


Enriched Web Development with Angular 11

SKILLSOFT ASPIRE JOURNEY

skillsoft 



 **percipio**™

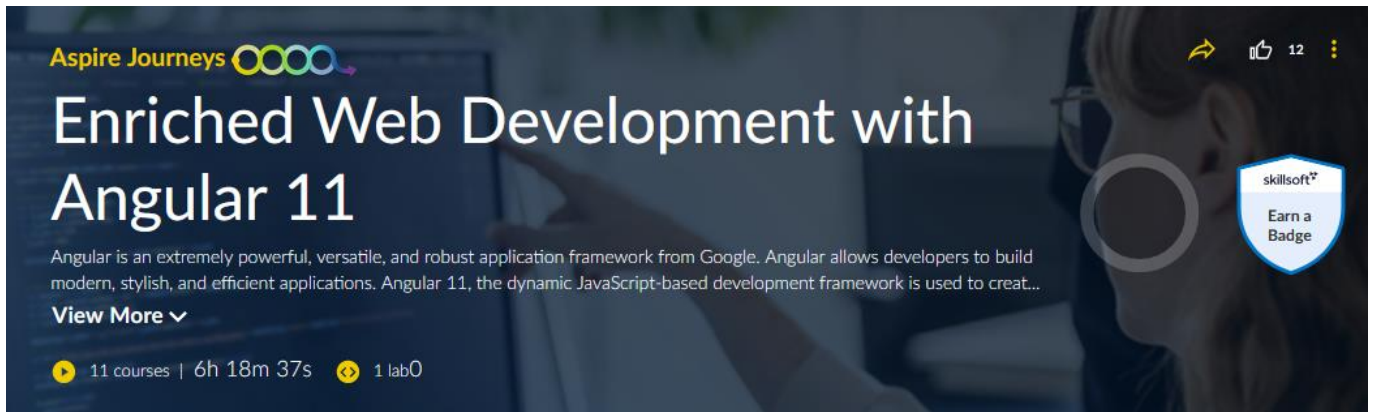

Aspire Journeys 

Enriched Web Development with Angular 11

Angular is an extremely powerful, versatile, and robust application framework from Google. Angular allows developers to build modern, stylish, and efficient applications. Angular 11, the dynamic JavaScript-based development framework is used to creat...

[View More](#) ▾

 11 courses | 6h 18m 37s  1 lab0



Tracks



Track 1: Building Web Applications with Angular 11

In this track of the Enriched Web Development with Angular 11 journey, you will explore the different building blocks of Angular, its architectural concepts, and Typescript as the scripting base for Angular. Then, you will discover the basics of the framework such as forms, modules, pipes, and components. The journey then deep dives to the more complex principles of routing, directives, templates, and services along with demos on how to build end-to-end, working applications that are maintainable and testable. Finally, you will learn the distinct features of Angular 11 and Angular 12.

[View Less](#)

[Explore](#)  11 courses | 6h 18m 37s  1 lab0

PREREQUISITES

In order to fully profit from the potential of this Aspire Journey, we recommend the following prerequisite skills:

- Familiar with JavaScript and front-end development methodologies.

Aspire Journeys: Enriched Web Development with Angular 11

Track 1: Building Web Applications with Angular 11

In this track of the Enriched Web Development with Angular 11 journey, you will explore the different building blocks of Angular, its architectural concepts, and Typescript as the scripting base for Angular. Then, you will discover the basics of the fr...

