Session 10

AngularJS Material

Session Objectives

- ✓ Describe AngularJS Material

- ✓ Learn about Customizations in AngularJS Material

- ✓ Explain how to use API references in AngularJS Material

- ✓ Explain the process of migration from AngularJS to Angular

What is AngularJS Material?

- Is a UI Component library

- Provides a set of reusable, well-tested, and accessible UI components for AngularJS developers

Pre-requisite: You need to have any version of AngularJS between 1.7.2 to 1.8.x, in order to work with AngularJS Material.

Autocomplete

Bottom Sheet

Button

Card

Checkbox

Chips

Colors

Content

Datepicker

Dialog

Divider

FAB Speed Dial

FAB Toolbar

Grid List

Icon

Input

Features and Benefits of AngularJS Material

In-built responsive designing

Standard CSS with minimal footprint

Cross-browser
Also, it can be used to create reusable Web components

Includes new versions of common user interface controls such as buttons, check boxes, and text fields that are adapted to follow Material Design concepts

Includes enhanced and specialized features such as cards, toolbar, speed dial, side nav, swipe, and so on

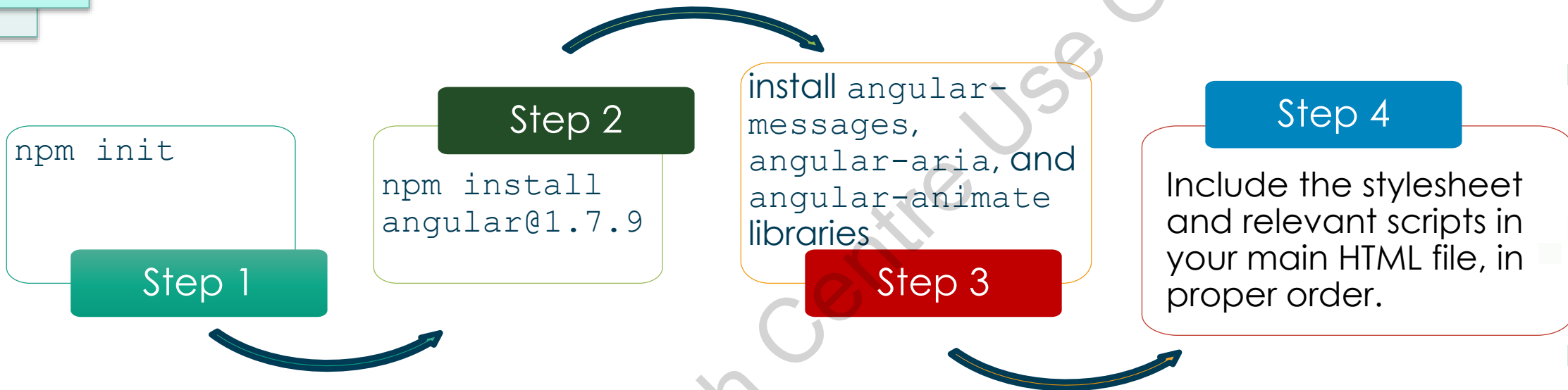AngularJS Material also provides extensible capabilities

AngularJS Material is by design minimal and flat.

It is designed considering 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.

Installing AngularJS Material

Step 1

```
npm init
```

Step 2

```
npm install
angular@1.7.9
```

Step 3

install `angular-messages`, `angular-aria`, and `angular-animate` libraries

Step 4

Include the stylesheet and relevant scripts in your main HTML file, in proper order.

Alternately, use Google CDN for an easier experience.

Customization

Customizations allow developers to make changes according to specific requirements.

In AngularJS Material, customizations are allowed in terms of following components:

- Cascading Stylesheets
- Themes
- Performance

CSS classes are provided for Typography, Buttons, and Check boxes.

Typography

Can be used to create visual consistency in AngularJS applications.
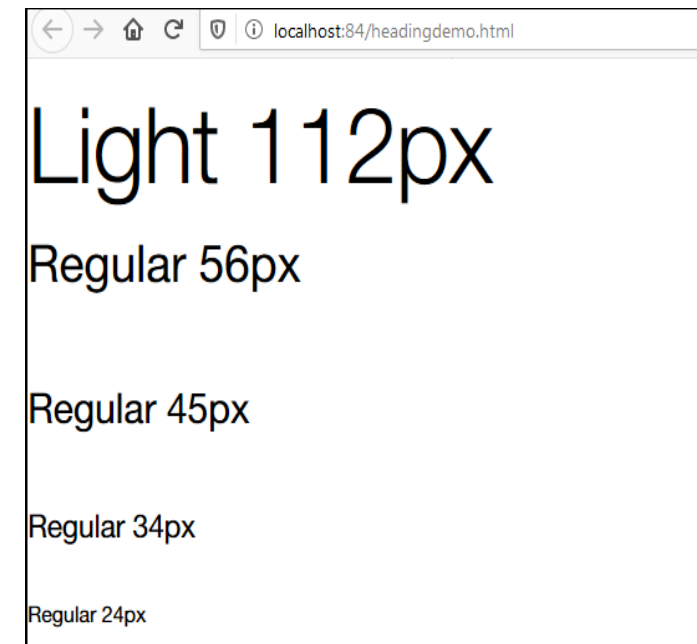
General Typography

- Roboto font is used by AngularJS Material for all of its components.

