

Introduction

BUILDING WEB APPLICATIONS USING ANGULAR



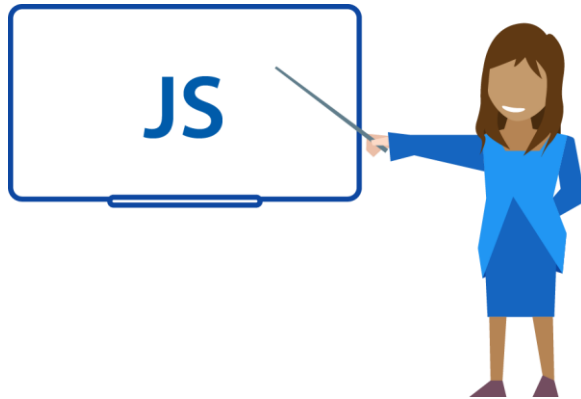
Overview

- Objectives
 - To explain the aims and objectives of the course
- Contents
 - Course administration
 - Course objectives and assumptions
 - Introductions
 - Any questions?
- Exercise
 - Locate the exercises
 - Locate the help files

Administration

- Front door security
 - Name card
 - Chairs
 - Fire exits
 - Toilets
 - Coffee Room
 - Timing
 - Breaks
 - Lunch
- Downloads and viruses
 - Admin. support
 - Messages
 - Taxis
 - Trains/Coaches
 - Hotels
 - First Aid
 - Telephones/Mobiles

Course delivery



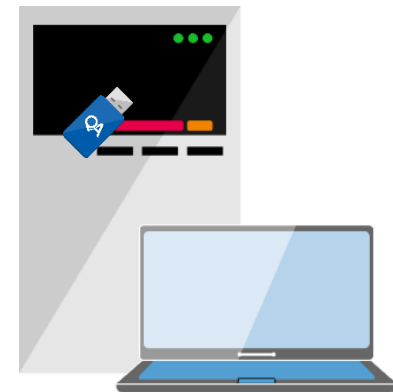
Lecture material



Hear and Forget
See and Remember
Do and Understand



Course workbooks



A training experience

A course should be

- A two-way process
- A group process
- An individual experience



Course aims and objectives

By the end of the course, you will be able to:

- Explain what Angular is
- Set up an Angular project
- Create Angular components, directives, pipes, services and more
- Use state and props in components
- Use forms in both Template and Reactive driven styles
- Make asynchronous requests for data to remote services
- Use the Angular Router to create Single Page Applications
- Use Test Driven Development tools and techniques to build better applications

Assumptions

- This course assumes the following prerequisites
 - You have experience with JavaScript (including ES2015)
 - You have experience with HTML and CSS
- If there are any issues, please tell your instructor now

Introductions

- Please say a few words about yourself
- What is your name and job?
- What is your current experience of
 - Programming?
 - Web development?
 - JavaScript
- What is your main objective for attending the course?

Any questions

- Golden Rule
 - "There is no such thing as a stupid question"
- First amendment to the Golden Rule
 - "... even when asked by an instructor"
- Corollary to the Golden Rule
 - "A question never resides in a single mind"

What is Angular?

- A framework for building scalable web applications
- Angular, aka Angular 2, aka Angular N* is the latest framework from the team that brought you AngularJS aka Angular 1
- It is a significant step-change from AngularJS
 - No \$scope
 - No controllers
 - Used TypeScript (you can use JavaScript, but there's not much support out there yet)

Why the big change?

- AngularJS has been around since 2009, before ECMAScript 5 and HTML5
- The face of web applications and the technology powering them has changed significantly and at an accelerating pace since AngularJS
- Introduction of transpilation has promoted the ability to build larger, more complex web applications

Semver

- Semantic versioning defines the versioning of a software project should follow the format X.Y.Z where
 - Z is the patch version, a change here indicates there are no breaking changes – just bug fixes
 - Y is the minor version, a change here indicates new features have been introduced but again, no breaking changes
 - X is the major version, a change here indicates breaking changes have been introduced and this version is no longer compatible with the previous
- Starting with Angular 2, Angular will now follow semver along with a new schedule
 - A new patch every week
 - Three monthly minor releases
 - One major release every six months
- AngularJS released major releases as 1.1, 1.2, 1.3 etc
- What happened to Angular 3 then? (skipped in order to bring versions in-line with Router)