[View More](#) ▾

▶ 11 courses | 6h 18m 37s ▶ 1 lab0



Angular 11: Getting Started with Angular

Objectives:

- outline a high-level overview of Angular – its past and present.
- recognize the various parts of an Angular application and how they relate to one other
- list the different building blocks of Angular, namely Angular components, services, modules, and routing
- demonstrate the installation of Node.js
- demonstrate the installation of Visual Studio Code
- demonstrate the installation of the Angular CLI
- create and run an Angular application using the CLI
- describe the characteristics of package.json files
- outline the basic characteristics of TypeScript



Angular 11: Introduction to Components

Objectives:

- define object-oriented programming concepts and outline how to create classes in TypeScript and convert TypeScript classes to functions
- describe how to add private and public properties to a TypeScript class and explain the use of public properties outside the class
- describe how to add functions to a TypeScript class and explain how to access public property in a class function
- describe the concept of decorators in TypeScript and how to add decorators to functions, class variables, and classes
- create a component using the Angular CLI, add the component to a module, and add inline HTML to the component
- add a multi-line template to an Angular component and an external template to a React component
- add inline styles to an Angular component template and use style tags to add inline styles
- add external style and multiple style files to Angular components
- describe what's meant by binding state variables with interpolation, calling functions and mathematical expressions inside interpolation, the use of "ng-bind" for interpolating data, and the use of ng-bind with different HTML elements



Angular 11: Working With Angular Modules

Objectives:

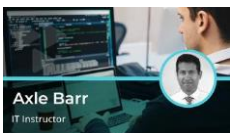
- describe different elements of Angular module and the decorators associated with it
- create an Angular component without CLI, add a new component to the module, and describe the decorator associated with the new module
- Bootstrap components with the desired component by making it the main component for bootstrapping the application
- outline the inclusion and use of third party modules like BrowserModule and explain its functionalities
- create a feature module, add components to it, and use it in the main module
- create a multi-modular system and share the module with multiple modules
- export and import different functionalities and components from other JavaScript files



Angular 11: Directives & Pipes

Objectives:

- use conditional operators in components and render HTML content
- iterate a list of arrays and display array items in the user interface (UI)
- perform iteration on complex objects and render employee details from an employee array
- define pagination, render data on the front-end, and implementing pagination in the application
- sort data in a given list and display an updated sorted list of data
- explain the concept of pipes in Angular and the use of built-in pipes
- create and implement custom pipes in Angular
- perform higher-order functions, filter data from a list, and update listed data based on user input



Angular 11: Working With Nested Components

Objectives:

- describe nested components and the parent-child component structure
- list the use of the @Input keyword and pass data to the child component
- list the use of @Output keyword, pass events from child component to parent component, and update properties of parent component using child component
- explain ViewChild and implement ViewChild property in Angular component



Angular 11: Reactive Programming With RxJS & Observables

Objectives:

- explore Angular RxJS library and describe event-based programming and streams
- examine the characteristics of observables in Angular and differentiate between promises and observables
- create a HTTP Get Request, handle HTTP Response, and create and handle HTTPClient observables
- extract data from observables and subscribe to observables for data



Angular 11: Working With Forms

Objectives:

- implement two-way binding, outline its benefits, and describe banana in a box notation
- create a simple form and bind data inside the form
- explain the form validation mechanism and disable the submit button until the form is validated
- add a mock API server to handle REST calls
- define and send data to POST calls, make asynchronous calls to POST API, and extract a response from POST APIs
- make a Delete call to the API, delete employee data using API calls, and refresh the screen with the updated list
- identify the need for caching and how shareReplay can be used to perform caching
- demonstrate how to implement a local cache with shareReplay



Angular 11: Navigation & Routing in Angular

Objectives:

- define the single page architecture, recognize the justification for building a single-page application (SPA), and describe the overall benefits of an SPA
- define the concept of routing and recognize how it's applied in Angular
- configure simple routes for an application and create a simple single-page application
- pass data to routes and use the data extracted in a component
- demonstrate the activation and deactivation of guards
- demonstrate security implementation using guards



Angular 11: Services & Dependency Injection

Objectives:

- relate dependency injection to Angular components
- create a service in Angular module and populate it with common shared data
- register a service to Angular module
- inject a service inside an Angular component
- access the data available inside a service



Angular 11: Security & Deployment in Angular

Objectives:

