs in the applcation. md-tabs is the grouping container for md-tab elements.
16	**Toolbars** The **md-toolbar**, an Angular directives is used to show a toolbar which is normally an area above content to show the title and the relevant buttons.
17	**Tooltips** The Angular Material provides various special methods to show unobtrusive tooltips to the users. Angular Material provides ways to assign directions for them and md-tooltip directive is used to show tooltips. A tooltip activates whenever the user focuses, hovers over, or touches the parent component.
18	**Chips** The **md-chips**, an Angular Directive, is used as a special component called the Chip and can be used to represent small set of information for example, a contact, tags etc. Custom template can be used to render the content of a chip. This can be achieved by specifying an md-chip-template element having the custom content as a child of md-chips.
19	**Contact Chips**

	The **md-contact-chips**, an Angular Directive, is an input control built on md-chips and uses the **md-autocomplete** element. The contact chip component accepts a query expression which returns a list of possible contacts. A user can select one of these and add it to the list of availble chips.

Angular Material – Buttons

The **md-button**, an Angular Directive, is a button directive having optional ink ripples (and are by default enabled). If **href** or **ng-href** attribute is provided, then this directive acts as an anchor element.

Attributes

The following table lists down the parameters and description of the different attributes of **md-button**.

S.NO.	Parameter & Description
1	**md-no-ink** If set as true, disable ripple ink effects.
2	**ng-disabled** Enable/Disable based on the expression.
3	**md-ripple-size** Overrides the default ripple size logic. Options: full, partial, auto.
4	**aria-label** Adds alternative text to button for accessibility, useful for icon buttons. If no default text is found, a warning will be logged.

Example

The following example shows the use of the md-button directive and also the various types of buttons.

am_buttons.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
```

```
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>

      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>

       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">

      <style>

         .buttondemo section {

            background: #f7f7f7;

            border-radius: 3px;

            text-align: center;

            margin: 1em;

            position: relative !important;

            padding-bottom: 10px;

         }

         .buttondemo md-content {

            margin-right: 7px;

         }

         .buttondemo section .md-button {

            margin-top: 16px;

            margin-bottom: 16px;

         }

         .buttondemo .label {

            position: absolute;

            bottom: 5px;

            left: 7px;

            font-size: 14px;

            opacity: 0.54;

         }

      </style>

      <script language="javascript">
```

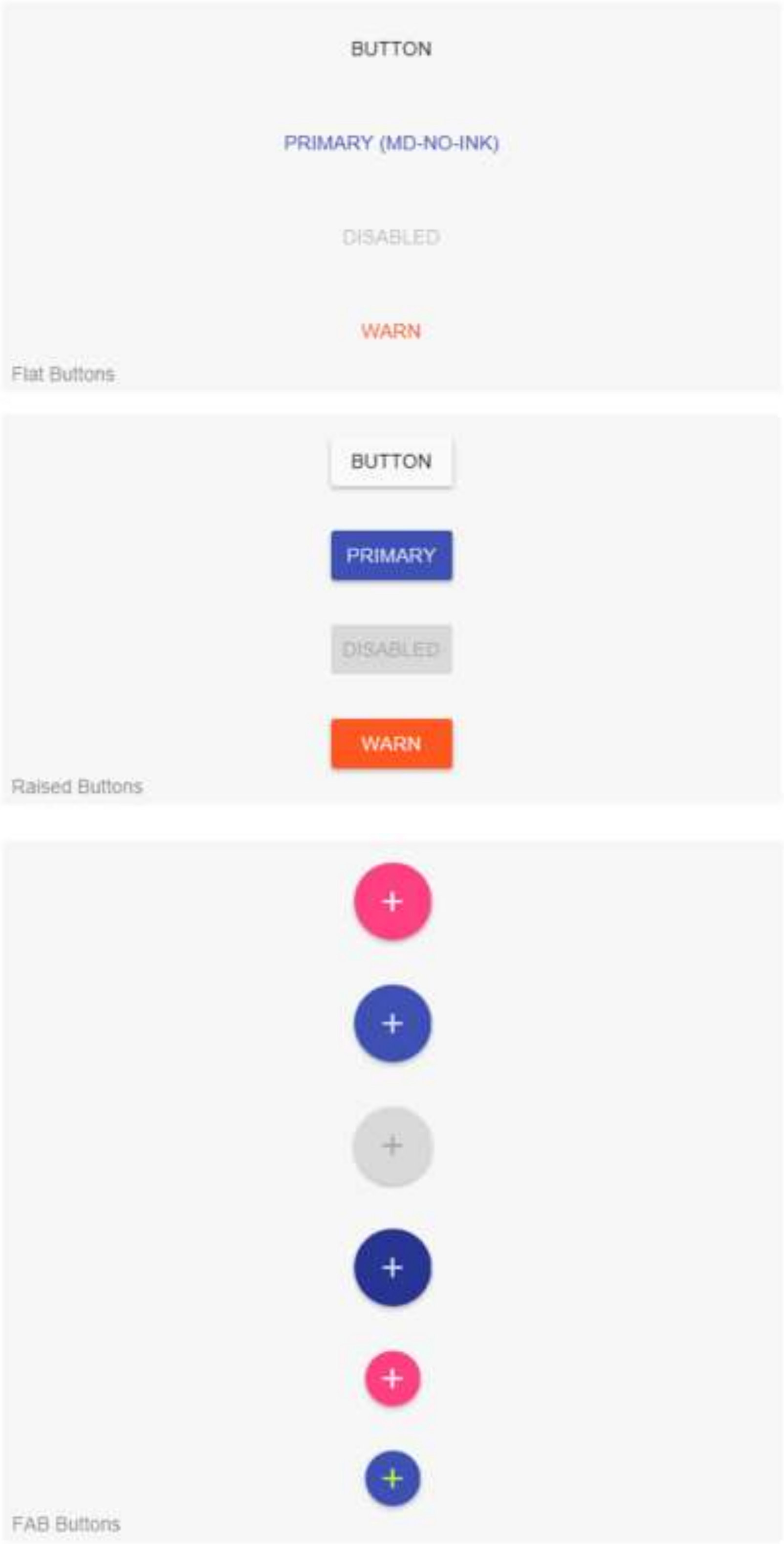
```
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('buttonController', buttonController);

         function buttonController ($scope) {
            $scope.title1 = 'Button';
            $scope.title4 = 'Warn';
            $scope.isDisabled = true;
            $scope.googleUrl = 'http://google.com';
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div class="buttondemo" ng-controller="buttonController">
         <md-content>
            <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
               <md-button>{{title1}}</md-button>
               <md-button md-no-ink class="md-primary">Primary (md-no-ink)</md-button>
               <md-button ng-disabled="true" class="md-primary">Disabled</md-button>
               <md-button class="md-warn">{{title4}}</md-button>
               <div class="label">Flat Buttons</div>
            </section>
            <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
               <md-button class="md-raised">Button</md-button>
               <md-button class="md-raised md-primary">Primary</md-button>
               <md-button ng-disabled="true" class="md-raised md-
primary">Disabled</md-button>
               <md-button class="md-raised md-warn">Warn</md-button>
               <div class="label">Raised Buttons</div>
            </section>
            <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
               <md-button class="md-fab" aria-label="add">
                  <md-icon class="material-icons">add</md-icon>
               </md-button>
```

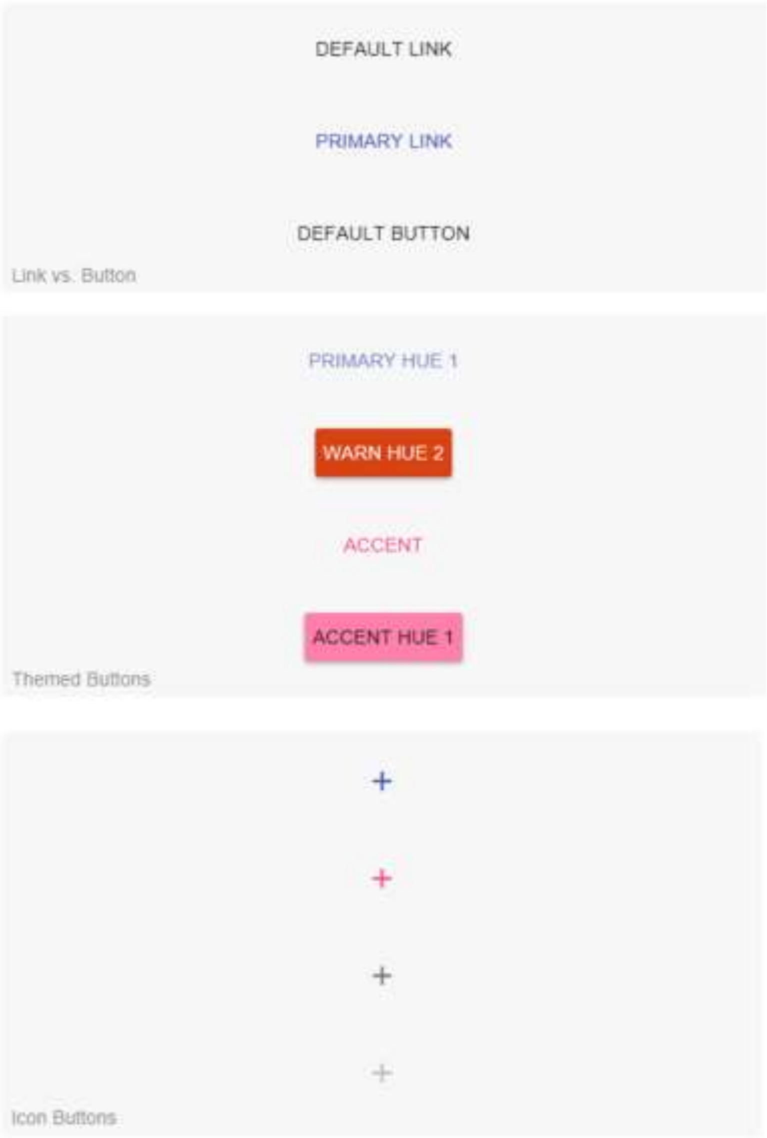
```
                <md-button class="md-fab md-primary" aria-label="add">
                   <md-icon class="material-icons">add</md-icon>
                </md-button>
                <md-button class="md-fab" ng-disabled="true" aria-label="add">
                   <md-icon class="material-icons">add</md-icon>
                </md-button>
                <md-button class="md-fab md-primary md-hue-2" aria-label="add">
                   <md-icon class="material-icons">add</md-icon>
                </md-button>
                <md-button class="md-fab md-mini" aria-label="add">
                   <md-icon class="material-icons">add</md-icon>
                </md-button>
                <md-button class="md-fab md-mini md-primary" aria-label="add">
                   <md-icon class="material-icons" style="color: greenyellow;">add</md-icon>
                </md-button>
                <div class="label">FAB Buttons</div>
             </section>
             <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
                <md-button ng-href="{{googleUrl}}" target="_blank">Default Link</md-button>
                <md-button class="md-primary" ng-href="{{googleUrl}}"
target="_blank">Primary Link</md-button>
                <md-button>Default Button</md-button>
                <div class="label">Link vs. Button</div>
             </section>
             <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
                <md-button class="md-primary md-hue-1">Primary Hue 1</md-button>
                <md-button class="md-warn md-raised md-hue-2">Warn Hue 2</md-button>
                <md-button class="md-accent">Accent</md-button>
                <md-button class="md-accent md-raised md-hue-1">Accent Hue 1</md-button>
                <div class="label">Themed Buttons</div>
             </section>
             <section layout="row" layout-sm="column" layout-align="center center"
layout-wrap>
                <md-button class="md-icon-button md-primary" aria-label="Add">
                   <md-icon class="material-icons">add</md-icon>
                </md-button>
```

```
               <md-button class="md-icon-button md-accent" aria-label="Add">
                  <md-icon class="material-icons">add</md-icon>
               </md-button>
               <md-button class="md-icon-button" aria-label="Add">
                  <md-icon class="material-icons">add</md-icon>
               </md-button>
               <md-button href="http://google.com"
                  title="Open Google.com in new window"
                  target="_blank"
                  ng-disabled="true"
                  class="md-icon-button launch" >
                  <md-icon  class="material-icons">add</md-icon>
               </md-button>
               <div class="label">Icon Buttons</div>
            </section>
         </md-content>
      </div>
   </body>
</html>
```

Result

Verify the result.





Angular Material – CheckBoxes

The **md-checkbox**, an Angular Directive, is used as a checkbox control.

Attributes

The following table lists down the parameters and description of the different attributes of **md-checkbox**.

S.NO.	Parameter & Description
1	*** ng-model** Assignable angular expression to data-bind to.

S.NO.	Parameter & Description
2	**name** The property name of the form under which the control is published.
3	**ng-true-value** The value to which the expression should be set when selected.
4	**ng-false-value** The value to which the expression should be set when not selected.
5	**ng-change** Angular expression to be executed when input changes due to user interaction with the input element.
6	**md-no-ink** Use of attribute indicates the use of ripple ink effects.
7	**aria-label** Adds label to checkbox for accessibility. Defaults to checkbox's text. If no default text is found, a warning will be logged.

Example

The following example shows the use of the md-checkbox directive and also the use of angular checkboxes.

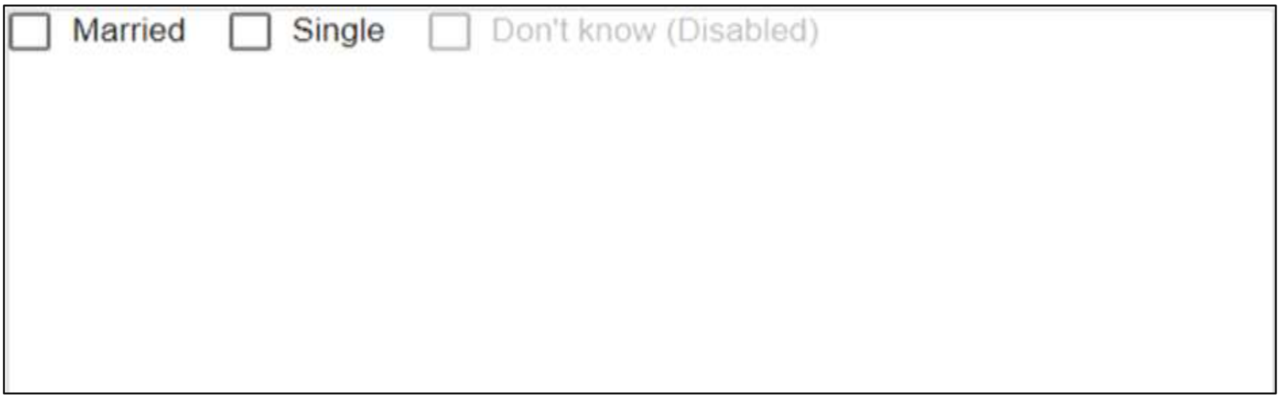
am_checkboxes.htm

```
<html lang="en" >

   <head>

      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">

      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
```

```
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('checkBoxController', checkBoxController);

         function checkBoxController ($scope) {
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <md-checkbox ng-model="isChecked" aria-label="Married">
         Married
      </md-checkbox>
      <md-checkbox md-no-ink ng-model="hasInk" aria-label="No Ink Effects">
         Single
      </md-checkbox>
      <md-checkbox ng-disabled="true" ng-model="isDisabled" aria-label="Disabled">
         Don't know (Disabled)
      </md-checkbox>
   </body>
</html>
```

Result

Verify the result.



□ Married □ Single □ Don't know (Disabled)

Angular Material – Content

The **md-content**, an Angular Directive, is a container element and is used for scrollable content. The **layout-padding** attribute can be added to have padded content.

Example

The following example shows the use of md-content directive and also the use of angular content display.

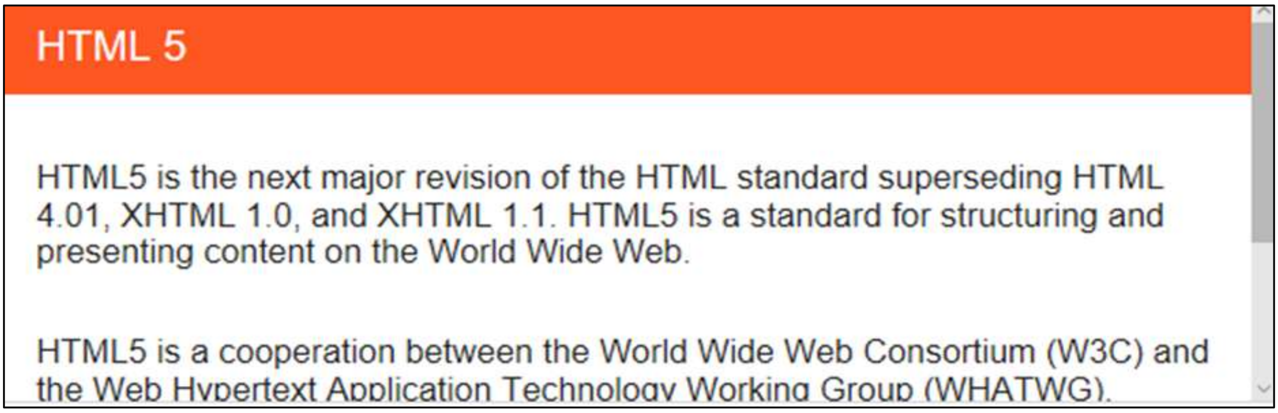
am_content.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script type="text/javascript">
         angular.module('firstApplication', ['ngMaterial']);
      </script>
   </head>
   <body ng-app="firstApplication" ng-cloak>
      <md-toolbar class="md-warn">
         <div class="md-toolbar-tools">
            <h2 class="md-flex">HTML 5</h2>
         </div>
      </md-toolbar>
      <md-content flex layout-padding>
         <p>HTML5 is the next major revision of the HTML standard superseding HTML
4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting
content on the World Wide Web.</p>
         <p>HTML5 is a cooperation between the World Wide Web Consortium (W3C) and
the Web Hypertext Application Technology Working Group (WHATWG).</p>
```

```
         <p>The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.</p>
      </md-content>
   </body>
</html>
```

Result

Verify the result.



HTML 5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

Angular Material – DatePicker

The **md-datepicker**, an Angular Directive, is an input control to select a date and supports **ngMessages** for input validation.

Attributes

The following table lists out the parameters and description of the different attributes of **md-datepicker**.

S.NO.	Parameter & Description
1	*** ng-model** This component's model expects a JavaScript Date object.
2	**ng-change** The expression evaluated when the model value changes.
3	**md-min-date** The expression representing a min date (inclusive).
4	**md-max-date** The expression representing a max date (inclusive).

S.NO.	Parameter & Description
5	**md-date-filter** The function expecting a date and returning a boolean and determines whether it can be selected or not.
6	**md-placeholder** The date input placeholder value.
7	**ng-disabled** This determines whether the datepicker is disabled.
8	**ng-required** This determines whether a value is required for the datepicker.