Heading Styles

- Heading tags <h1> - <h6> will be styled with styling class selectors.

Example

localhost:84/headingdemo.html

Light 112px

Regular 56px

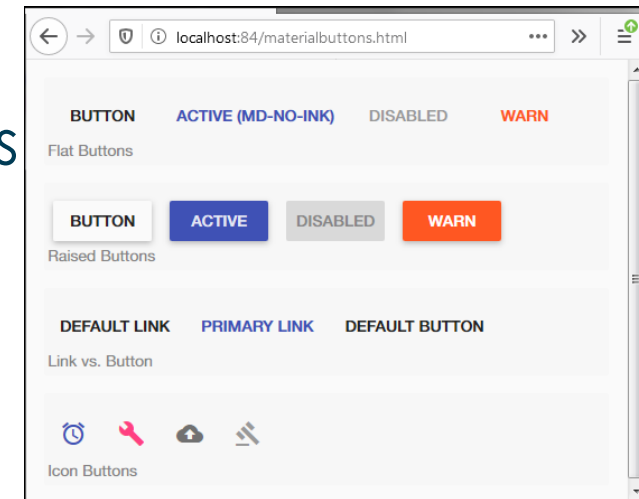### Regular 45px

Regular 34px

Regular 24px

Buttons

- AngularJS Material provides <md-button> classes with default themes and standard options.

- It is useful to create different types of buttons as per requirement inside your application.

- For icon buttons, you can use various icons.

Example

Check boxes

Key properties of `<md-checkbox>` component:

Ink color
- Useful to change color when the check box is checked.

Disabled
- Useful to change color when a disabled check box is checked.

Borders
- Useful to add custom borders.

Theming

Hues or Shades	Palettes	Color Intentions
It refers to a single color within a palette.	It is a collection of hues.	It is a mapping of a palette for a given intention into the application.

Theming

Following are valid intentions in AngularJS Material:

- primary - represent primary interface elements for a user
- accent - represent secondary interface elements for a user
- warn - represent interface elements that the user should be careful of

Theme Configuration

- AngularJS Material components use intention group classes such as **md-primary**, **md-accent**, and **md-warn**.

- Components used with earlier mentioned classes are as follows:
  - md-button
  - md-checkbox
  - md-progress-circular
  - md-progress-linear
  - md-radio-button
  - md-slider
  - md-switch
  - md-tabs
  - md-text-float
  - md-toolbar

- Additional classes for color differences are include **md-hue-1**, **md-hue-2**, or **md-hue-3**.

- Themes are configured using **$mdThemingProvider** during configuration process of applications.

Theme Configuration

Following properties are used to assign different color palettes:

- primaryPalette
- accentPalette
- warnPalette
- backgroundPalette

Layouts and Containers

- API for layout contains a set of AngularJS directives which are applied to any HTML element in an application.
- Attributes are useful to define layout and CSS classes are useful to assign styling.
- Layout directive on top of the container element is useful to specify layout direction for its children.
- Following values are used for the Layout Directive:

row

Items are arranged horizontally. max-height is given as 100% and max-width is set as the width of container in which item is present.

column

Items are arranged vertically. max-width is given as 100% and max-height is the height of container in which items are present.

APIs to Set Layout Direction

layout

It sets default layout direction until another breakpoint is provided.

layout-xs

width < 600px

layout-gt-xs

width >= 600px

layout-sm

600px <= width < 960px

layout-gt-sm

width >= 960px

APIs to Set Layout Direction

layout-md

960px <= width < 1280px

layout-gt-md

width >= 1280px

layout-lg

1280px <= width < 1920px

layout-gt-lg

width >= 1920px

layout-xl

width >= 1920px

Flex Directive

flex directive

Useful to customize size and position of HTML elements.

Helps to define and adjust size with respect to the parent container and other elements within the container.

Following values can be used for flex directive:

- 5 – 5, 10, 15 ... 100
- 33 – 33%
- 66 – 66%

Containers

- Containers are generally used to create overlaying elements such as dialogs, menus, navigation bars, and so on.

- Container components are programmable and used to show or hide elements.

Migration

- Migration from AngularJS Material to Angular Material involves multiple aspects such as features moved to Angular Component Development Kit (CDK), changes in theming system, separate library for AngularJS material, and so on.

- Official ngUpgrade guide covers general practices around AngularJS to Angular migration.

Summary

➢ AngularJS Material is a UI Component library that provides a set of reusable, well-tested, and accessible UI components for AngularJS developers.

➢ Typography is used to create visual consistency in AngularJS applications.

➢ md-button class is used to create different types of buttons such as Fab, Flat, Raised, and so on.

➢ md-checkbox class is used to create check boxes with different styles.

➢ Layout enables developers to create modern, responsive layouts on CSS3 flex box.

➢ flex directive is useful to customize size and position of the HTML elements.

➢ Overlaying elements are created using containers.

➢ AngularJS Material to Angular Material migration is supported through various helpful documents and guides.