Capgemini

Table of Content

- What you will learn
- Pre-requisite for the course
- Software requirement
- Day wise schedule
- Useful resource link
- Assignments to be solved

What you will learn

- Understand how single-page web application architectures are different than traditional web application architectures
- How to organize features and modules
- Properly structure components
- How to use new JavaScript (ES6) language features including Classes, Modules, and Arrow Functions
- How to use new TypeScript language features including Types, Decorators, Interfaces, and Generics
- Angular coding and architecture best practices including project layout and using container and presentation components.
- Understand and use Angular model-driven forms, observables, dependency injection, and routing
- How to communicate with a backend server using Angular's HttpClient to load and save data
- How to configure the router and navigate between components.
- How to manage state in Angular applications.
- You will understand Model-driven approach and Template-driven approach for form design and when to use which approach.

Pre-requisites

- **HTML :** Most of the templates we create in angular is in the form of handcrafted HTMLs. i.e. So you must know that what are forms in html and what are tags ng-form etc.
- **CSS :** While hand crafting template you should require CSS to make more attractive UI design.
- **DOM :** Document object model and how document is created. If you have good jQuery background you can easily pick up this part.
- **Object Oriented JavaScript:** Global name space: AngularJS heavily uses JavaScript name space. i.e.
- **TypeScript :** Basic Understanding of TypeScript.

Software requirement

- Download and Install Node.js from Nodejs.org website.
- Install Angular-CLI
- Download and install Microsoft visual studio code editor(https://code.visualstudio.com/)
- Any Browser(Google/ Internet Explorer/Firefox)

Day Wise Schedule

 # Day 1

Basic

- Overview
- Environment Setup
- Project Setup
- Components
- Module
- Data Binding , Event Binding
- Template
- Directives

Day Wise Schedule

 # Day 2 (**Advanced**)

- Pipes
- Routing
- Services (In built / Custom)
- Form (Create / Validation)
- Create Common Component / services And Integration
- Animation
- Lazy Loading - Component
- Integration With Bootstrap

Day Wise Schedule

 # Day 3 (Advanced)

- Create Application With Updated Version Of Angular. Including,
  - Recommended application structure
  - Create component for each module
  - Create Services for each module
  - Common Layout – Header , Left-Side Panel , Right-Side Panel, Footer (As per requirement of App)
  - Data Sharing between components
  - Implement external service using HttpClient Module.
  - Use Form with validation.
  - Implement Lazy Loading for component (Will increase performance of app)
  - Integrate Bootstrap with app and leverage in-built functionality.

Useful Resources

- **Angular Docs (Official Site)**
  - https://angular.io/docs

- **Angular Tutorials**
  - https://angular.io/tutorial
  - https://angular-templates.io/tutorials/about/learn-angular-from-scratch-step-by-step
  - https://mdbootstrap.com/education/angular/

- **Angular YouTube Videos Series**
  - https://www.youtube.com/watch?v=k5E2AVpwsko
  - https://www.youtube.com/watch?v=2OHbjep_WjQ
  - https://www.youtube.com/watch?v=5wtnKulcquA
  - https://www.youtube.com/watch?v=Fdf5aTYRW0E

Useful Resources

- **Angular Testing**
  - https://www.youtube.com/watch?v=D6qPDww2X8k
  - https://www.youtube.com/watch?v=o7N5JyhvKIY

- **Best Editor**
  - https://code.visualstudio.com/

- **Useful Links for Typescript**
  - https://blog.angular.io/
  - https://johnpapa.net/angular-style-guide/
  - https://johnpapa.net/essential-angular-vs-code-extensions/
  - https://johnpapa.net/rec-ng-extensions/

Angular Practical Lab Assignment

1. Install NodeJs
2. Install Angular CLI
3. Install Visual Studio Code Editor
4. Create Default Structure by using angular-cli commands.

Reference:

- https://nodejs.org/en/
- https://cli.angular.io/
- https://code.visualstudio.com/

2. Overview Of Assignment

Assignments

double click on image

Create web app which help to manage and maintain restaurant chain including,

- Module
  - -Admin
  - -Restaurant (create / Edit / Delete)
  - -Menu (create / Edit / Delete)
  - -Menu Assignment To Restaurant
  - -Restaurant Owner
  - -User (End User)

3.Steps

1. Create application Structure by using angular-cli

2. Create component
   - Admin
   - Menu
   - Restaurant
   - User

3. Implement routing

Thank you for your Participation

kindly reach atul.k@capgemini.com for queries.