Example

The following example shows the use of md-datepicker directive and also the uses of datepicker.

am_datepicker.htm

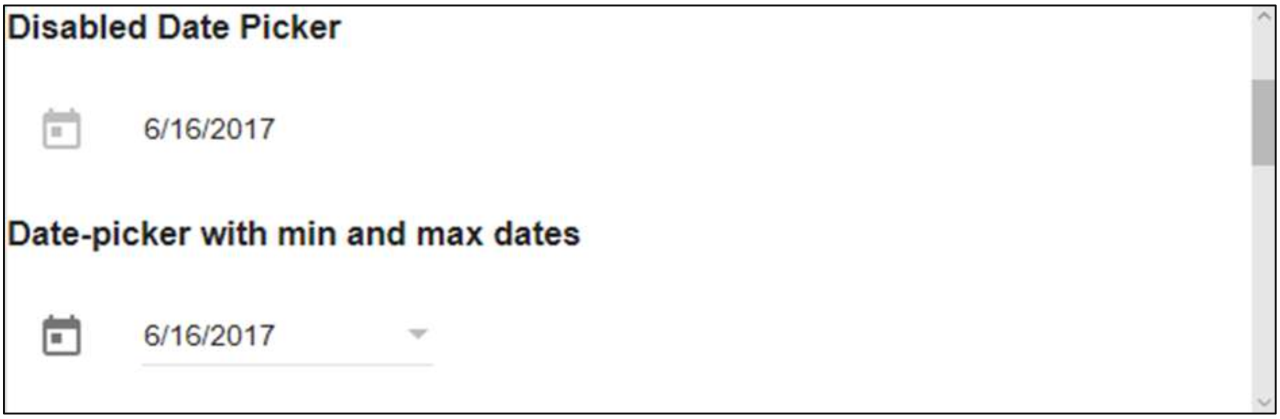
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <style>
         .datepickerdemo md-content {
            padding-bottom: 200px;
         }
        .datepickerdemo .validation-messages {
            font-size: 11px;
            color: darkred;
            margin: 10px 0 0 25px;
         }
      </style>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial','ngMessages'])
            .controller('dateController', dateController);

         function dateController ($scope) {
            $scope.myDate = new Date();
            $scope.minDate = new Date(
               $scope.myDate.getFullYear(),
               $scope.myDate.getMonth() - 2,
               $scope.myDate.getDate());
            $scope.maxDate = new Date(
               $scope.myDate.getFullYear(),
               $scope.myDate.getMonth() + 2,
               $scope.myDate.getDate());
            $scope.onlyWeekendsPredicate = function(date) {
               var day = date.getDay();
               return day === 0 || day === 6;
            }
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div class="datepickerdemo" ng-controller="dateController as ctrl"
layout="column" ng-cloak>
         <md-content>
            <h4>Standard Date Picker</h4>
            <md-datepicker
               ng-model="myDate"
               md-placeholder="Enter date"></md-datepicker>
            <h4>Disabled Date Picker</h4>
            <md-datepicker
```

```
               ng-model="myDate"
               md-placeholder="Enter date"
               disabled></md-datepicker>
            <h4>Date-picker with min and max dates</h4>
            <md-datepicker
               ng-model="myDate"
               md-placeholder="Enter date"
               md-min-date="minDate"
               md-max-date="maxDate"></md-datepicker>
            <h4>Date-picker with only selectable weekends</h4>
            <md-datepicker
               ng-model="myDate"
               md-placeholder="Enter date"
               md-date-filter="onlyWeekendsPredicate"></md-datepicker>
            <h4>Date-picker with only selectable weekends in a range</h4>
            <md-datepicker
               ng-model="myDate"
               md-placeholder="Enter date"
               md-min-date="minDate"
               md-max-date="maxDate"
               md-date-filter="onlyWeekendsPredicate"></md-datepicker>
            <h4>Date-picker with validation mesages</h4>
            <form name="myForm">
               <md-datepicker
                  name="dateField"
                  ng-model="myDate"
                  md-placeholder="Enter date"
                  required
                  md-min-date="minDate"
                  md-max-date="maxDate"
                  md-date-filter="onlyWeekendsPredicate"></md-datepicker>
               <div class="validation-messages" ng-
messages="myForm.dateField.$error">
                  <div ng-message="valid">The entered value is not a date!</div>
                  <div ng-message="required">This date is required!</div>
                  <div ng-message="mindate">Date is too early!</div>
```

```
                    <div ng-message="maxdate">Date is too late!</div>
                    <div ng-message="filtered">Only weekends are allowed!</div>
                 </div>
              </form>
           </md-content>
          </div>
    </body>
</html>
```

Result

Verify the result.



Disabled Date Picker

6/16/2017

Date-picker with min and max dates

6/16/2017

Angular Material – Dialogs

The **md-dialog**, an Angular Directive, is a container element and is used to display a dialog box. Its element, the **md-dialog-content**, contains the content of the dialog and the **md-dialog-actions** is responsible for the dialog actions.

The **mdDialog**, an Angular Service, opens a dialog over the application to inform the users about the information and help them make decisions.

The following example shows the use of md-dialog directive and mdDialog service and also the use of angular dialog boxes.

am_dialog.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('dialogController', dialogController);

         function dialogController ($scope, $mdDialog) {
            $scope.status = '';
                  $scope.items = [1,2,3,4,5];
            $scope.showAlert = function(ev) {
               $mdDialog.show(
                  $mdDialog.alert()

.parent(angular.element(document.querySelector('#dialogContainer')))
                     .clickOutsideToClose(true)
                     .title('TutorialsPoint.com')
                     .textContent('Welcome to TutorialsPoint.com')
                     .ariaLabel('Welcome to TutorialsPoint.com')
                     .ok('Ok!')
                     .targetEvent(ev)
               );
            };
           $scope.showConfirm = function(event) {
               var confirm = $mdDialog.confirm()
```

```
                  .title('Are you sure to delete the record?')
                  .textContent('Record will be deleted permanently.')
                  .ariaLabel('TutorialsPoint.com')
                  .targetEvent(event)
                  .ok('Yes')
                  .cancel('No');
                  $mdDialog.show(confirm).then(function() {
                     $scope.status = 'Record deleted successfully!';
                     }, function() {
                        $scope.status = 'You decided to keep your record.';
                  });
            };

            $scope.showCustom = function(event) {
               $mdDialog.show({
                  clickOutsideToClose: true,
                  scope: $scope,
                  preserveScope: true,
                  template: '<md-dialog>' +
                              '  <md-dialog-content>' +
                              '     Welcome to TutorialsPoint.com' +
                              '  </md-dialog-content>' +
                              '</md-dialog>',
                  controller: function DialogController($scope, $mdDialog) {
                     $scope.closeDialog = function() {
                        $mdDialog.hide();
                     }
                  }
               });
            };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="dialogContainer" ng-controller="dialogController as ctrl"
layout="row" ng-cloak>
```

```
         <md-content>
            <h4>Standard Alert</h4>
            <md-button class="md-primary md-raised" ng-click="showAlert($event)" flex="100"  flex-gt-md="auto">
               Alert
            </md-button>
            <h4>Confirm Dialog Box</h4>
            <md-button class="md-primary md-raised" ng-click="showConfirm($event)" flex="100"  flex-gt-md="auto">
               Confirm
            </md-button>
            <h4>Custom Dialog Box</h4>
            <md-button class="md-primary md-raised" ng-click="showCustom($event)" flex="100"  flex-gt-md="auto">
               Custom
            </md-button>
            <div ng-if="status">
               <br/>
               <b layout="row" layout-align="center center" class="md-padding">
                  {{status}}
               </b>
            </div>
         </md-content>
        </div>
   </body>
</html>
```

Result

Verify the result.

Standard Alert

ALERT

Confirm Dialog Box

CONFIRM

Angular Material – Divider

The **md-divider**, an Angular Directive, is a rule element and is used to display a thin lightweight rule to group and divide contents within lists and page layouts.

Attributes

The following table lists out the parameters and description of the different attributes of **md-divider**.

S.NO.	Parameter & Description
1	**md-inset** Add this attribute to activate the inset divider style.

Example

The following example shows the use of md-divider directive and also the uses of dividers.

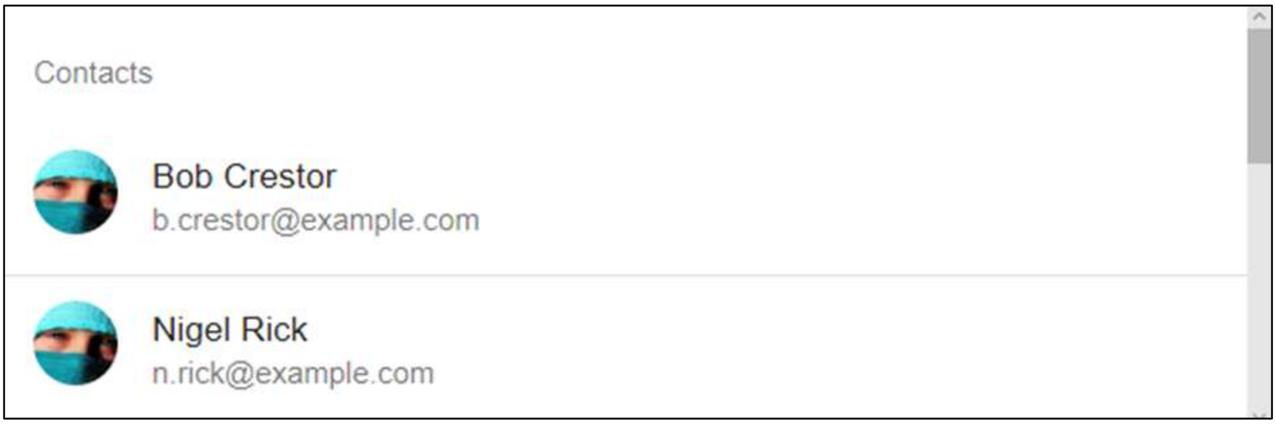
am_dividers.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('dividerController', dividerController);

         function dividerController ($scope) {
            var self = this;
            self.allContacts = loadContacts();
            self.contacts = [self.allContacts[0]];

            function loadContacts() {
               var contacts = [
                  'Roberto Karlos',
                  'Bob Crestor',
                  'Nigel Rick',
                  'Narayana Garner'
               ];
               return contacts.map(function (c, index) {
                     var cParts = c.split(' ');
                     var contact = {
                        name: c,
                        email: cParts[0][0].toLowerCase() + '.' +
cParts[1].toLowerCase() + '@example.com',
                        image: 'http://lorempixel.com/50/50/people?' + index
                     };
                     contact._lowername = contact.name.toLowerCase();
                     return contact;
               });
            }
         }
      </script>
   </head>
   <body ng-app="firstApplication">
```

```
      <div id="dividerContainer" ng-controller="dividerController as ctrl" 
layout="column" ng-cloak>
         <md-content>
             <md-list>
               <md-subheader class="md-no-sticky">Contacts</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index, 
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar" 
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                           <md-divider ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
                  <md-list>
               <md-subheader class="md-no-sticky">Contacts (With Insets)</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index, 
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar" alt="{{contact.name}}"/>
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                           <md-divider md-inset ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
         </md-content>
        </div>
   </body>
</html>
```

Result

Verify the result.



Contacts

Bob Crestor
b.crestor@example.com

Nigel Rick
n.rick@example.com

Angular Material – List

The **md-list**, an Angular directive, is a container component that contains the **md-list-item** elements as children. The md-list-item directive is a container component for row items of md-list container. The CSS classes, **md-2-line** and **md-3-line**, can be added to md-list-item to increase the height of row with 22px and 40px respectively.

Example

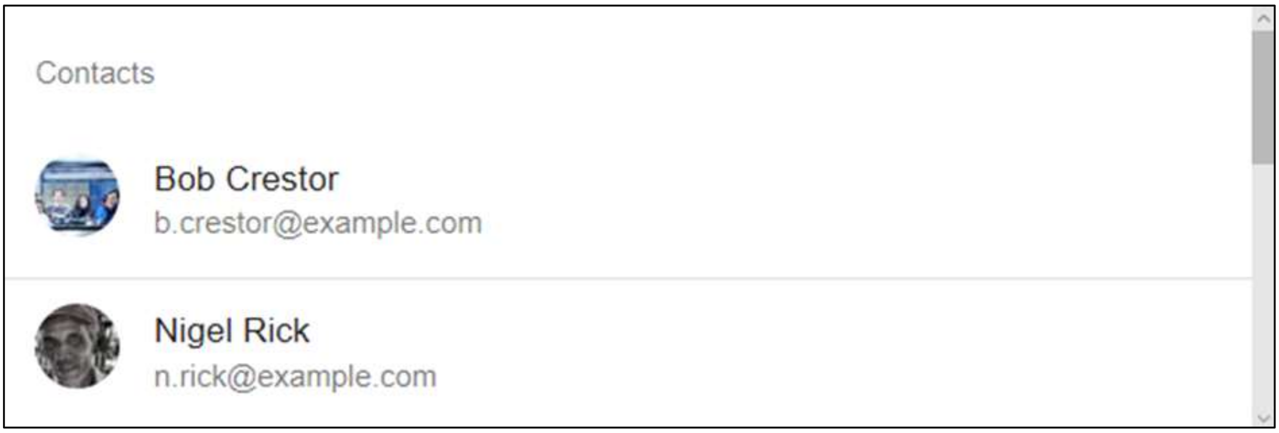
The following example shows the use of md-list directive and also the uses of lists.

am_lists.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.js"></script>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('listController', listController);
         function listController ($scope) {
            var self = this;
            self.allContacts = loadContacts();
            self.contacts = [self.allContacts[0]];

            function loadContacts() {
               var contacts = [
                  'Roberto Karlos',
                  'Bob Crestor',
                  'Nigel Rick',
                  'Narayana Garner'
               ];
               return contacts.map(function (c, index) {
                     var cParts = c.split(' ');
                     var contact = {
                        name: c,
                        email: cParts[0][0].toLowerCase() + '.' +
cParts[1].toLowerCase() + '@example.com',
                        image: 'http://lorempixel.com/50/50/people?' + index
                     };
                     contact._lowername = contact.name.toLowerCase();
                     return contact;
               });
            }
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="listContainer" ng-controller="listController as ctrl" layout="column"
ng-cloak>
         <md-content>
             <md-list>
               <md-subheader class="md-no-sticky">Contacts</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index,
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar" alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                           <md-divider ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
                  <md-list>
               <md-subheader class="md-no-sticky">Contacts (With Insets)</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index,
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar"
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                           <md-divider md-inset ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
         </md-content>
        </div>
   </body>
</html>
```

Result

Verify the result.



Contacts

Bob Crestor
b.crestor@example.com

Nigel Rick
n.rick@example.com

Angular Material – Menu

The **md-menu**, an Angular directive, is a component to display addition options within the context of action performed. **md-menu** have two child elements. The first element is the trigger element and is used to open the menu. The second element is the **md-menu-content** which represents the content of the menu when menu is opened. The **md-menu-content** usually carries the menu items as **md-menu-item**.

Attributes

The following table lists out the parameters and description of the different attributes of **md-menu**.

S. No.	Parameter & Description
1	*** md-position-mode** The position mode in the form of x, y. Default value is target,target. Right now the x axis also suppports target-right.
2	*** md-offset** An offset to apply to the dropdown after positioning x, y. Default value is 0,0.

Example

The following example shows the use of md-menu directive and also the uses of menus.

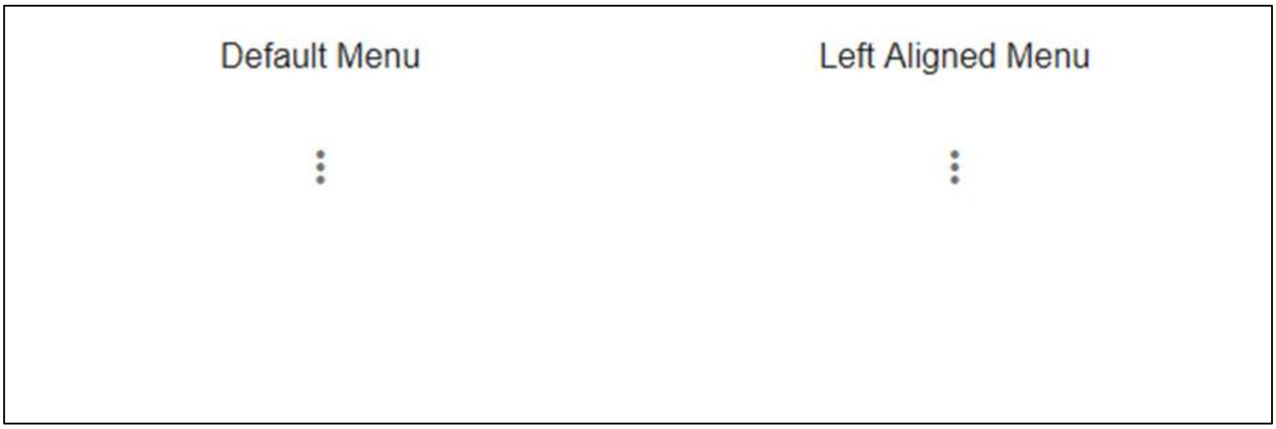
am_menus.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('menuController', menuController);

         function menuController ($scope, $mdDialog) {
            var originatorEv;
            this.openMenu = function($mdOpenMenu, ev) {
            originatorEv = ev;
               $mdOpenMenu(ev);
            };
            this.menuItemClick = function(index) {
               $mdDialog.show(
                  $mdDialog.alert()
                    .title('TutorialsPoint.com')
                    .textContent('Menu Item clicked, index: ' + index)
                    .ok('OK')
                    .targetEvent(originatorEv)
               );
               originatorEv = null;
            };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="menuContainer" ng-controller="menuController as ctrl" layout="row" ng-cloak>
         <div layout="column" flex="33" flex-sm="100" layout-align="center center">
            <p>Default Menu</p>
            <md-menu>
               <md-button aria-label="Sample Menu" class="md-icon-button" ng-click="$mdOpenMenu($event)">
                  <md-icon md-menu-origin class="material-icons">more_vert</md-icon>
               </md-button>
               <md-menu-content width="6">
                  <md-menu-item ng-repeat="item in [1, 2, 3]">
                     <md-button ng-click="ctrl.menuItemClick($index)">
                        <div layout="row">
                           <md-icon md-menu-align-target class="material-icons">add</md-icon>
                           <p flex>Option {{item}}</p>
                        </div>
                     </md-button>
                  </md-menu-item>
               </md-menu-content>
            </md-menu>
         </div>
         <div layout="column" flex-sm="100" flex="33" layout-align="center center">
            <p>Left Aligned Menu</p>
            <md-menu md-position-mode="target-right target" >
               <md-button aria-label="Sample Menu" class="md-icon-button" ng-click="$mdOpenMenu($event)">
                  <md-icon md-menu-origin class="material-icons">more_vert</md-icon>
               </md-button>
               <md-menu-content width="4" >
                  <md-menu-item ng-repeat="item in [1, 2, 3]">
                     <md-button ng-click="ctrl.menuItemClick($index)">
                        <div layout="row">
```

```
                           <p flex>Option {{item}}</p>
                           <md-icon md-menu-align-target class="material-
icons">add</md-icon>
                        </div>
                     </md-button>
                  </md-menu-item>
               </md-menu-content>
            </md-menu>
          </div>
        </div>
   </body>
</html>
```

Result

Verify the result.



Angular Material - Menu Bar

The md-menu-bar an Angular directive, is a container component to hold multiple menus. Menu bar helps to create a operating system provided menu effect.

Example

The following example shows the use of md-menu-bar directive and also the uses of menu-bar.

am_menubar.htm

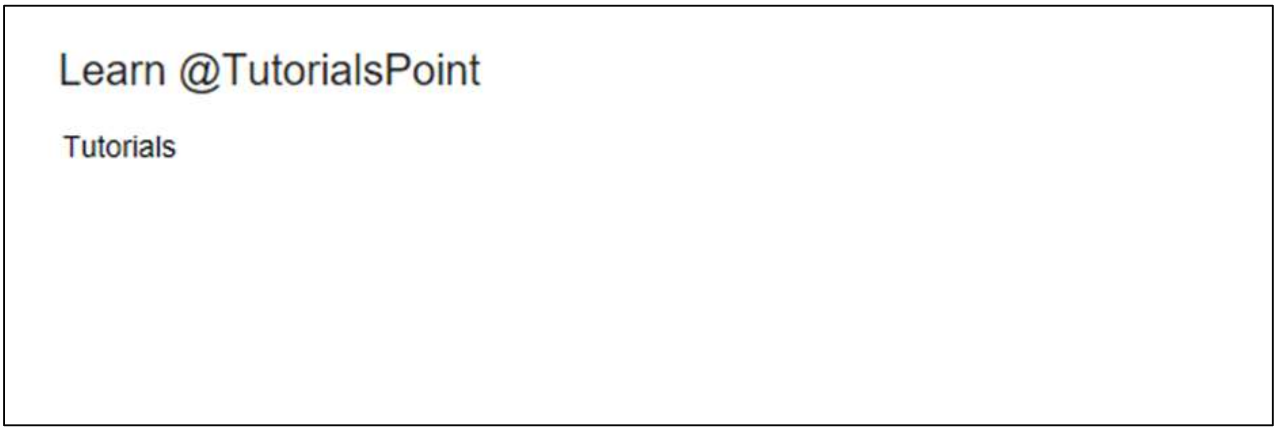
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('menubarController', menubarController);

         function menubarController ($scope, $mdDialog) {
            this.sampleAction = function(name, ev) {
               $mdDialog.show($mdDialog.alert()
                  .title(name)
                  .textContent('Start learning "' + name + '!')
                  .ok('OK')
                  .targetEvent(ev)
               );
            };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="menubarContainer" ng-controller="menubarController as ctrl"
layout="row" ng-cloak>
```

```
         <md-toolbar class="md-menu-toolbar">
            <div layout="row">
               <div>
                  <h2 class="md-toolbar-tools">Learn @TutorialsPoint</h2>
                  <md-menu-bar>
                     <md-menu>
                        <button ng-click="$mdOpenMenu()">Tutorials</button>
                        <md-menu-content>
                           <md-menu-item>
                              <md-button ng-click="ctrl.sampleAction('share',
$event)">Share...</md-button>
                           </md-menu-item>
                           <md-menu-divider></md-menu-divider>
                           <md-menu-item>
                              <md-menu>
                                 <md-button ng-click="$mdOpenMenu()">Learn</md-button>
                                 <md-menu-content>
                                    <md-menu-item><md-button ng-
click="ctrl.sampleAction('HTML5', $event)">HTML5</md-button></md-menu-item>
                                    <md-menu-item><md-button ng-
click="ctrl.sampleAction('jQuery', $event)">jQuery</md-button></md-menu-item>
                                    <md-menu-item><md-button ng-
click="ctrl.sampleAction('AngularJS', $event)">AngularJS      </md-button></md-menu-
item>
                                    <md-menu-item><md-button disabled="disabled" ng-
click="ctrl.sampleAction('AngularJS 2.0', $event)">AngularJS 2.0</md-button></md-
menu-item>
                                    <md-menu-divider></md-menu-divider>
                                    <md-menu-item><md-button ng-
click="ctrl.sampleAction('CSS', $event)">CSS</md-button></md-menu-item>
</md-menu-content>
                              </md-menu>
                           </md-menu-item>
                        </md-menu-content>
                     </md-menu>
                  </md-menu-bar>
               </div>
            </div>
         </md-toolbar>
```

```
      </div>
   </body>
</html>
```

Result

Verify the result.



Learn @TutorialsPoint

Tutorials

Angular Material – Progress Bars

The **md-progress-circular** and **md-progress-linear** are Angular progress directives, and are used to show loading content message in application.

Attributes – md-progress-circular

The following table lists out the parameters and description of the different attributes of **md-progress-circular**.

S.NO.	Parameter & Description
1	*** md-mode** Select from one of the two modes: **'determinate'** and **'indeterminate'**. If the md-mode value is set as undefined or specified as not 1 of the two (2) valid modes, then .ng-hide will be auto-applied as a style to the component; if not configured, the md-mode="indeterminate" will be auto injected as an attribute. If value="" is also specified, however, then md-mode="determinate" will be auto-injected instead.
2	**value** In determinate mode, this number represents the percentage of the circular progress. By default, this is 0.
3	**md-diameter** This specifies the diameter of the circular progress. The value may be a

S.NO.	Parameter & Description
	percentage (eg '25%') or a pixel-size value (eg '48'). If this attribute is not present, then a default value of '48px' is assumed.

Example

The following example shows the use of the **md-progress-circular** directive and also the uses of circular progress bars.

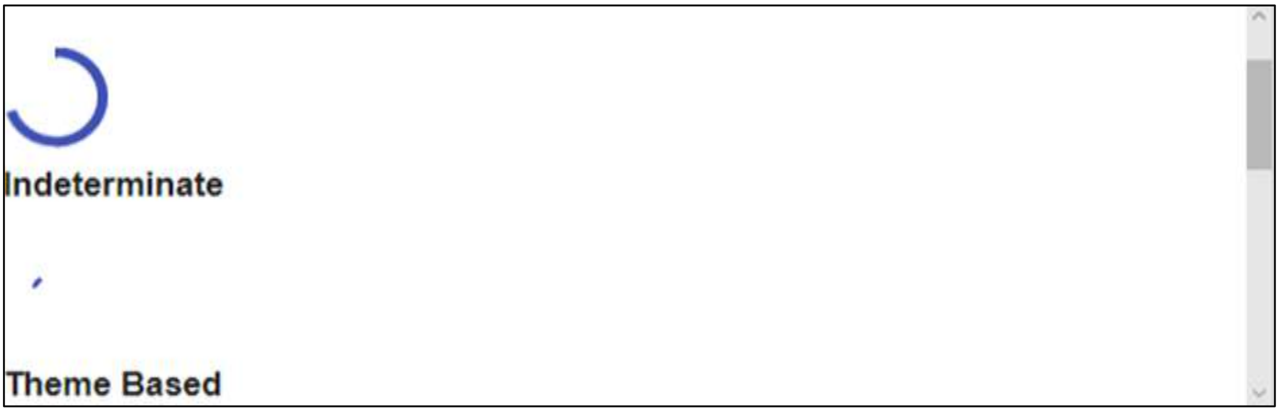
am_circularprogressbars.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('progressbarsController', progressbarsController);

         function progressbarsController ($scope, $interval) {
            var self = this,  j= 0, counter = 0;
            self.modes = [ ];
            self.activated = true;
            self.determinateValue = 30;
            self.toggleActivation = function() {
               if ( !self.activated ) self.modes = [ ];
               if (  self.activated ) j = counter = 0;
            };

            $interval(function() {
               self.determinateValue += 1;
               if (self.determinateValue > 100) {
                  self.determinateValue = 30;
               }
               if ( (j < 5) && !self.modes[j] && self.activated ) {
                  self.modes[j] = 'indeterminate';
               }
               if ( counter++ % 4 == 0 ) j++;
            }, 100, 0, true);
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="progressbarsContainer" ng-controller="progressbarsController as ctrl"
layout="column" ng-cloak>
         <h4 style="margin-top:10px">Determinate </h4><p></p>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-circular md-mode="determinate"
value="{{ctrl.determinateValue}}"></md-progress-circular>
         </div>
         <h4 style="margin-top:10px">Indeterminate </h4>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-circular md-mode="indeterminate"></md-progress-circular>
         </div>
         <h4 style="margin-top:10px">Theme Based</h4>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-circular class="md-hue-2" md-mode="{{ctrl.modes[0]}}" md-
diameter="20px"></md-progress-circular>
            <md-progress-circular class="md-accent" md-mode="{{ctrl.modes[1]}}" md-
diameter="40"></md-progress-circular>
            <md-progress-circular class="md-accent md-hue-1" md-
mode="{{ctrl.modes[2]}}" md-diameter="60"></md-progress-circular>
            <md-progress-circular class="md-warn md-hue-3" md-
mode="{{ctrl.modes[3]}}" md-diameter="70"></md-progress-circular>
```

```
            <md-progress-circular md-mode="{{ctrl.modes[4]}}" md-diameter="96"></md-progress-circular>
         </div>
         <hr ng-class="{'visible' : ctrl.activated}">
         <div id="loaders" layout="row" layout-align="start center">
            <p>Progress Circular Indicators:    </p>
            <h5>Off</h5>
            <md-switch
               ng-model="ctrl.activated"
               ng-change="ctrl.toggleActivation()"
               aria-label="Toggle Progress Circular Indicators">
               <h5>On</h5>
            </md-switch>
         </div>
       </div>
   </body>
</html>
```

Result

Verify the result.



Attributes – md-progress-linear

The following table lists out the parameters and description of the different attributes of **md-progress-linear**.

S.NO.	Parameter & Description
1	*** md-mode**

S.NO.	Parameter & Description
	Select from one of the two modes: **'determinate'** and **'indeterminate'**. If the md-mode value is set as undefined or specified as not 1 of the two (2) valid modes, then **.ng-hide** will be auto-applied as a style to the component; if not configured, the **md-mode="indeterminate"** will be auto injected as an attribute. If **value=""** is also specified, however, then **md-mode="determinate"** will be auto-injected instead.
2	**md-buffer-value** In determinate mode, this number represents the percentage of the primary progress bar. By default, this is 0.
3	**md-diameter** In the buffer mode, this number represents the percentage of the secondary progress bar. By default, this is 0.

Example

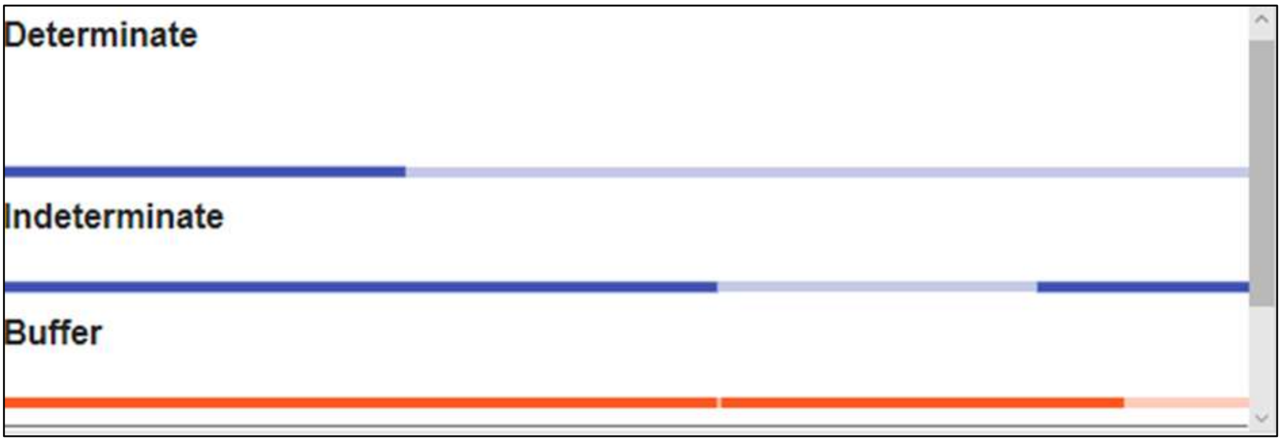
The following example shows the use of the **md-progress-circular** directive and also the uses of linear progress bars.

am_linearprogressbars.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('progressbarsController', progressbarsController);

         function progressbarsController ($scope, $interval) {
            var self = this,  j= 0, counter = 0;
            self.modes = [ ];
            self.activated = true;
            self.determinateValue = 30;
            self.toggleActivation = function() {
               if ( !self.activated ) self.modes = [ ];
               if (  self.activated ) j = counter = 0;
            };

            $interval(function() {
               self.determinateValue += 1;
               if (self.determinateValue > 100) {
                  self.determinateValue = 30;
               }
               if ( (j < 5) && !self.modes[j] && self.activated ) {
                  self.modes[j] = 'indeterminate';
               }
               if ( counter++ % 4 == 0 ) j++;
            }, 100, 0, true);
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="progressbarsContainer" ng-controller="progressbarsController as ctrl"
layout="column" ng-cloak>
         <h4 style="margin-top:10px">Determinate </h4><p></p>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-linear md-mode="determinate"
value="{{ctrl.determinateValue}}"></md-progress-circular>
         </div>
         <h4 style="margin-top:10px">Indeterminate </h4>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-linear md-mode="indeterminate"></md-progress-circular>
         </div>
         <h4 style="margin-top:10px">Buffer</h4>
         <div layout="row" layout-sm="column" layout-align="space-around">
            <md-progress-linear class="md-warn" md-mode="{{ctrl.modes[0]}}" 
value="{{ctrl.determinateValue}}" md-buffer-value="{{ctrl.determinateValue2}}">
</md-progress-linear>
         </div>
         <hr ng-class="{'visible' : ctrl.activated}">
         <div id="loaders" layout="row" layout-align="start center">
            <p>Progress Buffer Indicators:    </p>
            <h5>Off</h5>
            <md-switch
               ng-model="ctrl.activated"
               ng-change="ctrl.toggleActivation()"
               aria-label="Toggle Buffer Progress Indicators">
               <h5>On</h5>
            </md-switch>
         </div>
       </div>
   </body>
</html>
```

Result

Verify the result.



Angular Material – Radio Buttons

The **md-radio-group** and **md-radio-button** Angular directives are used to show radio buttons in the applcation. The md-radio-group is the grouping container for md-radio-button elements.

Attributes – md-radio-group

The following table lists out the parameters and description of the different attributes of md-radio-group.

S.NO.	Parameter & Description
1	*** ng-model** Assignable angular expression to data-bind to.
2	**md-no-ink** Use of attribute indicates flag to disable ink ripple effects.

Attributes – md-radio-button

S.NO.	Parameter & Description
1	*** ng-model** Assignable angular expression to data-bind to.
2	*** ngValue** Angular expression which sets the value to which the expression should be set when selected.
3	*** value** The value to which the expression should be set when selected.

S.NO.	Parameter & Description
4	**ngChange** Angular expression to be executed when input changes due to user interaction with the input element.
5	**name** Property name of the form under which the control is published.
6	**aria-label** Adds label to radio button for accessibility. This defaults to radio button's text. If no text content is available, a warning will be logged.

Example

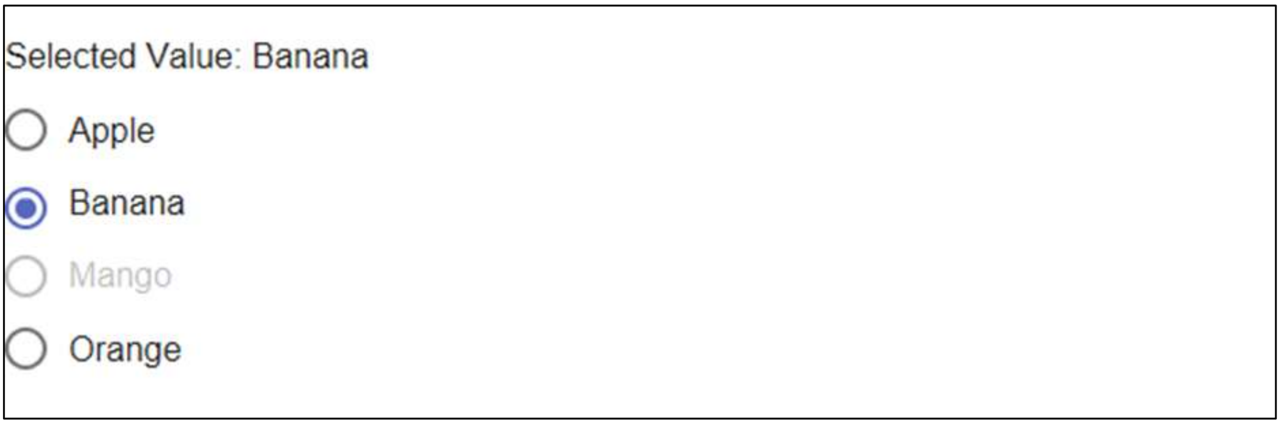
The following example shows the use of **md-radio-group** and **md-radio-button** directives and also the uses of radio buttons.

am_radiobuttons.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('radioButtonsController', radioButtonsController);


          function radioButtonsController ($scope) {
            $scope.radioData = [
               { label: 'Apple', value: 'Apple' },
               { label: 'Banana', value: 'Banana' },
               { label: 'Mango', value: 'Mango', isDisabled: true },
               { label: 'Orange', value: 'Orange' }
            ];
            $scope.group='Banana';
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="radioButtonsContainer" ng-controller="radioButtonsController as ctrl"
layout="column" ng-cloak>
         <p>Selected Value: <span>{{ group }}</span> </p>
         <md-radio-group ng-model="group" class="md-primary">
            <md-radio-button ng-repeat="d in radioData"
               ng-value="d.value"
               ng-disabled=" d.isDisabled "
               ng-class="{'md-align-top-left': $index==1}" >
               {{ d.label }}<br/>
            </md-radio-button>
         </md-radio-group>
      </div>
   </body>
</html>
```

Result

Verify the result.



Selected Value: Banana

○ Apple

● Banana

○ Mango

○ Orange

Angular Material – Selects

The **md-select**, an Angular directive is used to show Select box, bounded by ng-model.

Attributes – md-radio-group

The following table lists out the parameters and description of the different attributes of **md-radio-group**.

S.NO.	Parameter & Description
1	*** ng-model** The model!
2	**multiple** Whether it's multiple.
3	**md-on-close** Expression to be evaluated when the select is closed.
4	**placeholder** Placeholder hint text.
5	Optional label for accessibility. Only necessary if no placeholder or explicit label is present.
6	**md-container-class** Class list to get applied to the **.md-select-menu-container** element (for custom styling).

Example

The following example shows the use of md-select and also the uses of select box.

am_selects.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('selectController', selectController);

          function selectController ($scope) {
             $scope.users = [
               { id: 1, name: 'Mahesh' },
               { id: 2, name: 'Ramesh' },
               { id: 3, name: 'Sohan' }
            ];
            $scope.selectedUser = { id: 2, name: 'Ramesh' };
            $scope.drinks = [
               { category: 'hard', name: 'Beer' },
               { category: 'hard', name: 'Rum' },
               { category: 'hard', name: 'Vodka' },
               { category: 'hard', name: 'Whiski' },
               { category: 'soft', name: 'Coke' },
               { category: 'soft', name: 'Pepsi' },
               { category: 'soft', name: 'Pulp Maid' },
               { category: 'soft', name: 'Real Juice' }
            ];
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="selectContainer" ng-controller="selectController as ctrl"
layout="row" ng-cloak>
         <md-input-container>
            <label>User</label>
            <md-select ng-model="selectedUser">
               <md-option ng-value="user" ng-repeat="user in users">{{ user.name
}}</md-option>
            </md-select>
         </md-input-container>
         <md-input-container>
            <label>Drinks</label>
            <md-select ng-model="drink">
               <md-optgroup label="Hard Drinks">
                  <md-option ng-value="drink.name" ng-repeat="drink in drinks |
filter: {category: 'hard' }">{{drink.name}}</md-option>
               </md-optgroup>
               <md-optgroup label="Soft Drinks">
                  <md-option ng-value="drink.name" ng-repeat="drink in drinks |
filter: {category: 'soft' }">{{drink.name}}</md-option>
               </md-optgroup>
            </md-select>
         </md-input-container>
      </div>
   </body>
</html>
```

Result

Verify the result.

Angular Material – Fab Toolbars

The **md-fab-toolbar**, an Angular directive, is used to show a toolbar of elements or buttons for quick access to common actions.

Attributes

The following table lists out the parameters and description of the different attributes of **md-fab-toolbar**.

S.NO.	Parameter & Description
1	*** md-direction** This determines from which direction the toolbar items will appear relative to the trigger element. Supports left and right directions.
2	**md-open** Programmatically control whether or not the toolbar is visible.

Example

The following example shows the use of **md-fab-toolbar** directive and also the uses of toolbar.

am_fabtoolbar.htm

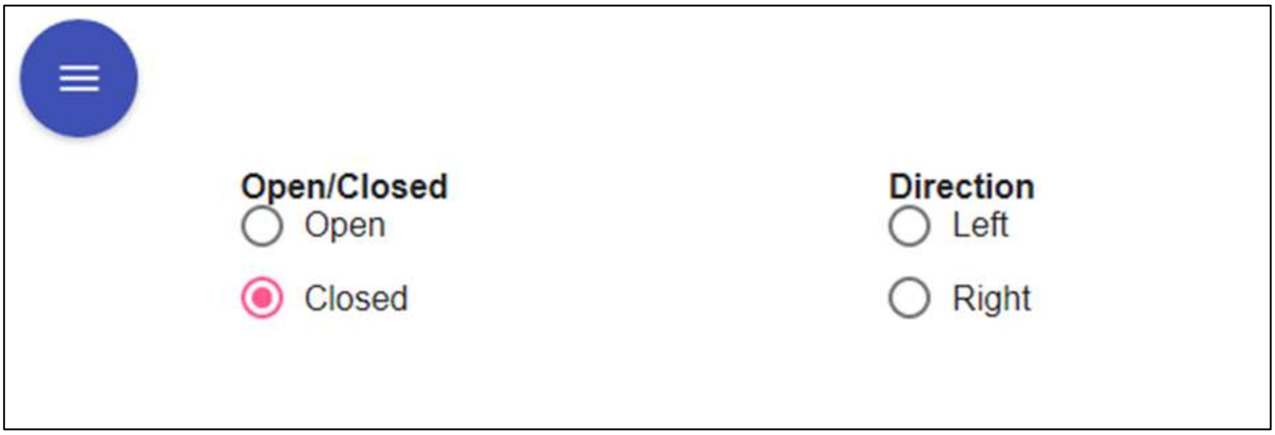
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
```

```
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('toolbarController', toolbarController);

         function toolbarController ($scope) {
            $scope.isOpen = false;
            $scope.count = 0;
                  $scope.selectedDirection = 'left';
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="toolbarContainer" ng-controller="toolbarController as ctrl" ng-cloak>
         <md-fab-toolbar md-open="ctrl.isOpen" md-
direction="{{ctrl.selectedDirection}}"
            count="ctrl.count">
            <md-fab-trigger class="align-with-text">
               <md-button aria-label="menu" class="md-fab md-primary">
                  <md-icon class="material-icons">menu</md-icon>
               </md-button>
            </md-fab-trigger>
            <md-toolbar>
               <md-fab-actions class="md-toolbar-tools">
                  <md-button aria-label="Add" class="md-fab md-raised md-mini md-accent">
                     <md-icon class="material-icons" aria-label="Add">add</md-icon>
                  </md-button>
                  <md-button aria-label="Insert Link" class="md-fab md-raised md-
mini md-accent">
```

```
                     <md-icon class="material-icons" aria-label="Insert
Link">insert_link</md-icon>
                  </md-button>
                  <md-button aria-label="Edit" class="md-fab md-raised md-mini md-
accent">
                     <md-icon class="material-icons" aria-
label="Edit">mode_edit</md-icon>
                  </md-button>
               </md-fab-actions>
            </md-toolbar>
         </md-fab-toolbar>
         <md-content class="md-padding" layout="column">
            <div layout="row" layout-align="space-around">
               <div layout="column">
                 <b>Open/Closed</b>
                 <md-radio-group ng-model="ctrl.isOpen">
                   <md-radio-button ng-value="true">Open</md-radio-button>
                   <md-radio-button ng-value="false">Closed</md-radio-button>
                 </md-radio-group>
               </div>
               <div layout="column">
                 <b>Direction</b>
                 <md-radio-group ng-model="ctrl.selectedDirection">
                   <md-radio-button ng-value="'left'">Left</md-radio-button>
                   <md-radio-button ng-value="'right'">Right</md-radio-button>
                 </md-radio-group>
               </div>
             </div>
         </md-content>
      </div>
   </body>
</html>
```

Result

Verify the result.



Angular Material – Sliders

The md-slider, an Angular directive is used to show a range component. It has two modes:

- **normal** - User can slide between wide range of values. This mode exists by default.
- **discrete** - User can slide between selected values. To enable discrete mode use md-discrete and step attributes.

Attributes

The following table lists out the parameters and description of the different attributes of the **md-slider**.

S.NO.	Parameter & Description
1	**md-discrete** This determines whether to enable discrete mode.
2	**step** The distance between values the user is allowed to pick. By default, it is 1.
3	**min** The minimum value the user is allowed to pick. By default, it is 0.
4	**max** The maximum value the user is allowed to pick. By default, it is 100.

Example

The following example shows the use of md-sidenav and also the uses of sidenav component.

am_sliders.htm

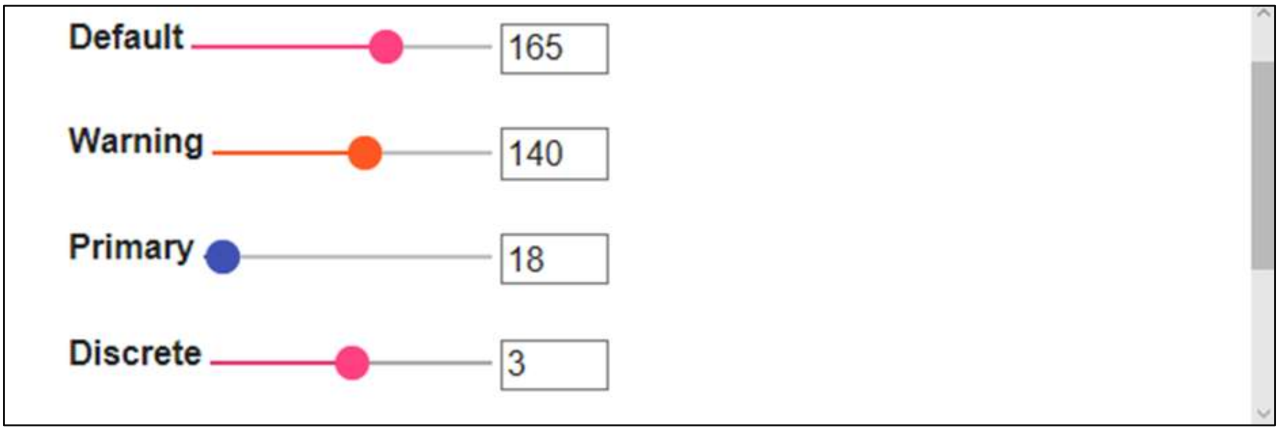
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('sliderController', sliderController);

          function sliderController ($scope, $mdSidenav) {
             $scope.color = {
                red: Math.floor(Math.random() * 255),
                green: Math.floor(Math.random() * 255),
                blue: Math.floor(Math.random() * 255)
             };
             $scope.rating = 3;
             $scope.disabled = 70;
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="sliderContainer" ng-controller="sliderController as ctrl"
layout="row" ng-cloak>
          <md-content style="margin: 16px; padding:16px">
```

```
            <div layout>
               <h4 style="margin-top:10px">Default</h4>
               <md-slider flex min="0" max="255" ng-model="color.red" aria-
label="red" id="red-slider" class></md-slider>
               <div flex="20" layout layout-align="center center">
                  <input flex type="number" ng-model="color.red" aria-label="red"
aria-controls="red-slider">
               </div>
            </div>
                <div layout>
               <h4 style="margin-top:10px">Warning</h4>
               <md-slider class="md-warn" flex min="0" max="255" ng-
model="color.green" aria-label="green" id="green-slider"></md-slider>
               <div flex="20" layout layout-align="center center">
                  <input flex type="number" ng-model="color.green" aria-
label="green" aria-controls="green-slider">
               </div>
            </div>
                  <div layout>
               <h4 style="margin-top:10px">Primary</h4>
               <md-slider class="md-primary" flex min="0" max="255" ng-
model="color.blue" aria-label="blue" id="blue-slider"></md-slider>
               <div flex="20" layout layout-align="center center">
                  <input flex type="number" ng-model="color.blue" aria-label="blue"
aria-controls="blue-slider">
               </div>
            </div>
                  <div layout>
               <h4 style="margin-top:10px">Discrete</h4>
               <md-slider flex md-discrete ng-model="rating" step="1" min="1"
max="5" aria-label="rating"></md-slider>
               <div flex="20" layout layout-align="center center">
                  <input flex type="number" ng-model="rating" aria-label="rating"
aria-controls="rating-slider">
               </div>
            </div>
                  <div layout>
               <h4 style="margin-top:10px">Disabled</h4>
```

```
                <md-slider flex min="0" max="255" ng-model="disabled" ng-disabled="true" aria-label="Disabled"></md-slider>
             </div>
             <div layout>
                <h4 style="margin-top:10px">Disabled, Discrete</h4>
                <md-slider flex md-discrete ng-model="rating" step="1" min="1" max="5" aria-label="disabled"  ng-disabled="true"></md-slider>
             </div>
              </md-content>
       </div>
   </body>
</html>
```

Result

Verify the result.



Angular Material – Tabs

The **md-tabs** and **md-tab Angular** directives are used to show tabs in the applcation. md-tabs is the grouping container for md-tab elements.

Attributes – md-tabs

The following table lists out the parameters and description of the different attributes of **md-tabs**.

S.NO.	Parameter & Description
1	**md-selected** Index of the active/selected tab.

S.NO.	Parameter & Description
2	**md-no-ink** If present, disables the ink ripple effects.
3	**md-no-ink-bar** If present, disables the selection ink bar.
4	**md-align-tabs** Attribute to indicate position of tab buttons: bottom or top; default is top.
5	**md-stretch-tabs** Attribute to indicate whether or not to stretch tabs: auto, always, or never; default is auto.
6	**md-dynamic-height** When enabled, the tab wrapper will resize based on the contents of the selected tab.
7	**md-center-tabs** When enabled, tabs will be centered provided there is no need for pagination.
8	**md-no-pagination** When enabled, pagination will remain off.
9	**md-swipe-content** When enabled, swipe gestures will be enabled for the content area to jump between tabs.
10	**md-enable-disconnect** When enabled, scopes will be disconnected for tabs that are not being displayed. This provides a performance boost, but may also cause unexpected issues and is not recommended for most users.
11	**md-autoselect** When present, any tabs added after the initial load will be automatically selected.

Attributes – md-tab

The following table lists out the parameters and description of the different attributes of **md-tab**.

S.NO.	Parameter & Description
1	**label** Optional attribute to specify a simple string as the tab label.
2	**ng-disabled** If present, disabled tab selection.
3	**md-on-deselect** Expression to be evaluated after the tab is de-selected.
4	**md-on-select** Expression to be evaluated after the tab is selected.
5	**md-active** When true, sets the active tab. **Note:** There can only be one active tab at a time.

Example

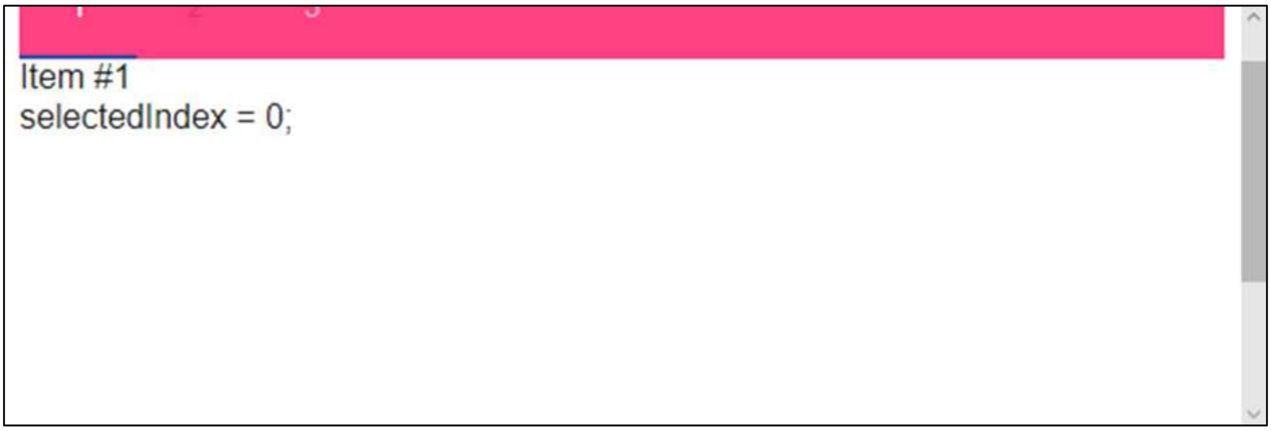
The following example shows the use of md-tabs and also the uses of tabs components.

am_tabs.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('tabController', tabController);

          function tabController ($scope) {
             $scope.data = {
                selectedIndex: 0,
                secondLocked:  true,
                secondLabel:   "2",
                bottom:        false
             };
             $scope.next = function() {
                $scope.data.selectedIndex = Math.min($scope.data.selectedIndex + 1, 2) ;
             };
             $scope.previous = function() {
                $scope.data.selectedIndex = Math.max($scope.data.selectedIndex - 1, 0);
             };
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="tabContainer" ng-controller="tabController as ctrl" ng-cloak>
         <md-content class="md-padding">
            <md-tabs class="md-accent" md-selected="data.selectedIndex" md-align-
tabs="{{data.bottom ? 'bottom' : 'top'}}">
               <md-tab id="tab1">
                  <md-tab-label>1</md-tab-label>
                  <md-tab-body>Item #1 <br/>selectedIndex = 0;</md-tab-body>
               </md-tab>
               <md-tab id="tab2" ng-disabled="data.secondLocked">
                  <md-tab-label>{{data.secondLabel}}</md-tab-label>
                  <md-tab-body>Item #2 <br/>selectedIndex = 1;</md-tab-body>
               </md-tab>
               <md-tab id="tab3">
                  <md-tab-label>3</md-tab-label>
                  <md-tab-body>Item #3<br/>selected Index = 2;</md-tab-body>
               </md-tab>
            </md-tabs>
         </md-content>
         <div class="md-padding" layout="row" layout-sm="column" layout-align="left
center" style="padding-top: 0;">
            <md-checkbox ng-model="data.secondLocked" aria-label="Disable tab 2?"
style="margin: 5px;">Disable tab 2?</md-checkbox>
            <md-checkbox ng-model="data.bottom" aria-label="Align tabs to bottom?"
style="margin: 5px;">Align tabs to bottom?</md-checkbox>
         </div>
      </div>
   </body>
</html>
```

Result

Verify the result.



Item #1
selectedIndex = 0;

Angular Material – Toolbars

The **md-toolbar**, an Angular directive is used to show a toolbar which is normally an area above the content to show the title and the relevant buttons.

Attributes

The following table lists out the parameters and description of the different attributes of the **md-toolbar**.

S.NO.	Parameter & Description
1	**md-scroll-shrink** This determines whether the header should shrink away as the user scrolls down, and reveal itself as the user scrolls up.

S.NO.	Parameter & Description
	for **scrollShrink** to work, the toolbar must be a sibling of an md-content element, placed before it. The **md-scroll-shrink** attribute is only parsed on component initialization, it does not watch for scope changes.
2	**md-shrink-speed-factor** How much to change the speed of the toolbar's shrinking by. For example, if 0.25 is given then the toolbar will shrink at one fourth the rate at which the user scrolls down. By default, it is 0.5.

Example

The following example shows the use of toolbar.

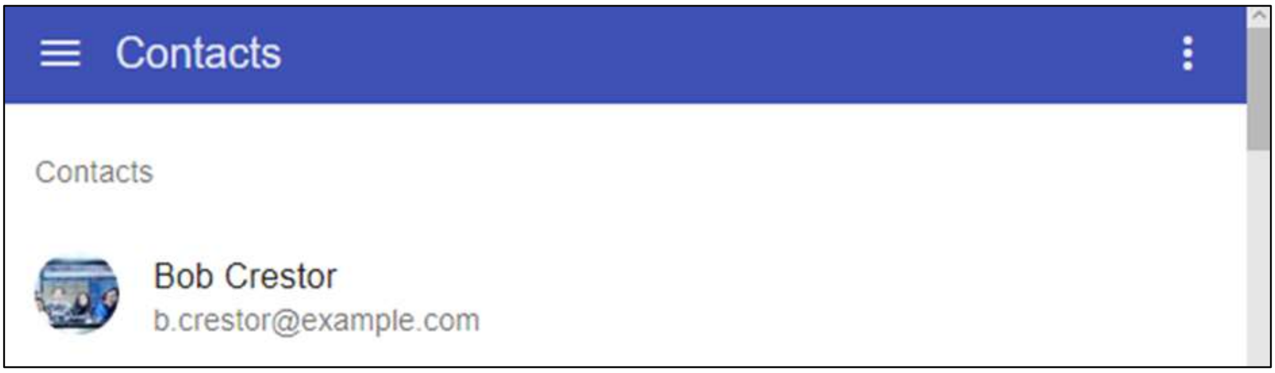
am_toolbar.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('toolbarController', toolbarController);


            function toolbarController ($scope) {
               var self = this;
               self.allContacts = loadContacts();
               self.contacts = [self.allContacts[0]];

               function loadContacts() {
                  var contacts = [
                     'Roberto Karlos',
                     'Bob Crestor',
                     'Nigel Rick',
                     'Narayana Garner'
                  ];
                  return contacts.map(function (c, index) {
                     var cParts = c.split(' ');
                     var contact = {
                        name: c,
                        email: cParts[0][0].toLowerCase() + '.' +
cParts[1].toLowerCase() + '@example.com',
                        image: 'http://lorempixel.com/50/50/people?' + index
                     };
                     contact._lowername = contact.name.toLowerCase();
                     return contact;
                  });
               }
            }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="toolbarContainer" ng-controller="toolbarController as ctrl"
layout="column" ng-cloak>
         <md-content>
            <md-toolbar md-scroll-shrink>
               <div class="md-toolbar-tools">
                  <md-button class="md-icon-button" aria-label="Settings">
                     <md-icon class="material-icons">menu</md-icon>
                  </md-button>
                  <h2>
                     <span>Contacts</span>
                  </h2>
                  <span flex></span>
                  <md-button class="md-icon-button" aria-label="More">
                     <md-icon class="material-icons">more_vert</md-icon>
                  </md-button>
               </div>
            </md-toolbar>
            <md-list>
               <md-subheader class="md-no-sticky">Contacts</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index, 
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar" 
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                  <md-divider ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
            <md-list>
               <md-subheader class="md-no-sticky">Contacts (With Insets)</md-
subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index, 
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar" 
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
                  <md-divider md-inset ng-if="!$last"></md-divider>
               </md-list-item>
            </md-list>
         </md-content>
      </div>
```

```
   </body>
</html>
```

Result

Verify the result.



Angular Material – Tooltips

Angular Material provides various special methods to show unobtrusive tooltips to the users. It provides ways to assign directions for them and the **md-tooltip** directive is used to show tooltips. A tooltip activates whenever the user focuses, hovers over, or touches the parent component.

Attributes

S.NO.	Parameter & Description
1	**md-visible** This is Boolean bound and determines whether the tooltip is currently visible.
2	**md-delay** How many milliseconds to wait to show the tooltip after the user focuses, hovers, or touches the parent. By default, it is 300ms.
3	**md-autohide** If present or provided with a boolean value, the tooltip will hide on mouse leave, regardless of focus.
4	**md-direction** This determines the direction in which the tooltip goes - supports left, right, top, and bottom. By default, the direction is towards bottom.

Example

The following example shows the use of tooltips.

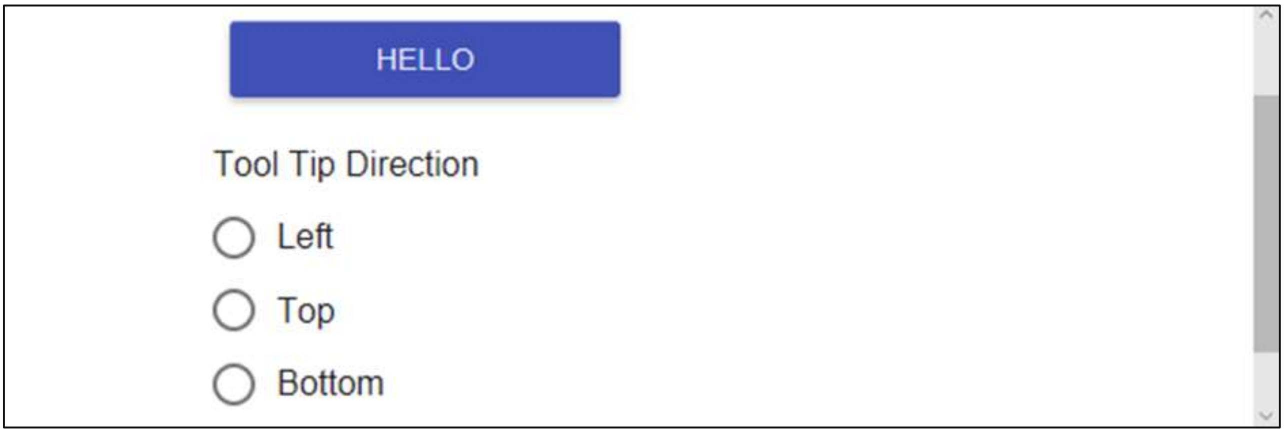
am_tooltips.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('tooltipController', tooltipController);

            function tooltipController ($scope) {
               $scope.demo = {
                  showTooltip : false,
                  tooltipDirection : ''
               };
            }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="tooltipContainer" ng-controller="tooltipController as ctrl"
layout="column" style="width:200px;margin-left: 100px; margin-right: 20px"  ng-
cloak>
         <br/><br/><br/>
         <md-button class="md-raised md-primary">Hello
            <md-tooltip md-visible="demo.showTooltip" md-
direction="{{demo.tooltipDirection}}">Hello World!</md-tooltip>
```

```
          </md-button>
          <p>Tool Tip Direction </p>
          <md-radio-group ng-model="demo.tooltipDirection" >
             <md-radio-button value="left">Left </md-radio-button>
             <md-radio-button value="top">Top</md-radio-button>
             <md-radio-button value="bottom"  class="md-primary">Bottom</md-radio-button>
             <md-radio-button value="right">Right</md-radio-button>
          </md-radio-group>
       </div>
    </body>
</html>
```

Result

Verify the result.



Angular Material – Chips

The **md-chips**, an Angular Directive, is used as a special component called Chip and can be used to represent a small set of information for example, a contact, tags etc. A custom template can be used to render the content of a chip. This can be achieved by specifying an **md-chip-template** element having the custom content as a child of md-chips.

Attributes

The following table lists out the parameters and description of the different attributes of **md-chips**.

S.NO.	Parameter & Description
1	*** ng-model** Assignable angular expression to data-bind to.
2	*** ng-model** A model to bind the list of items to.
3	*** md-transform-chip** An expression of form **myFunction($chip)** that when called expects one of the following return values: an object representing the $chip input string. undefined to simply add the $chip input string, or. null to prevent the chip from being appended.
4	*** md-require-match** If true, and the chips template contains an autocomplete, only allows selection of pre-defined chips (i.e. you cannot add new ones).
5	**placeholder** Placeholder text that will be forwarded to the input.
6	**secondary-placeholder** Placeholder text that will be forwarded to the input, displayed when there is at least one item in the list.
7	**readonly** Disables list manipulation (deleting or adding list items), hiding the input and the delete buttons.
8	**md-on-add** An expression which will be called when a chip has been added.

S.NO.	Parameter & Description
9	**md-on-remove** An expression which will be called when a chip has been removed.
10	**md-on-select** An expression which will be called when a chip is selected.
11	**delete-hint** A string read by screen readers instructing users that pressing the delete key will remove the chip.
12	**delete-button-label** A label for the **Delete** button. Also hidden and read by screen readers.
13	**md-separator-keys** An array of key codes used to separate chips.

Example

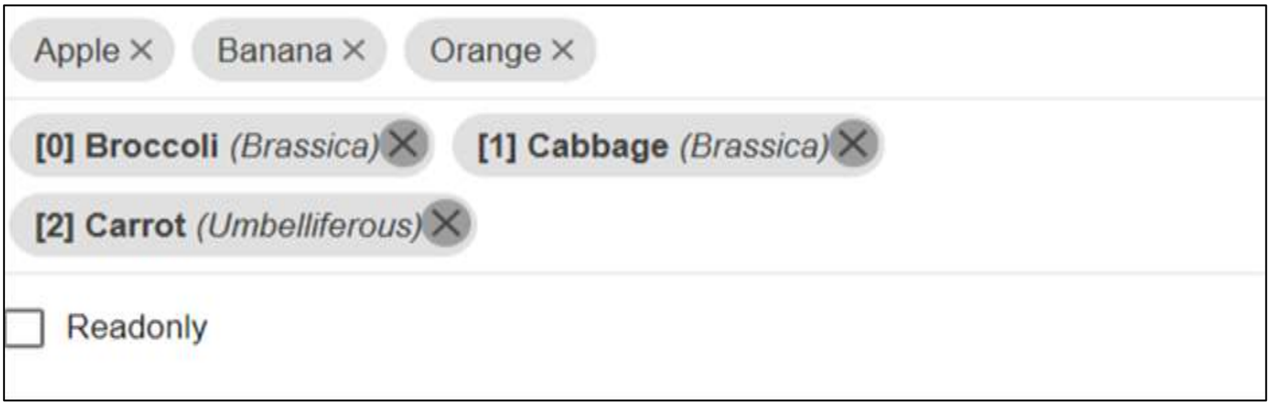
The following example shows the use of the **md-chips** directive and also the use of angular chips.

am_chips.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
         .closeButton {
            position: relative;
            height: 24px;
            width: 24px;
            line-height: 30px;
            text-align: center;
            background: rgba(0, 0, 0, 0.3);
            border-radius: 50%;
            border: none;
            box-shadow: none;
            padding: 0;
            margin: 3px;
            transition: background 0.15s linear;
            display: block;
         }
       </style>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('chipController', chipController);

         function chipController ($scope) {
            var self = this;
            self.readonly = false;
            // Lists of fruit names and Vegetable objects
            self.fruitNames = ['Apple', 'Banana', 'Orange'];
            self.roFruitNames = angular.copy(self.fruitNames);
            self.tags = [];
            self.vegObjs = [
               {
                  'name' : 'Broccoli',
                  'type' : 'Brassica'
               },
               {
                  'name' : 'Cabbage',
                  'type' : 'Brassica'
               },
               {
                  'name' : 'Carrot',
                  'type' : 'Umbelliferous'
               }
            ];
            self.newVeg = function(chip) {
               return {
                  name: chip,
                  type: 'unknown'
               };
            };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
<div ng-controller="chipController as ctrl" layout="column" ng-cloak>
<md-chips ng-model="ctrl.fruitNames" readonly="ctrl.readonly"></md-chips>
 <md-chips class="custom-chips" ng-model="ctrl.vegObjs" readonly="ctrl.readonly" md-transform-chip="ctrl.newVeg($chip)">
      <md-chip-template>
        <span>
          <strong>[{{$index}}] {{$chip.name}}</strong>
          <em>({{$chip.type}})</em>
        </span>
      </md-chip-template>
      <button md-chip-remove class="md-primary closeButton">
        <md-icon md-svg-icon="md-close"></md-icon>
      </button>
    </md-chips>
    <br/>
    <md-checkbox ng-model="ctrl.readonly">Readonly</md-checkbox>
     </div>
   </body>
</html>
```

Result

Verify the result.



Angular Material – Contact Chips

The **md-contact-chips**, an Angular Directive, is an input control built on md-chips and uses the **md-autocomplete** element. The contact chip component accepts a query expression which returns a list of possible contacts. The user can select one of these and add it to the list of availble chips.

Attributes

The following table lists out the parameters and description of the different attributes of **md-contact-chips**.

S.NO.	Parameter & Description
1	*** ng-model** A model to bind the list of items to.
2	*** md-contacts** An expression expected to return contacts matching the search test, $query.
3	*** md-contact-name** The field name of the contact object representing the contact's name.
4	*** md-contact-email** The field name of the contact object representing the contact's email address.
5	*** md-contact-image** The field name of the contact object representing the contact's image.

S.NO.	Parameter & Description
6	**placeholder** Placeholder text that will be forwarded to the input.
7	**secondary-placeholder** Placeholder text that will be forwarded to the input, displayed when there is at least on item in the list.
8	**filter-selected** Whether to filter selected contacts from the list of suggestions shown in the autocomplete.

Example

The following example shows the use of the md-contact-chips directive and also the use of angular contact chips.

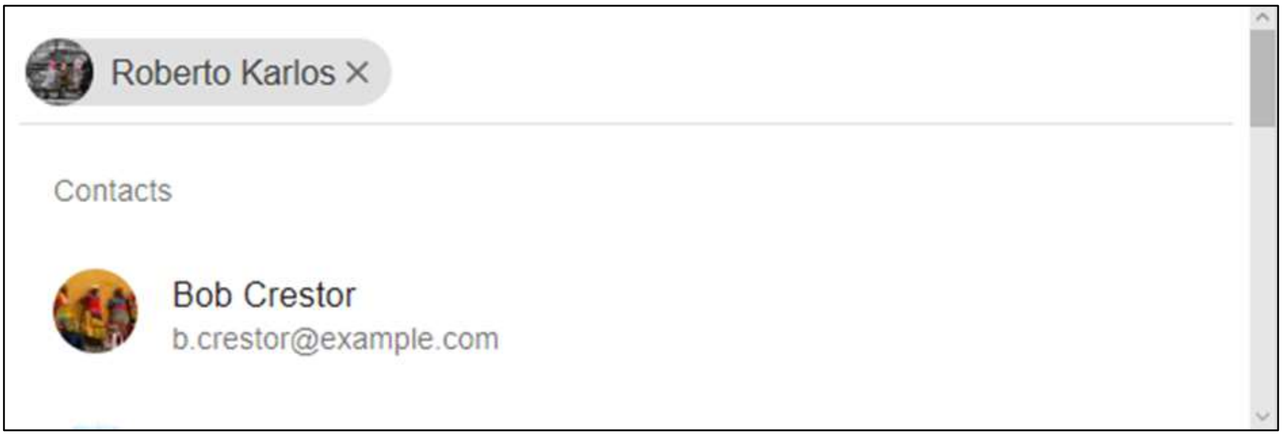
am_contact_chips.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('contactsChipController', contactsChipController);

         function contactsChipController ($scope) {
            var self = this;
            self.querySearch = querySearch;
            self.allContacts = loadContacts();
            self.contacts = [self.allContacts[0]];
            self.filterSelected = true;

            function querySearch (query) {
               var results = query ?
               self.allContacts.filter(createFilterFor(query)) : [];
               return results;
            }

            function createFilterFor(query) {
               var lowercaseQuery = angular.lowercase(query);
               return function filterFn(contact) {
                  return (contact._lowername.indexOf(lowercaseQuery) != -1);;
               };
            }

            function loadContacts() {
               var contacts = [
                  'Roberto Karlos',
                  'Bob Crestor',
                  'Nigel Rick',
                  'Narayana Garner',
                  'Anita Gros',
                  'Megan Smith',
                  'Tsvetko Metzger',
                  'Hector Simek',
                  'James Roody'
               ];
               return contacts.map(function (c, index) {
                     var cParts = c.split(' ');
                     var contact = {
                        name: c,
                        email: cParts[0][0].toLowerCase() + '.' +
cParts[1].toLowerCase() + '@example.com',
                        image: 'http://lorempixel.com/50/50/people?' + index
                     };
                     contact._lowername = contact.name.toLowerCase();
                     return contact;
               });
            }
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div ng-controller="contactsChipController as ctrl" layout="column" ng-cloak>
         <md-content class="md-padding autocomplete" layout="column">
            <md-contact-chips
               ng-model="ctrl.contacts"
               md-contacts="ctrl.querySearch($query)"
               md-contact-name="name"
               md-contact-image="image"
               md-contact-email="email"
               md-require-match="true"
               md-highlight-flags="i"
               filter-selected="ctrl.filterSelected"
               placeholder="To">
            </md-contact-chips>
            <md-list class="fixedRows">
               <md-subheader class="md-no-sticky">Contacts</md-subheader>
               <md-list-item class="md-2-line contact-item" ng-repeat="(index,
contact) in ctrl.allContacts"
                  ng-if="ctrl.contacts.indexOf(contact) < 0">
                  <img ng-src="{{contact.image}}" class="md-avatar"
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
               </md-list-item>
               <md-list-item class="md-2-line contact-item selected" ng-
repeat="(index, contact) in ctrl.contacts">
```

```
                  <img ng-src="{{contact.image}}" class="md-avatar"
alt="{{contact.name}}" />
                  <div class="md-list-item-text compact">
                     <h3>{{contact.name}}</h3>
                     <p>{{contact.email}}</p>
                  </div>
               </md-list-item>
            </md-list>
         </md-content>
     </div>
   </body>
</html>
```

Result

Verify the result.



Roberto Karlos ×

Contacts

Bob Crestor
b.crestor@example.com

7. Angular Material – Layouts

Layout Directive

Layout directive on a container element is used to specify the layout direction for its children. Following are the assignable values for the Layout Directive:

- **row** - Items are arranged horizontally with max-height = 100% and max-width is the width of the items in the container.

- **column** - Items are arranged vertically with max-width = 100% and max-height is the height of the items in the container.

For responsive design such as layout to be automatically changed depending upon the device screen size, the layout APIs listed in the following table can be used to set the layout direction for devices with view widths.

S.NO.	API & Device width when breakpoint overrides default
1	**layout** Sets default layout direction unless overridden by another breakpoint.
2	**layout-xs** width < 600px
3	**layout-gt-xs** width >= 600px
4	**layout-sm** 600px <= width < 960px
5	**layout-gt-sm** width >= 960px
6	**layout-md** 960px <= width < 1280px
7	**layout-gt-md** width >= 1280px
8	**layout-lg** 1280px <= width < 1920px
9	**layout-gt-lg** width >= 1920px

S.NO.	API & Device width when breakpoint overrides default
10	**layout-xl** width >= 1920px

Example

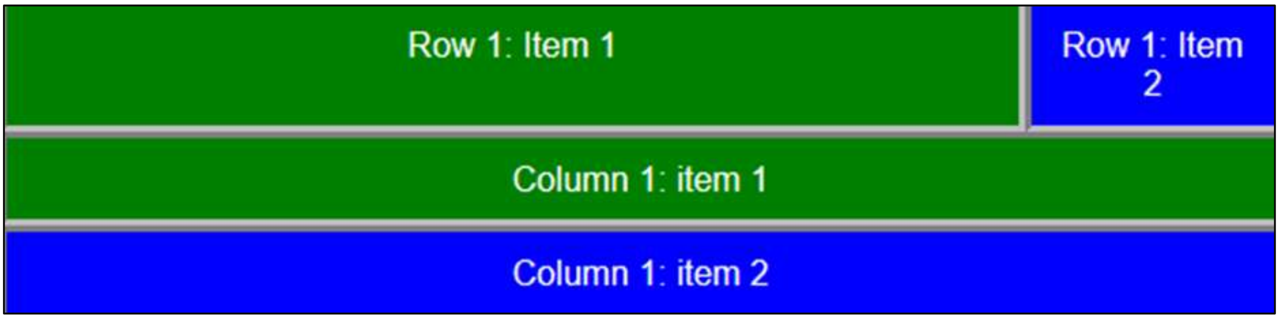
The following example shows the use of layout directive and also the uses of layout.

am_layouts.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
        .box {
              color:white;
              padding:10px;
              text-align:center;
              border-style: inset;
        }
        .green {
            background:green;
        }
        .blue {
            background:blue;
        }
       </style>
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('layoutController', layoutController);

          function layoutController ($scope) {
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="layoutContainer" ng-controller="layoutController as ctrl"
style="height:100px;" ng-cloak>
         <div layout="row" layout-xs="column">
            <div flex class="green box">Row 1: Item 1</div>
            <div flex="20" class="blue box">Row 1: Item 2</div>
         </div>
         <div layout="column" layout-xs="column">
            <div flex="33" class="green box">Column 1: item 1</div>
            <div flex="66" class="blue box">Column 1: item 2</div>
         </div>
      </div>
   </body>
</html>
```

Result

Verify the result.



Row 1: Item 1

Row 1: Item 2

Column 1: item 1

Column 1: item 2

Flex Directive

The flex directive on a container element is used to customize the size and position of the elements. It defines the way how the element is to adjust its size with respect to its parent container and the other elements within the container. Following are the assignable values for the flex directive:

- **multiples of 5** - 5, 10, 15 ... 100
- **33** - 33%
- **66** - 66%

Example

The following example shows the use of the flex directive and also the uses of flex.

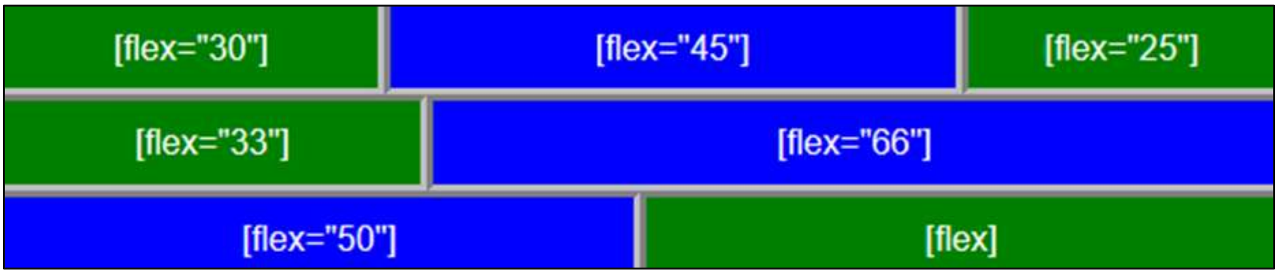
am_flex.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
        .box {
              color:white;
              padding:10px;
              text-align:center;
              border-style: inset;
        }
        .green {
            background:green;
        }
        .blue {
            background:blue;
        }
       </style>
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('layoutController', layoutController);

          function layoutController ($scope) {
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="layoutContainer" ng-controller="layoutController as ctrl"
layout="row" style="height:100px;" ng-cloak layout-wrap>
         <div flex="30" class="green box">
            [flex="30"]
         </div>
         <div flex="45" class="blue box">
            [flex="45"]
         </div>
         <div flex="25" class="green box">
            [flex="25"]
         </div>
         <div flex="33" class="green box">
            [flex="33"]
         </div>
         <div flex="66" class="blue box">
            [flex="66"]
         </div>
         <div flex="50" class="blue box">
            [flex="50"]
         </div>
         <div flex class="green box">
```

```
            [flex]
         </div>
      </div>
   </body>
</html>
```

Result

Verify the result.



8. Angular Material – Inputs

The **md-input-container**, an Angular directive, is a container component that contains any **<input>** or **<textarea>** element as a child. It also supports error handling using the standard ng-messages directives and animates the messages using the ngEnter/ngLeave events or the ngShow/ngHide events.

Attributes

The following table lists out the parameters and description of the different attributes of the **md-input-container**.

S.NO.	Parameter & Description
1	**md-maxlength** The maximum number of characters allowed in this input. If this is specified, a character counter will be shown underneath the input. The purpose of the md-maxlength is to show the max length counter text. If you don't want the counter text and only need "plain" validation, you can use the "simple" ng-maxlength or maxlength attributes.
2	**aria-label** Aria-label is required when no label is present. A warning message will be logged in the console if a label is not present.
3	**placeholder** An alternative approach to using aria-label when the label is not present. The placeholder text is copied to the aria-label attribute.
4	**md-no-autogrow** When present, the textareas will not grow automatically.
5	**md-detect-hidden** When present, the textareas will be sized properly when they are revealed after being hidden. This is off by default for performance reasons because it guarantees a reflow every digest cycle.

Example

The following example shows the use of the md-input-container directive and also the uses of inputs.

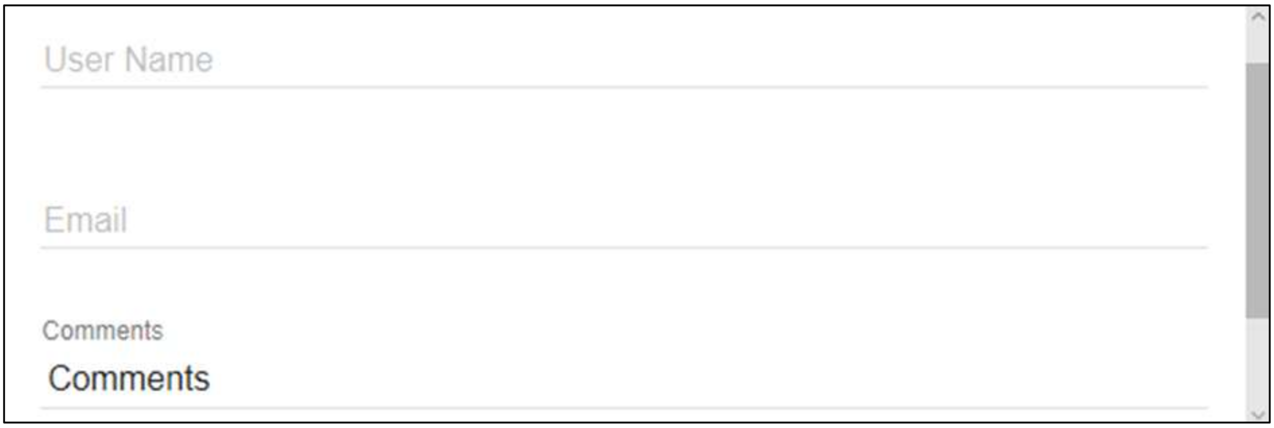
am_inputs.htm

```
<html lang="en" >
   <head>
```

```
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>

      </style>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('inputController', inputController);

         function inputController ($scope) {
           $scope.project = {
              comments: 'Comments',
           };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="inputContainer" class="inputDemo" ng-controller="inputController as 
ctrl" ng-cloak>
         <md-content layout-padding>
            <form name="projectForm">
               <md-input-container class="md-block">
                  <label>User Name</label>
                  <input required name="userName" ng-model="project.userName">
                  <div ng-messages="projectForm.userName.$error">
                     <div ng-message="required">This is required.</div>
                  </div>
               </md-input-container>
               <md-input-container class="md-block">
                  <label>Email</label>
                  <input required type="email" name="userEmail" ng-
model="project.userEmail"
                     minlength="10" maxlength="100" ng-pattern="/^.+@.+\..+$/" />
                  <div ng-messages="projectForm.userEmail.$error" role="alert">
                     <div ng-message-exp="['required', 'minlength', 'maxlength',
'pattern']">
                        Your email must be between 10 and 100 characters long and
should be a valid email address.
                     </div>
                  </div>
               </md-input-container>
               <md-input-container class="md-block">
                  <label>Comments</label>
                  <input md-maxlength="300" required name="comments" ng-
model="project.comments">
                  <div ng-messages="projectForm.comments.$error">
                     <div ng-message="required">This is required.</div>
                     <div ng-message="md-maxlength">The comments has to be less than
300 characters long.</div>
                  </div>
               </md-input-container>
            </form>
         </md-content>
      </div>
   </body>
</html>
```

Result

Verify the result.



9. Angular Material – Icons

The **md-icon**, an Angular directive, is a component to show vector-based icons in application. It supports icon fonts and SVG icons also apart from using the Google Material Icons.

Attributes

The following table lists out the parameters and description of the different attributes of **md-icon**.

S.NO.	Parameter & Description
1	*** md-font-icon** This is the string name of CSS icon associated with the font-face, which will be used to render the icon. Requires the fonts and the named CSS styles to be preloaded.
2	*** md-font-set** This is the CSS style name associated with the font library, which will be assigned as the class for the font-icon ligature. This value may also be an alias that is used to lookup the classname; internally use $mdIconProvider.fontSet(<alias>) to determine the style name.
3	*** md-svg-src** This is the String URL (or expression) used to load, cache, and display an external SVG.
4	*** md-svg-icon** This is the string name used for lookup of the icon from the internal cache; interpolated strings or expressions may also be used. Specific set names can be used with the syntax <set name>:<icon name>. To use icon sets, developers are required to pre-register the sets using the $mdIconProvider service.
5	**aria-label** This labels icon for accessibility. If an empty string is provided, icon will be hidden from accessibility layer with aria-hidden="true". If there's no aria-label on the icon nor a label on the parent element, a warning will be logged to the console.
6	**alt** This labels icon for accessibility. If an empty string is provided, icon will be hidden from accessibility layer with aria-hidden="true". If there's no alt on the icon nor a label on the parent element, a warning will be logged to the console.

Example

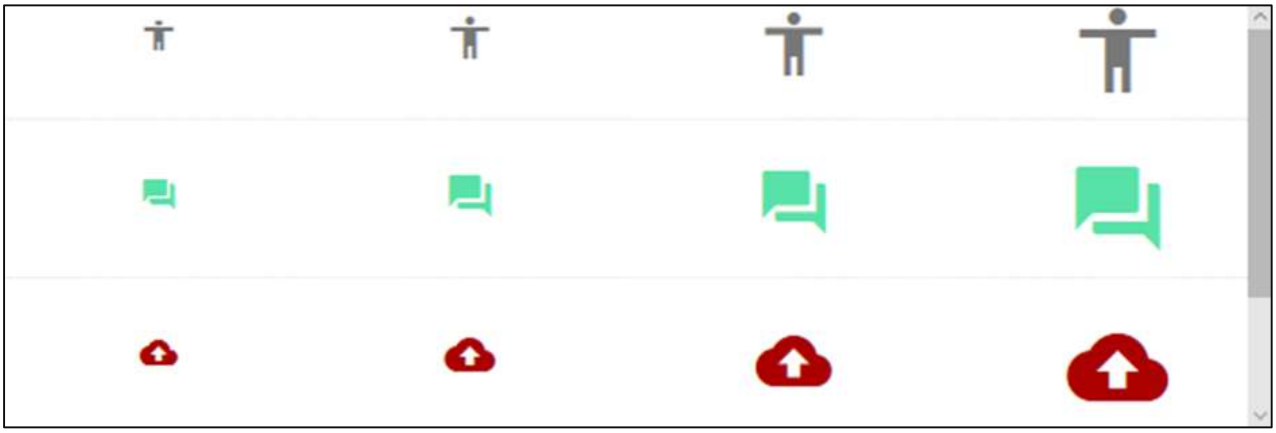
The following example shows the use of md-icons directive and also the uses of icons.

am_icons.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
         .iconDemo .glyph {
            border-bottom: 1px dotted #ccc;
            padding: 10px 0 20px;
            margin-bottom: 20px;
         }
         .iconDemo .preview-glyphs {
            display: flex;
            flex-direction: row;
         }
         .iconDemo .step {
            flex-grow: 1;
            line-height: 0.5;
         }
         .iconDemo .material-icons.md-18 {
            font-size: 18px;
         }
         .iconDemo .material-icons.md-24 {
            font-size: 24px;
         }
         .iconDemo .material-icons.md-36 {
            font-size: 36px;
         }
         .iconDemo .material-icons.md-48 {
            font-size: 48px;
         }
         .iconDemo .material-icons.md-dark {
            color: rgba(0, 0, 0, 0.54);
         }
         .iconDemo .material-icons.md-dark.md-inactive {
            color: rgba(0, 0, 0, 0.26);
         }
         .iconDemo .material-icons.md-light {
            color: white;
         }
         .iconDemo .material-icons.md-light.md-inactive {
            color: rgba(255, 255, 255, 0.3);
         }
      </style>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('iconController', iconController);

         function iconController ($scope) {
            var iconData = [
               {name: 'accessibility'  , color: "#777" },
               {name: 'question_answer', color: "rgb(89, 226, 168)" },
               {name: 'backup'         , color: "#A00" },
               {name: 'email'          , color: "#00A" }
            ];
            $scope.fonts = [].concat(iconData);
            $scope.sizes = [
               {size:"md-18",padding:0},
               {size:"md-24",padding:2},
               {size:"md-36",padding:6},
               {size:"md-48",padding:10}
            ];
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="iconContainer" class="iconDemo" ng-controller="iconController as
ctrl" ng-cloak>
         <div class="glyph" ng-repeat="font in fonts" layout="row">
            <div ng-repeat="it in sizes" flex layout-align="center center"
style="text-align: center;" layout="column">
            <div flex></div>
               <div class="preview-glyphs">
                  <md-icon ng-style="{color: font.color}"
                     aria-label="{{ font.name }}"
                     class="material-icons step"
                     ng-class="it.size">
                     {{ font.name }}
                  </md-icon>
               </div>
            </div>
         </div>
      </div>
   </body>
</html>
```

Result

Verify the result.



10. Angular Material – Grids

The **md-grid-list**, an Angular directive, is a component for laying out content for varying screen sizes. A grid has 12 columns in the desktop size screen, 8 in the tablet size screen, and 4 in the phone size screen, where each size have predefined margins and gutters. Cells are laid out in sequential manner in a row, in the order they are defined.

Attributes

The following table lists out the parameters and description of the different attributes of **md-grid-list**.

S.NO.	Parameter & Description
1	*** md-cols** This is for the number of columns in the grid.
2	*** md-row-height** One of CSS length - Fixed height rows (eg. 8px or 1rem). {width}:{height} - Ratio of width to height (eg. md-row-height="16:9"). "fit" - Height will be determined by subdividing the available height by the number of rows.
3	**md-gutter** The amount of space between tiles in CSS units (default 1px).
4	**md-on-layout** Expression to evaluate after layout. Event object is available as $event, and contains performance information.

Example

The following example shows the use of the **md-grid-list** directive and also the uses of grid.

am_grid.htm

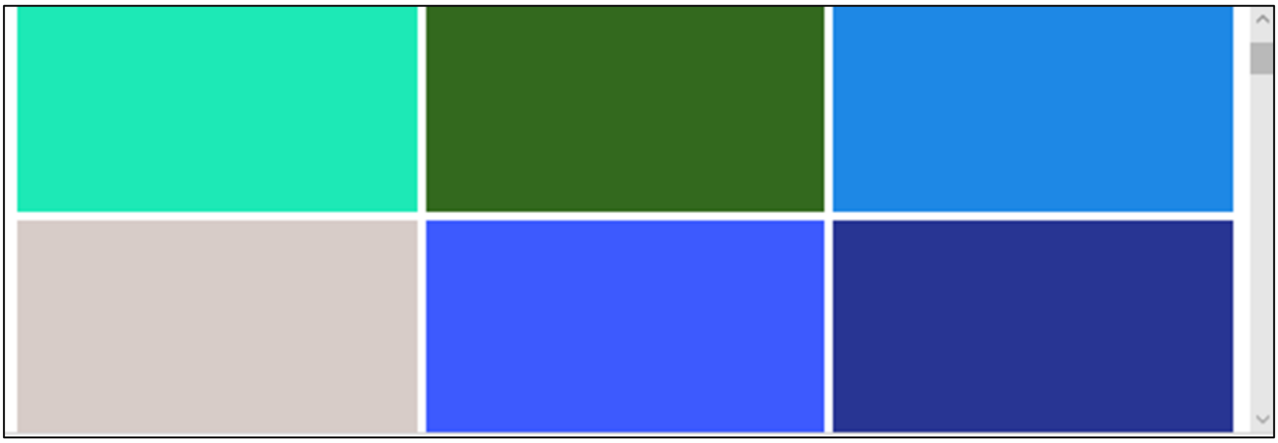
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
```

```
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('gridController', gridController);


         function gridController ($scope) {
            var COLORS = ['#ffebee', '#ffcdd2', '#ef9a9a', '#e57373', '#ef5350',
'#f44336', '#e53935', '#d32f2f', '#c62828', '#b71c1c', '#ff8a80',
'#ff5252','#ff1744', '#d50000', '#f8bbd0', '#f48fb1', '#f06292', '#ec407a',
'#e91e63', '#d81b60', '#c2185b', '#ad1457', '#880e4f', '#ff80ab', '#ff4081',
'#f50057', '#c51162', '#e1bee7', '#ce93d8', '#ba68c8', '#ab47bc', '#9c27b0',
'#8e24aa', '#7b1fa2', '#4a148c', '#ea80fc', '#e040fb', '#d500f9', '#aa00ff',
'#ede7f6', '#d1c4e9', '#b39ddb', '#9575cd', '#7e57c2', '#673ab7', '#5e35b1',
'#4527a0', '#311b92', '#b388ff', '#7c4dff', '#651fff', '#6200ea', '#c5cae9',
'#9fa8da', '#7986cb', '#5c6bc0', '#3f51b5', '#3949ab', '#303f9f', '#283593',
'#1a237e', '#8c9eff', '#536dfe', '#3d5afe', '#304ffe', '#e3f2fd', '#bbdefb',
'#90caf9', '#64b5f6', '#42a5f5', '#2196f3', '#1e88e5', '#1976d2', '#1565c0',
'#0d47a1', '#82b1ff', '#448aff', '#2979ff', '#2962ff', '#b3e5fc', '#81d4fa',
'#4fc3f7', '#29b6f6', '#03a9f4', '#039be5', '#0288d1', '#0277bd', '#01579b',
'#80d8ff', '#40c4ff', '#00b0ff', '#0091ea', '#e0f7fa', '#b2ebf2', '#80deea',
'#4dd0e1', '#26c6da', '#00bcd4', '#00acc1', '#0097a7', '#00838f', '#006064',
'#84ffff', '#18ffff', '#00e5ff', '#00b8d4', '#e0f2f1', '#b2dfdb', '#80cbc4',
'#4db6ac', '#26a69a', '#009688', '#00897b', '#00796b', '#00695c', '#a7ffeb',
'#64ffda', '#1de9b6', '#00bfa5', '#e8f5e9', '#c8e6c9', '#a5d6a7', '#81c784',
'#66bb6a', '#4caf50', '#43a047', '#388e3c', '#2e7d32', '#1b5e20', '#b9f6ca',
'#69f0ae', '#00e676', '#00c853', '#f1f8e9', '#dcedc8', '#c5e1a5', '#aed581',
'#9ccc65', '#8bc34a', '#7cb342', '#689f38', '#558b2f', '#33691e', '#ccff90',
'#b2ff59', '#76ff03', '#64dd17', '#f9fbe7', '#f0f4c3', '#e6ee9c', '#dce775',
'#d4e157', '#cddc39', '#c0ca33', '#afb42b', '#9e9d24', '#827717', '#f4ff81',
'#eeff41', '#c6ff00', '#aeea00', '#fffde7', '#fff9c4', '#fff59d', '#fff176',
'#ffee58', '#ffeb3b', '#fdd835', '#fbc02d', '#f9a825', '#f57f17', '#ffff8d',
'#ffff00', '#ffea00', '#ffd600', '#fff8e1', '#ffecb3', '#ffe082', '#ffd54f',
'#ffca28', '#ffc107', '#ffb300', '#ffa000', '#ff8f00', '#ff6f00', '#ffe57f',
'#ffd740', '#ffc400', '#ffab00', '#fff3e0', '#ffe0b2', '#ffcc80', '#ffb74d',
'#ffa726', '#ff9800', '#fb8c00', '#f57c00', '#ef6c00', '#e65100', '#ffd180',
'#ffab40', '#ff9100', '#ff6d00', '#fbe9e7', '#ffccbc', '#ffab91', '#ff8a65',
'#ff7043', '#ff5722', '#f4511e', '#e64a19', '#d84315', '#bf360c', '#ff9e80',
'#ff6e40', '#ff3d00', '#dd2c00', '#d7ccc8', '#bcaaa4', '#795548', '#d7ccc8',
'#bcaaa4', '#8d6e63', '#eceff1', '#cfd8dc', '#b0bec5', '#90a4ae', '#78909c',
'#607d8b', '#546e7a', '#cfd8dc', '#b0bec5', '#78909c'];
            this.colorTiles = (function() {
               var tiles = [];
               for (var i = 0; i < 46; i++) {
                  tiles.push({
                     color: randomColor(),
                     colspan: randomSpan(),
                     rowspan: randomSpan()
                  });
               }
               return tiles;
            })();

            function randomColor() {
               return COLORS[Math.floor(Math.random() * COLORS.length)];
            }

            function randomSpan() {
               var r = Math.random();
               if (r < 0.8) {
                  return 1;
               } else if (r < 0.9) {
                  return 2;
               } else {
                  return 3;
               }
            }
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="gridContainer" ng-controller="gridController as ctrl" ng-cloak>
```

```
          <md-content layout-padding>
             <md-grid-list
                md-cols-gt-md="12" md-cols-sm="3" md-cols-md="8"
                md-row-height-gt-md="1:1" md-row-height="4:3"
                md-gutter-gt-md="16px" md-gutter-gt-sm="8px" md-gutter="4px">
                <md-grid-tile
                   ng-repeat="tile in ctrl.colorTiles"
                   ng-style="{
                      'background': tile.color
                   }"
                   md-colspan-gt-sm="{{tile.colspan}}"
                   md-rowspan-gt-sm="{{tile.rowspan}}">
                </md-grid-tile>
             </md-grid-list>
          </md-content>
       </div>
    </body>
</html>
```

Result

Verify the result.



Resize the screen to see the effect.

11. Angular Material – SideNav

The **md-sidenav**, an Angular directive is used to show a container component which can be shown or hide programmatically. It slides out over the top of the main content region by default.

Attributes

The following table lists out the parameters and description of the different attributes of **md-sidenav**,

S.NO.	Parameter & Description
1	**md-is-open** A model bound to whether the sidenav is opened.
2	**md-component-id** componentId to use with $mdSidenav service.
3	**md-is-locked-open** When this expression evalutes to true, the sidenav 'locks open': it falls into the content's flow instead of appearing over it. This overrides the **is-open** attribute. The **$mdMedia()** service is exposed to the **is-locked-open** attribute, which can be given a media query or one of the sm, gt-sm, md, gt-md, lg or gt-lg presets. Examples: <md-sidenav md-is-locked-open="shouldLockOpen"></md-sidenav> <md-sidenav md-is-locked-open="$mdMedia('min-width: 1000px')"></md-sidenav> <md-sidenav md-is-locked-open="$mdMedia('sm')"></md-sidenav> (locks open on small screens)

Example

The following example shows the use of **md-sidenav** and also the uses of the **sidenav** component.

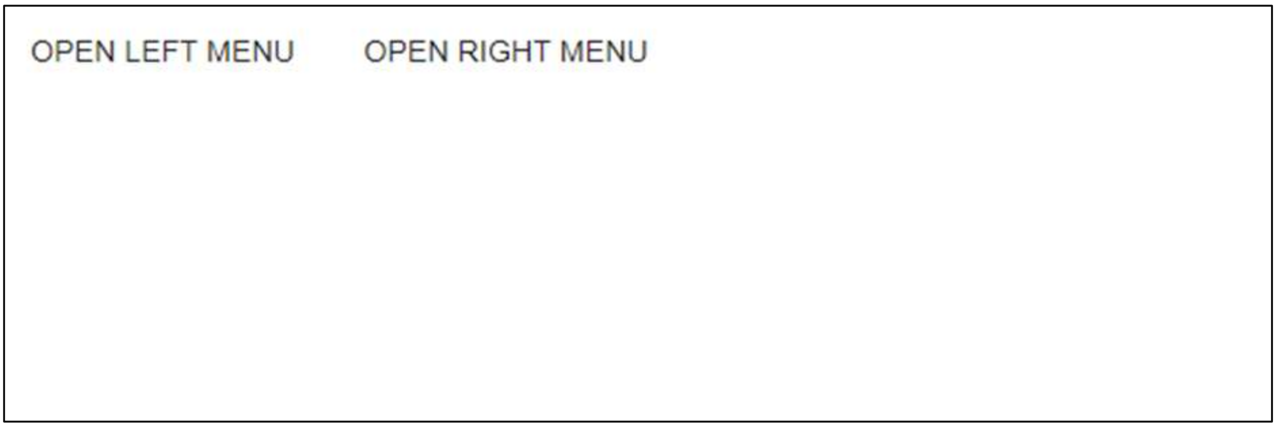
am_sidenav.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
```

```
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('sideNavController', sideNavController);

          function sideNavController ($scope, $mdSidenav) {
             $scope.openLeftMenu = function() {
               $mdSidenav('left').toggle();
             };
                   $scope.openRightMenu = function() {
               $mdSidenav('right').toggle();
             };
         }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="sideNavContainer" ng-controller="sideNavController as ctrl"
layout="row" ng-cloak>
         <md-sidenav md-component-id="left" class="md-sidenav-left">Welcome to
TutorialsPoint.Com</md-sidenav>
         <md-content>
            <md-button ng-click="openLeftMenu()">Open Left Menu</md-button>
                  <md-button ng-click="openRightMenu()">Open Right Menu</md-button>
         </md-content>
         <md-sidenav md-component-id="right" class="md-sidenav-right">
            <md-button href="http://google.com">Google</md-button>
         </md-sidenav>
      </div>
   </body>
</html>
```

Result

Verify the result.



12. Angular Material – Fab Speed Dial

The **md-fab-speed-dial**, an Angular directive, is used to show a series of popup elements or buttons for quick access to common actions.

Attributes

The following table lists out the parameters and description of the different attributes of **md-fab-speed-dial**.

S.NO.	Parameter & Description
1	*** md-direction** This determines the direction at which the speed dial is to appear relative to the trigger element.
2	**md-open** This helps programmatically control whether or not the speed-dial is visible.

Example

The following example shows the use of md-fab-speed-dial directive and also the uses of speed dial.

am_speeddial.htm

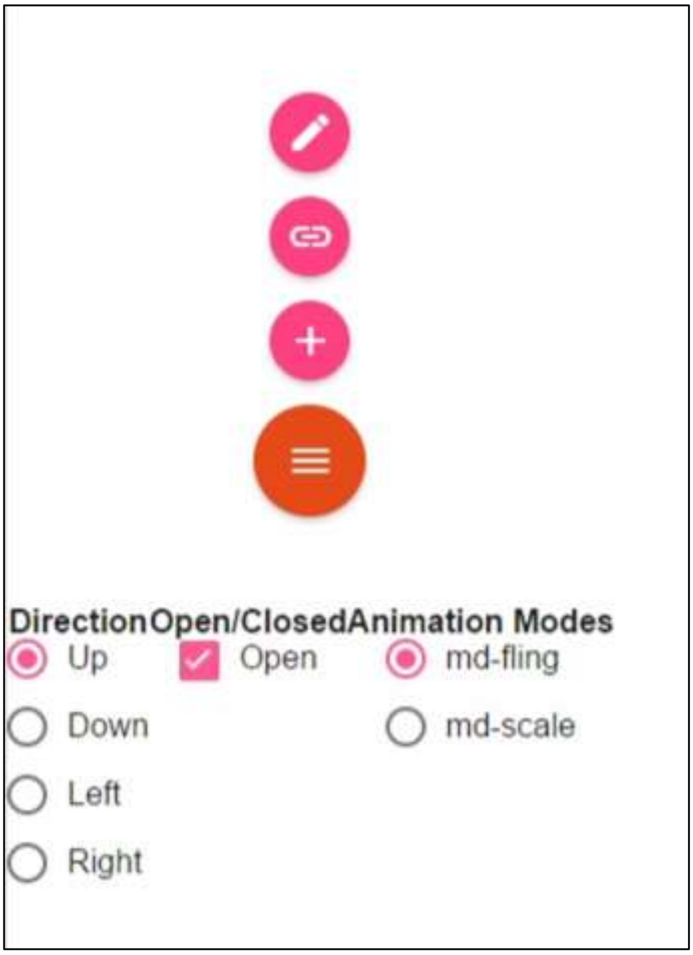
```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
```

```
   <style>
   .fabSpeedDial .text-capitalize {
      text-transform: capitalize;
   }
   .fabSpeedDial .md-fab:hover, .fabSpeedDialdemoBasicUsage .md-fab.md-focused {
      background-color: #000 !important;
   }
   .fabSpeedDial p.note {
      font-size: 1.2rem;
   }
   .fabSpeedDial .lock-size {
      min-width: 300px;
      min-height: 300px;
      width: 300px;
      height: 300px;
      margin-left: auto;
      margin-right: auto;
   }
   </style>
   <script language="javascript">
      angular
         .module('firstApplication', ['ngMaterial'])
         .controller('speedDialController', speedDialController);

      function speedDialController ($scope) {
         this.topDirections = ['left', 'up'];
         this.bottomDirections = ['down', 'right'];
         this.isOpen = false;
         this.availableModes = ['md-fling', 'md-scale'];
         this.selectedMode = 'md-fling';
         this.availableDirections = ['up', 'down', 'left', 'right'];
         this.selectedDirection = 'up';
      }
   </script>
</head>
<body ng-app="firstApplication">
```

```
      <div class="fabSpeedDial" id="speedDialContainer" ng-
controller="speedDialController as ctrl" layout="row" ng-cloak>
         <md-content>
            <div class="lock-size" layout="row" layout-align="center center">
               <md-fab-speed-dial md-open="ctrl.isOpen" md-
direction="{{ctrl.selectedDirection}}"
                  ng-class="ctrl.selectedMode">
                  <md-fab-trigger>
                     <md-button aria-label="menu" class="md-fab md-warn">
                        <md-icon class="material-icons">menu</md-icon>
                     </md-button>
                  </md-fab-trigger>
                  <md-fab-actions>
                     <md-button aria-label="Add" class="md-fab md-raised md-mini md-
accent">
                        <md-icon class="material-icons" aria-label="Add">add</md-
icon>
                     </md-button>
                     <md-button aria-label="Insert Link" class="md-fab md-raised md-
mini md-accent">
                        <md-icon class="material-icons" aria-label="Insert
Link">insert_link</md-icon>
                     </md-button>
                     <md-button aria-label="Edit" class="md-fab md-raised md-mini
md-accent">
                        <md-icon class="material-icons" aria-
label="Edit">mode_edit</md-icon>
                     </md-button>
                  </md-fab-actions>
               </md-fab-speed-dial>
            </div>
            <div layout="row" layout-align="space-around">
               <div layout="column" layout-align="start center">
                  <b>Direction</b>
                  <md-radio-group ng-model="ctrl.selectedDirection">
                     <md-radio-button ng-repeat="direction in
ctrl.availableDirections"
                        ng-value="direction" class="text-capitalize">
                        {{direction}}
                     </md-radio-button>
                  </md-radio-group>
               </div>
               <div layout="column" layout-align="start center">
                  <b>Open/Closed</b>
                  <md-checkbox ng-model="ctrl.isOpen">
                     Open
                  </md-checkbox>
               </div>
               <div layout="column" layout-align="start center">
                  <b>Animation Modes</b>
                  <md-radio-group ng-model="ctrl.selectedMode">
                     <md-radio-button ng-repeat="mode in ctrl.availableModes" ng-
value="mode">
                        {{mode}}
                     </md-radio-button>
                  </md-radio-group>
               </div>
            </div>
         </md-content>
        </div>
   </body>
</html>
```

Result

Verify the result.



Direction Open/Closed Animation Modes

● Up ☑ Open ● md-fling

○ Down ○ md-scale

○ Left

○ Right

13. Angular Material – Subheaders

The **md-subheader**, an Angular directive, is used to show a subheader for a section.

Example

The following example shows the use of md-subheader and also the uses of subheader component.

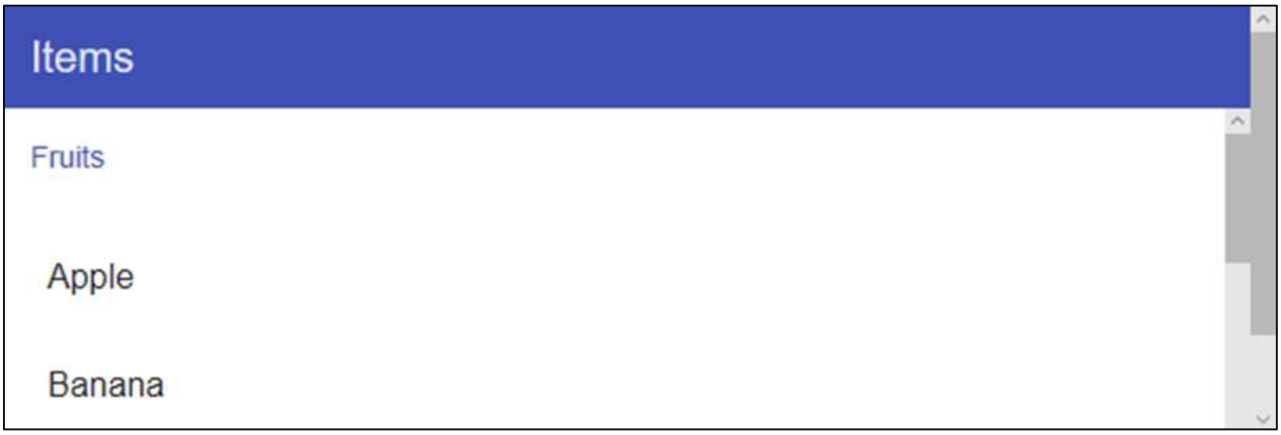
am_subheaders.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('subheaderController', subheaderController);

          function subheaderController ($scope) {
             $scope.fruitNames = ['Apple', 'Banana', 'Orange'];
                   $scope.vegNames = ['Carrot', 'Potato', 'Cabbage'];
                   $scope.eateries = ['Milk', 'Bread'];
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="subheaderContainer" ng-controller="subheaderController as ctrl"
layout="column" flex layout-fill ng-cloak>
         <md-toolbar md-scroll-shrink>
            <div class="md-toolbar-tools">Items</div>
         </md-toolbar>
          <md-content style="height: 600px;">
            <section>
               <md-subheader class="md-primary">Fruits</md-subheader>
                  <md-list layout-padding>
                     <md-list-item ng-repeat="fruits in fruitNames">
                        <div>
                           <p>{{fruits}}</p>
                        </div>
                     </md-list-item>
                  </md-list>
               </section>
               <section>
               <md-subheader class="md-warn">Vegetables</md-subheader>
                  <md-list layout-padding>
                     <md-list-item ng-repeat="veg in vegNames">
                        <div>
                           <p>{{veg}}</p>
                        </div>
                     </md-list-item>
                  </md-list>
               </section>
               <section>
               <md-subheader>Eateries</md-subheader>
                  <md-list layout-padding>
                     <md-list-item ng-repeat="eatery in eateries">
                        <div>
                           <p>{{eatery}}</p>
                        </div>
                     </md-list-item>
                  </md-list>
```

```
                </section>
              </md-content>
        </div>
    </body>
</html>
```

Result

Verify the result.



Items

Fruits

Apple

Banana

14. Angular Material – Swipe

The Swipe functionality is meant for mobile devices. The following directives are used to handle swipes.

- **md-swipe-down**, an Angular directive is used to specify custom behavior when an element is swiped down.

- **md-swipe-left**, an Angular directive is used to specify custom behavior when an element is swiped left.

- **md-swipe-right**, an Angular directive is used to specify custom behavior when an element is swiped right.

- **md-swipe-up**, an Angular directive is used to specify custom behavior when an element is swiped up.

Example

The following example shows the use of md-swipe-* and also the uses of swipe components.

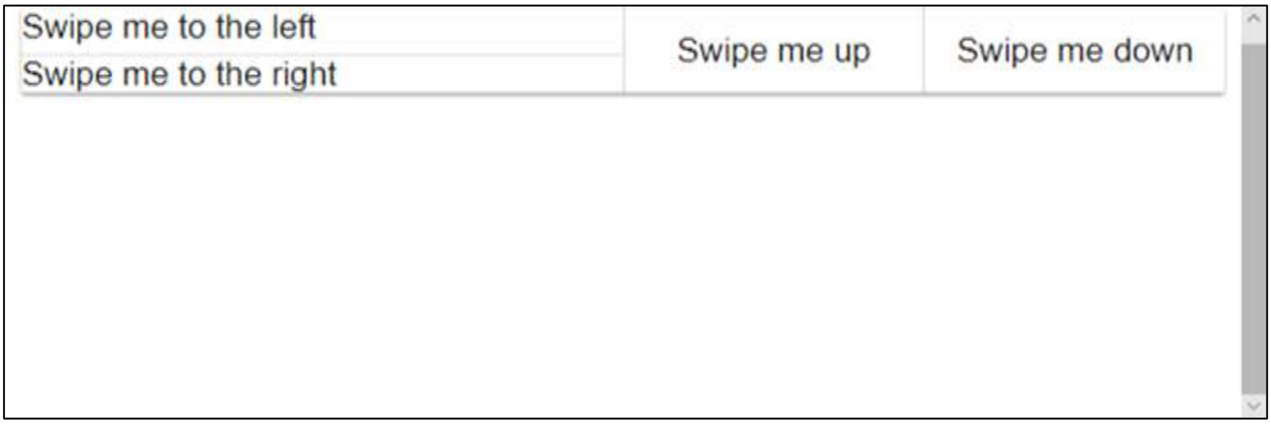
am_swipes.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
         .swipeContainer .demo-swipe {
            padding: 20px 10px;
         }
         .swipeContainer .no-select {
            -webkit-touch-callout: none;
            -webkit-user-select: none;
            -khtml-user-select: none;
            -moz-user-select: none;
            -ms-user-select: none;
            user-select: none;
         }
       </style>
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('swipeController', swipeController);

          function swipeController ($scope) {
             $scope.onSwipeLeft = function(ev) {
                alert('Swiped Left!');
             };
             $scope.onSwipeRight = function(ev) {
                alert('Swiped Right!');
             };
             $scope.onSwipeUp = function(ev) {
                alert('Swiped Up!');
             };
             $scope.onSwipeDown = function(ev) {
                alert('Swiped Down!');
             };
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <md-card>
         <div id="swipeContainer" ng-controller="swipeController as ctrl"
layout="row" ng-cloak>
            <div flex>
               <div class="demo-swipe" md-swipe-left="onSwipeLeft()">
```

```
                  <span class="no-select">Swipe me to the left</span>
                  <md-icon md-font-icon="android" aria-label="android"></md-icon>
               </div>
               <md-divider></md-divider>
               <div class="demo-swipe" md-swipe-right="onSwipeRight()">
                  <span class="no-select">Swipe me to the right</span>
               </div>
            </div>
            <md-divider></md-divider>
            <div flex layout="row">
               <div flex layout="row" layout-align="center center"
                  class="demo-swipe" md-swipe-up="onSwipeUp()">
                  <span class="no-select">Swipe me up</span>
               </div>
               <md-divider></md-divider>
               <div flex layout="row" layout-align="center center"
                  class="demo-swipe" md-swipe-down="onSwipeDown()">
                  <span class="no-select">Swipe me down</span>
               </div>
            </div>
         </div>
      </md-card>
   </body>
</html>
```

Result

Verify the result.



15. Angular Material – Switches

The **md-switch**, an Angular directive, is used to show a switch.

Attributes

The following table lists out the parameters and description of the different attributes of **md-switch**.

S.NO.	Parameter & Description
1	*** ng-model** The assignable angular expression to data-bind to.
2	**name** The property name of the form under which the control is published.
3	**ng-true-value** The value to which the expression should be set when selected.
4	**ng-false-value** The value to which the expression should be set when not selected.
5	**ng-change** The Angular expression to be executed when input changes due to user interaction with the input element.
6	**ng-disabled** En/Disable based on the expression.
7	**md-no-ink** The use of attribute indicates the use of ripple ink effects.
8	**aria-label** This publishes the button label used by the screen-readers for accessibility. This defaults to the switch's text.

Example

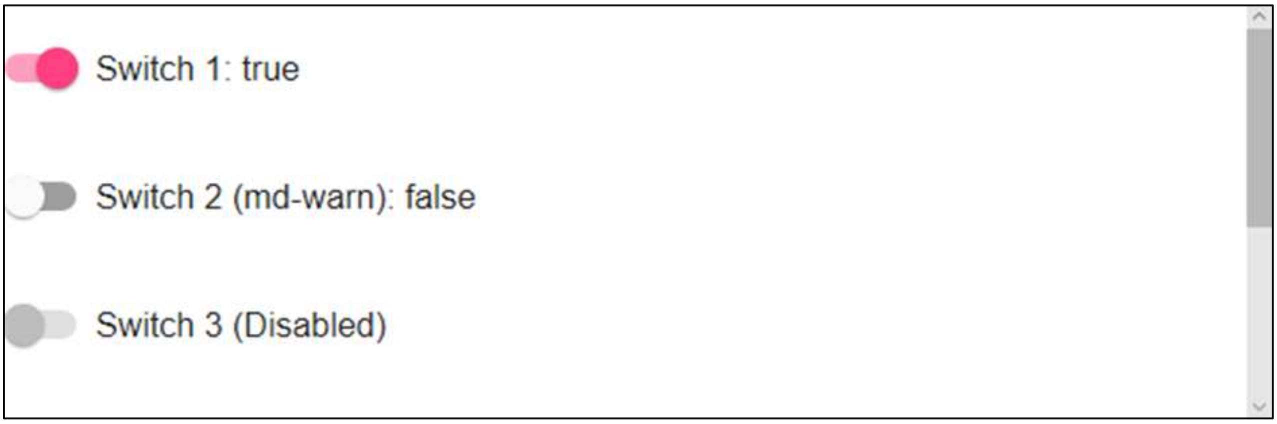
The following example shows the use of md-swipe-* and also the uses of swipe components.

am_switches.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
       <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
          angular
            .module('firstApplication', ['ngMaterial'])
            .controller('switchController', switchController);

          function switchController ($scope) {
             $scope.data = {
                switch1: true,
                         switch2: false,
                         switch3: false,
                         switch4: true,
                switch5: true,
                switch6: false
             };
             $scope.message = 'false';
             $scope.onChange = function(state) {
                $scope.message = state;
             };
          }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="switchContainer" ng-controller="switchController as ctrl"
layout="column" ng-cloak>
         <md-switch ng-model="data.switch1" aria-label="Switch 1">
            Switch 1: {{ data.switch1 }}
         </md-switch>
         <md-switch ng-model="data.switch2" aria-label="Switch 2" ng-true-
value="'true'" ng-false-value="'false'" class="md-warn">
            Switch 2 (md-warn): {{ data.switch2 }}
         </md-switch>
         <md-switch ng-disabled="true" aria-label="Disabled switch" ng-
model="disabledModel">
            Switch 3 (Disabled)
         </md-switch>
         <md-switch ng-disabled="true" aria-label="Disabled active switch" ng-
model="data.switch4">
            Switch 4 (Disabled, Active)
         </md-switch>
         <md-switch class="md-primary" md-no-ink aria-label="Switch No Ink" ng-
model="data.switch5">
            Switch 5 (md-primary): No Ink
         </md-switch>
         <md-switch ng-model="data.switch6" aria-label="Switch 6" ng-
change="onChange(data.switch6)">
            Switch 6 : {{ message }}
         </md-switch>
      </div>
   </body>
</html>
```

Result

Verify the result.



Switch 1: true

Switch 2 (md-warn): false

Switch 3 (Disabled)

16. Angular Material – Theme

The Angular Material components use the intention group classes like md-primary, md-accent, md-warn and additional classes for color differences like md-hue-1, md-hue-2, or md-hue-3. The following components supporrt the use of the above mentioned classes.

- md-button
- md-checkbox
- md-progress-circular
- md-progress-linear
- md-radio-button
- md-slider
- md-switch
- md-tabs
- md-text-float
- md-toolbar

Themes can be configured using $mdThemingProvider during application configuration. The following properties can be used to assign different color palletes.

- primaryPalette
- accentPalette
- warnPalette
- backgroundPalette

Example

The following example shows the use of themes in Angular JS application.

am_themes.htm

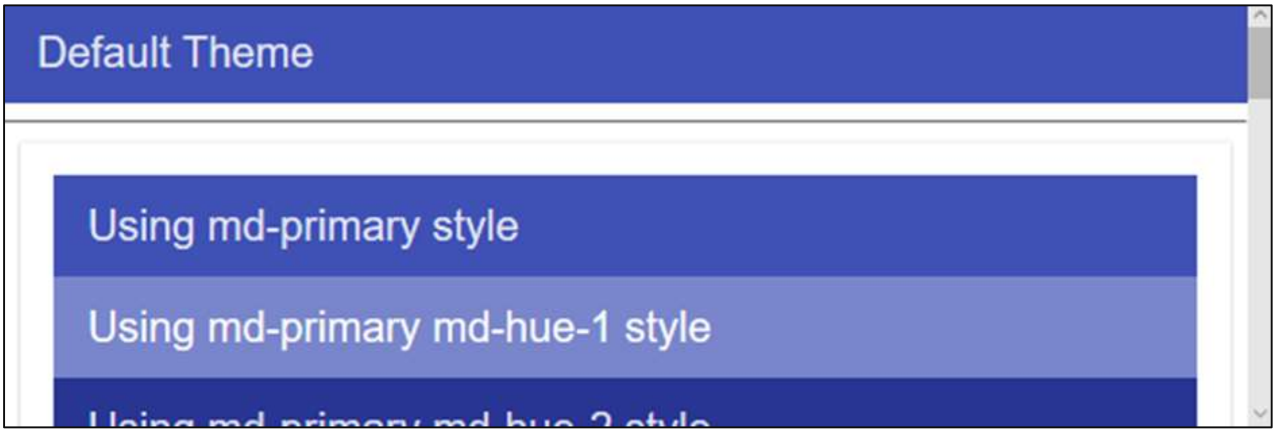
```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
```

```
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>

      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>

      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">

      <script language="javascript">

         angular

            .module('firstApplication', ['ngMaterial'])

            .controller('themeController', themeController)

            .config(function($mdThemingProvider) {

               $mdThemingProvider.theme('customTheme')

                  .primaryPalette('grey')

                  .accentPalette('orange')

                  .warnPalette('red');

               });

            function themeController ($scope) {

            }

      </script>

   </head>

   <body ng-app="firstApplication">

      <div id="themeContainer" ng-controller="themeController as ctrl" ng-cloak>

         <md-toolbar class="md-primary"><div class="md-toolbar-tools"><h2 class="md-
flex">Default Theme</h2></div></md-toolbar>

      <hr/>

      <md-card>

         <md-card-content layout="column">

            <md-toolbar class="md-primary"><div class="md-toolbar-tools"><h2
class="md-flex">Using md-primary style</h2></div></md-toolbar>

            <md-toolbar class="md-primary md-hue-1"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-primary md-hue-1 style</h2></div>

</md-toolbar>

            <md-toolbar class="md-primary md-hue-2"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-primary md-hue-2 style</h2></div></md-toolbar>

            <md-toolbar class="md-accent"><div class="md-toolbar-tools"><h2
class="md-flex">Using md-accent style</h2></div></md-toolbar>

            <md-toolbar class="md-accent md-hue-1"><div class="md-toolbar-tools"><h2
class="md-flex">Using md-accent md-hue-1 style</h2></div></md-toolbar>
```

```
            <md-toolbar class="md-accent md-hue-2"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-accent md-hue-2 style</h2></div></md-toolbar>
            <md-toolbar class="md-warn"><div class="md-toolbar-tools"><h2 class="md-
flex">Using md-warn style</h2></div></md-toolbar>
            <md-toolbar class="md-warn md-hue-1"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-warn md-hue-1 style</h2></div></md-toolbar>
            <md-toolbar class="md-warn md-hue-2"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-warn md-hue-2 style</h2></div></md-toolbar>
         </md-card-content>
      </md-card>
         <div md-theme="customTheme">
            <md-toolbar class="md-primary"><div class="md-toolbar-tools"><h2 
class="md-flex">Custom Theme</h2></div></md-toolbar>
            <hr/>
            <md-card>
               <md-card-content layout="column">
                  <md-toolbar class="md-primary"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-primary style</h2></div></md-toolbar>
                  <md-toolbar class="md-primary md-hue-1"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-primary md-hue-1 style</h2></div></md-toolbar>
                  <md-toolbar class="md-primary md-hue-2"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-primary md-hue-2 style</h2></div></md-toolbar>
                  <md-toolbar class="md-accent"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-accent style</h2></div></md-toolbar>
                  <md-toolbar class="md-accent md-hue-1"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-accent md-hue-1 style</h2></div></md-toolbar>
                  <md-toolbar class="md-accent md-hue-2"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-accent md-hue-2 style</h2></div></md-toolbar>
                  <md-toolbar class="md-warn"><div class="md-toolbar-tools"><h2 
class="md-flex">Using md-warn style</h2></div></md-toolbar>
                  <md-toolbar class="md-warn md-hue-1"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-warn md-hue-1 style</h2></div></md-toolbar>
                  <md-toolbar class="md-warn md-hue-2"><div class="md-toolbar-
tools"><h2 class="md-flex">Using md-warn md-hue-2 style</h2></div></md-toolbar>
               </md-card-content>
            </md-card>
         </div>
      </div>
   </body>
</html>
```

Result

Verify the result.



Default Theme

Using md-primary style

Using md-primary md-hue-1 style

Using md-primary md-hue-2 style

17. Angular Material – Toasts

The Angular Material provides various special methods to show unobtrusive alerts to the users. It also provides a term toast for them. The $mdToast service is used to show toasts.

Example

The following example shows the use of toasts.

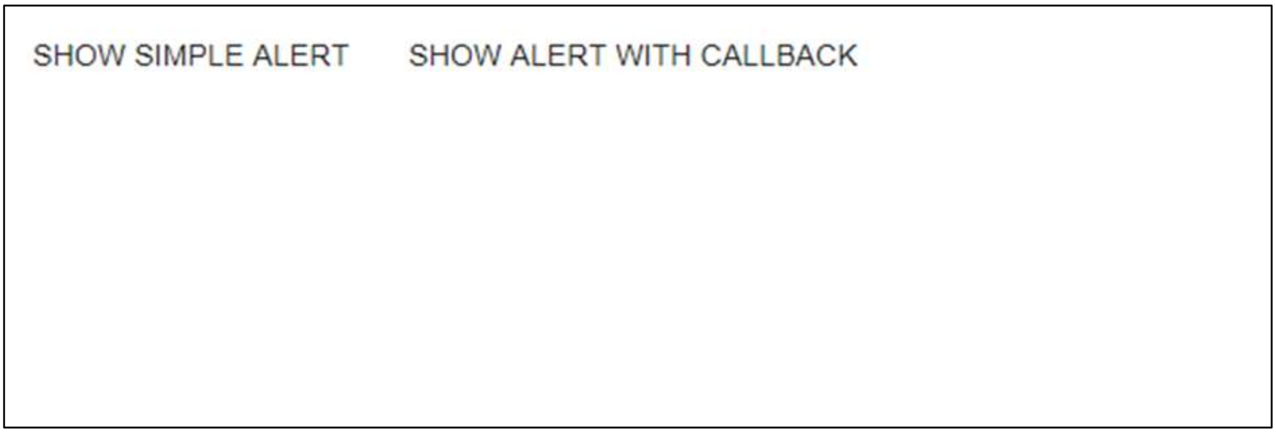
am_toasts.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('toastController', toastController);

            function toastController ($scope, $mdToast, $document) {
                  $scope.showToast1 = function() {
                  $mdToast.show(
                     $mdToast.simple()
                        .textContent('Hello World!')
                        .hideDelay(3000)
                  );
```

```
                };

                $scope.showToast2 = function() {
                   var toast = $mdToast.simple()
                      .textContent('Hello World!')
                      .action('OK')
                      .highlightAction(false);
                   $mdToast.show(toast).then(function(response) {
                      if ( response == 'ok' ) {
                         alert('You clicked \'OK\'.');
                      }
                   });
                }
             }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div id="toastContainer" ng-controller="toastController as ctrl" layout="row"
ng-cloak>
         <md-button ng-click="showToast1()">Show Simple Alert</md-button>
         <md-button ng-click="showToast2()">Show Alert with callback</md-button>
      </div>
   </body>
</html>
```

Result

Verify the result.



18. Angular Material – Typography

Angular Material provides various typography CSS classes which can be used to create visual consistency across Angular JS application.

The following table lists down the different classes with their description.

S.NO.	Class Name & Description
1	**md-display-1** Shows the text with Regular 34px.
2	**md-display-2** Shows the text with Regular 45px.
3	**md-display-3** Shows the text with Regular 56px.
4	**md-display-4** Shows the text with Light 112px.
5	**md-headline** Shows the text with Regular 24px.
6	**md-title** Shows the text with Medium 20px.
7	**md-subhead** Shows the text with Regular 16px.
8	**md-body-1** Shows the text with Regular 14px.
9	**md-body-2** Shows the text with Medium 14px.

S.NO.	Class Name & Description
10	**md-button** Shows the button with Medium 14px.
11	**md-caption** Shows the text with Regular 12px.

Example

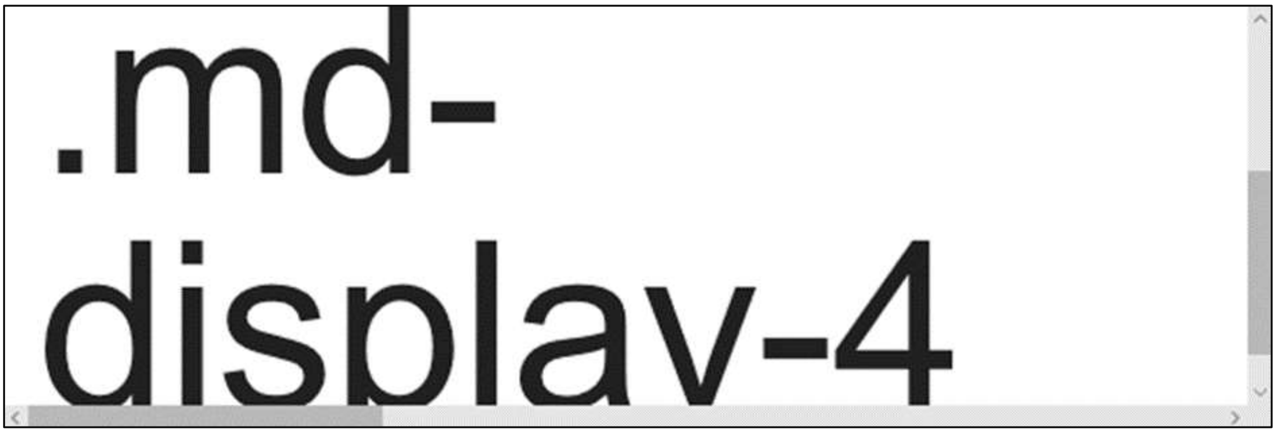
The following example shows the use of typography CSS classes.

am_typography.htm

```
<html lang="en" >
   <head>
      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('typographyController', typographyController);

            function typographyController ($scope) {
            }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div class="frameContainer" ng-controller="typographyController as ctrl"
layout="column" layout-padding layout-wrap layout-fill style="padding-bottom: 32px;"
ng-cloak>
         <p class="md-display-4">.md-display-4</p>
         <p class="md-display-3">.md-display-3</p>
         <p class="md-display-2">.md-display-2</p>
         <p class="md-display-1">.md-display-1</p>
         <p class="md-headline">.md-headline</p>
             <p class="md-title">.md-title</p>
             <p class="md-subhead">.md-subhead</p>
         <p class="md-body-1">.md-body-1</p>
         <p class="md-body-2">.md-body-2</p>
         <md-button>.md-button</md-button>
         <p class="md-caption">.md-caption</p>
      </div>
   </body>
</html>
```

Result

Verify the result.



.md-display-4

19. Angular Material – Virtual Repeat

The **md-virtual-repeat-container** is the scroll container for the md-virtual-repeat component.

Attributes

The following table lists out the parameters and description of the different attributes of **md-virtual-repeat-container**.

S.NO.	Parameter & Description
1	**md-top-index** Binds the index of the item that is at the top of the scroll container to $scope. It can both read and set the scroll position.
2	**md-orient-horizontal** Determines whether the container should scroll horizontally (defaults to orientation and scrolling vertically).
3	**md-auto-shrink** When present, the container will shrink to fit the number of items when that number is less than its original size.
4	**md-auto-shrink-min** Minimum number of items that md-auto-shrink will shrink to (default: 0).

md-virtual-repeat

Virtual repeat is a substitute for ng-repeat to renders only enough html elements to fill the container and reuse them when the user scrolls.

Attributes

The following table lists out the parameters and description of the different attributes of **md-virtual-repeat**.

S.NO.	Parameter & Description
1	**md-item-size** The height or width of the repeated elements (which must be identical for each element). This is optional. This attempts to read the size from the dom if missing, but still assumes that all repeated nodes have same height or width.

S.NO.	Parameter & Description
2	**md-extra-name** Evaluates to an additional name to which the current iterated item can be assigned on the repeated scope (needed for use in md-autocomplete).
3	**md-on-demand** When present, treats the **md-virtual-repeat** argument as an object that can fetch rows rather than an array.This object must implement the following interface with two (2) methods: getItemAtIndex: function(index) [object] - The item at that index or null if it is not yet loaded (it should start downloading the item in that case). getLength: function() [number] - The data length to which the repeater container should be sized. Ideally, when the count is known, this method should return it. Otherwise, return a higher number than the currently loaded items to produce an infinite-scroll behavior.

Example

The following example showe the use of virtual repeat.

am_virtualrepeat.htm

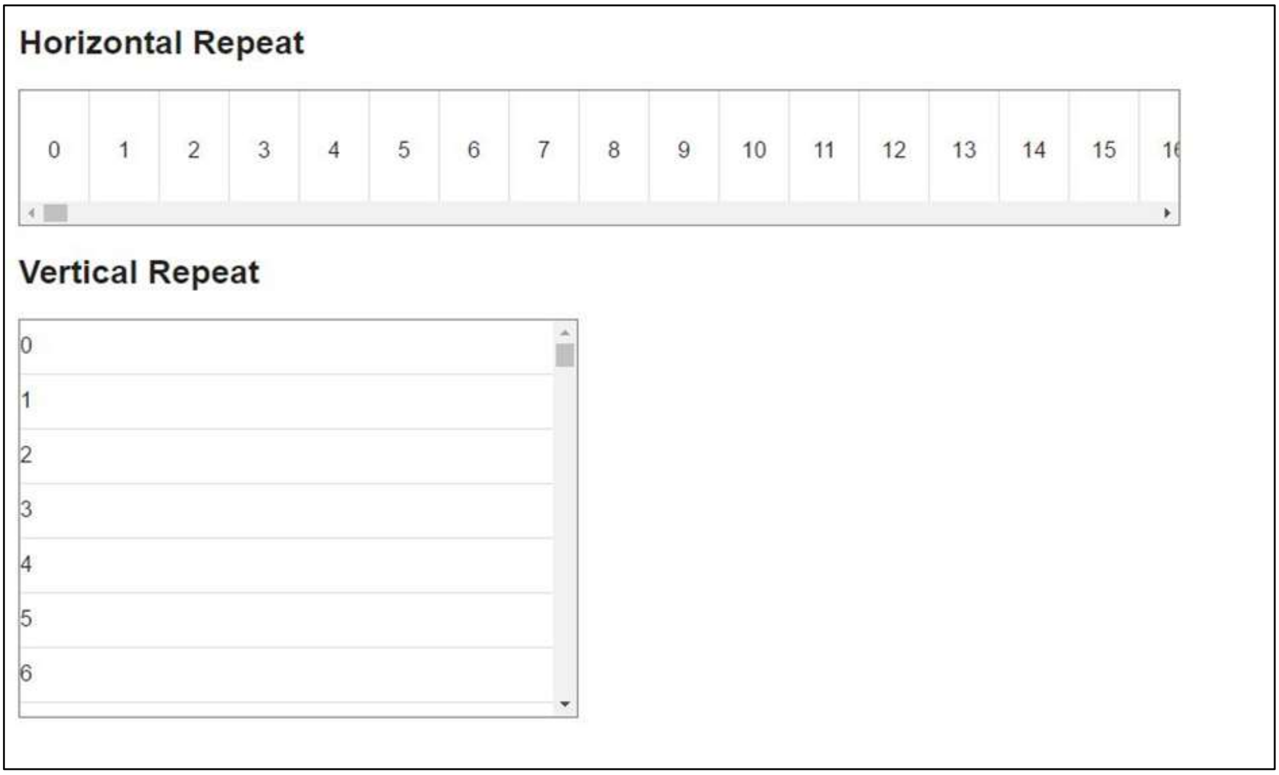
```
<html lang="en" >
   <head>
      <link rel="stylesheet" 
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">
      <script 
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>
      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
      <script 
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet" 
href="https://fonts.googleapis.com/icon?family=Material+Icons">
        <style>
          .vrepeatContainer #horizontal-container {
```

```
         height: 100px;
         width: 830px;
       }

       .vrepeatContainer #vertical-container {
           height: 292px;
           width: 400px;
       }

       .vrepeatContainer .repeated-item-horizontal {
          border-right: 1px solid #ddd;
          box-sizing: border-box;
          display: inline-block;
          height: 84px;
          padding-top: 35px;
          text-align: center;
          width: 50px;
       }

       .vrepeatContainer .repeated-item-vertical {
          border-bottom: 1px solid #ddd;
          box-sizing: border-box;
          height: 40px;
          padding-top: 10px;
       }

       .vrepeatContainer md-content {
          margin: 16px;
       }
       .vrepeatContainer md-virtual-repeat-container {
          border: solid 1px grey;
       }
     </style>
    <script language="javascript">
       angular
          .module('firstApplication', ['ngMaterial'])
```

```
             .controller('vrepeatController', vrepeatController);

             function vrepeatController ($scope) {
                this.items = [];
                for (var i = 0; i < 1000; i++) {
                   this.items.push(i);
                }
             }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div class="vrepeatContainer" ng-controller="vrepeatController as ctrl" ng-
cloak>
         <md-content layout="column">
            <h2>Horizontal Repeat</h2>
            <md-virtual-repeat-container id="horizontal-container" md-orient-
horizontal>
               <div md-virtual-repeat="item in ctrl.items"
                  class="repeated-item-horizontal" flex>
                  {{item}}
               </div>
            </md-virtual-repeat-container>
            <h2>Vertical Repeat</h2>
            <md-virtual-repeat-container id="vertical-container">
                <div md-virtual-repeat="item in ctrl.items"
                  class="repeated-item-vertical" flex>
                  {{item}}
               </div>
            </md-virtual-repeat-container>
         </md-content>
      </div>
   </body>
</html>
```

Result

Verify the result.



Horizontal Repeat

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Vertical Repeat

0
1
2
3
4
5
6

20. Angular Material – WhiteFrame

Angular Material has several special classes to display containers as paper-like cards with shadow.

The following table lists down the different classes with their description.

S.NO.	Class Name & Description
1	**md-whiteframe-1dp** This styles a container for any HTML content with 1px bordered shadow. Adds z-depth of 1.
2	**md-whiteframe-2dp** This styles a container for any HTML content with 2px bordered shadow. Adds z-depth of 2.
3	**md-whiteframe-3dp** This styles a container for any HTML content with 3px bordered shadow. Adds z-depth of 3.
4	**md-whiteframe-4dp** This styles a container for any HTML content with 4px bordered shadow. Adds z-depth of 4.
5	**md-whiteframe-5dp** This styles a container for any HTML content with 5px bordered shadow. Adds z-depth of 5.
6	**md-whiteframe-6dp** This styles a container for any HTML content with 6px bordered shadow. Adds z-depth of 6.
7	**md-whiteframe-7dp** This styles a container for any HTML content with 7px bordered shadow. Adds z-depth of 7.
8	**md-whiteframe-8dp** This styles a container for any HTML content with 8px bordered shadow. Adds z-depth of 8.
9	**md-whiteframe-9dp**

S.NO.	Class Name & Description
	This styles a container for any HTML content with 9px bordered shadow. Adds z-depth of 9.
10	**md-whiteframe-10dp** This styles a container for any HTML content with 10px bordered shadow. Adds z-depth of 10.
11	**md-whiteframe-11dp** This styles a container for any HTML content with 11px bordered shadow. Adds z-depth of 11.
12	**md-whiteframe-12dp** This styles a container for any HTML content with 12px bordered shadow. Adds z-depth of 12.
13	**md-whiteframe-13dp** This styles a container for any HTML content with 13px bordered shadow. Adds z-depth of 13.
14	**md-whiteframe-14dp** This styles a container for any HTML content with 14px bordered shadow. Adds z-depth of 14.
15	**md-whiteframe-15dp** This styles a container for any HTML content with 15px bordered shadow. Adds z-depth of 15.
16	**md-whiteframe-16dp** This styles a container for any HTML content with 16px bordered shadow. Adds z-depth of 16.
17	**md-whiteframe-17dp** This styles a container for any HTML content with 17px bordered shadow. Adds z-depth of 17.
18	**md-whiteframe-18dp** This styles a container for any HTML content with 18px bordered shadow. Adds z-depth of 18.

19	**md-whiteframe-19dp** This styles a container for any HTML content with 19px bordered shadow. Adds z-depth of 19.
20	**md-whiteframe-20dp** This styles a container for any HTML content with 20px bordered shadow. Adds z-depth of 20.
21	**md-whiteframe-21dp** This styles a container for any HTML content with 21px bordered shadow. Adds z-depth of 21.
22	**md-whiteframe-22dp** This styles a container for any HTML content with 22px bordered shadow. Adds z-depth of 22.
23	**md-whiteframe-23dp** This styles a container for any HTML content with 23px bordered shadow. Adds z-depth of 23.
24	**md-whiteframe-24dp** This styles a container for any HTML content with 24px bordered shadow. Adds z-depth of 24.

Example

The following example shows the use of shadow classes.

am_whiteframes.htm

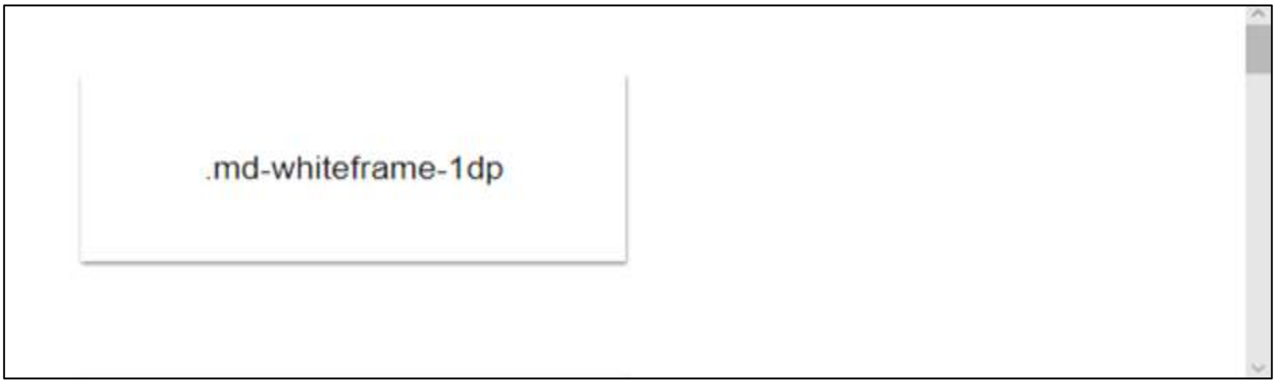
```
<html lang="en" >

   <head>

      <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.css">

      <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
animate.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
aria.min.js"></script>

      <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular-
messages.min.js"></script>
```

```
      <script
src="https://ajax.googleapis.com/ajax/libs/angular_material/1.0.0/angular-
material.min.js"></script>
      <link rel="stylesheet"
href="https://fonts.googleapis.com/icon?family=Material+Icons">
       <style>
       .frameContainer md-whiteframe {
          background: #fff;
          margin: 30px;
          height: 100px;
       }
       </style>
      <script language="javascript">
         angular
            .module('firstApplication', ['ngMaterial'])
            .controller('frameController', frameController);

            function frameController ($scope) {
            }
      </script>
   </head>
   <body ng-app="firstApplication">
      <div class="frameContainer" ng-controller="frameController as ctrl"
layout="row" layout-padding layout-wrap layout-fill style="padding-bottom: 32px;"
ng-cloak>
         <md-whiteframe class="md-whiteframe-1dp" flex-sm="45" flex-gt-sm="35" flex-
gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-1dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-2dp" flex-sm="45" flex-gt-sm="35" flex-
gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-2dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-3dp" flex-sm="45" flex-gt-sm="35" flex-
gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-3dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-10dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
```

```
            <span>.md-whiteframe-10dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-15dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-15dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-20dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-20dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-22dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-22dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-23dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-23dp</span>
         </md-whiteframe>
         <md-whiteframe class="md-whiteframe-24dp" flex-sm="45" flex-gt-sm="35"
flex-gt-md="25" layout layout-align="center center">
            <span>.md-whiteframe-24dp</span>
         </md-whiteframe>
      </div>
   </body>
</html>
```

Result

Verify the result.



.md-whiteframe-1dp