- outline the basic workings of authentication and use of route guards in Angular
- add authentication to an Angular application
- create, activate, and deactivate guards for authentication and authorization in Angular routing
- describe scenarios where optimization would be required in an Angular app
- implement lazy loading in Angular
- debug an Angular application and perform unit testing on a component
- deploy an Angular application on the NGINX server



Final Exam:
Building Web
Applications with
Angular 11

Objectives:

- add inline styles to an Angular component template and use style tags to add inline styles
- Bootstrap components with the desired component by making it the main component for bootstrapping the application
- create a component using the Angular CLI, add the component to a module, and add inline HTML to the component
- Create, activate, and deactivate guards for authentication and authorization in Angular routing.
- create a feature module, add components to it, and use it in the main module
- create a HTTP Get Request, handle HTTP Response, and create and handle HTTPClient observables
- create and implement custom pipes in Angular
- define and send data to POST calls, make asynchronous calls to POST API, and extract a response from POST APIs
- define object-oriented programming concepts and outline how to create classes in TypeScript and convert TypeScript classes to functions
- define the single page architecture, recognize the justification for building a single-page application (SPA), and describe the overall benefits of an SPA
- demonstrate how to implement a local cache with shareReplay
- demonstrate security implementation using guards
- demonstrate the installation of the Angular CLI
- describe different elements of Angular module and the decorators associated with it
- describe how to add functions to a TypeScript class and explain how to access public property in a class function
- describe how to add private and public properties to a TypeScript class and explain the use of public properties outside the class
- describe the characteristics of package.json files
- Discuss the features that are new in Angular 11 version and compare those features with older version, if present
- Discuss the requirement of optimization
- Discuss the steps in migration to Angular 11
- examine the characteristics of observables in Angular and differentiate between promises and observables
- explain the concept of pipes in Angular and the use of built-in pipes
- explain the form validation mechanism and disable the submit button until the form is validated
- export and import different functionalities and components from other JavaScript files
- identify the need for caching and how shareReplay can be used to perform caching
- Implement Lazy Loading
- implement two-way binding, outline its benefits, and describe banana in a box notation
- inject a service inside an Angular component
- iterate a list of arrays and display array items in the user interface (UI)
- list the different building blocks of Angular, namely Angular components, services, modules, and routing
- list the use of @Output keyword, pass events from child component to parent component, and update properties of parent component using child component
- list the use of the @Input keyword and pass data to the child component
- outline the basic characteristics of TypeScript
- outline the inclusion and use of third party modules like BrowserModule and explain its functionalities
- pass data to routes and use the data extracted in a component

- perform iteration on complex objects and render employee details from an employee array
- recognize the various parts of an Angular application and how they relate to one other
- register a service to Angular module
- relate dependency injection to Angular components
- use conditional operators in components and render HTML content



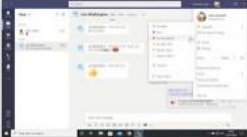

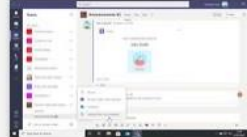




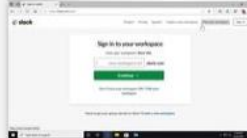











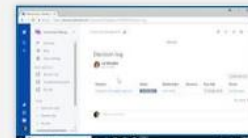

Building Web Applications with Angular 11

Objectives:

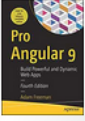







- In this practice lab, learners will be presented with a series of exercises to practice developing in Angular. Exercises include tasks such as defining and adding private and public properties to a typescript class, applying form as input for handling user inputs, and creating a custom pipe for handling data format in a HTML web page. Learners will also share data using query parameters, create custom forms, develop a webapp for multiple page routing, and share data using service injection.

Productivity Tools for Enriched Web Development with Angular 11









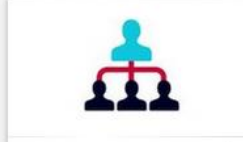

Optional

 <p>COURSE</p> <p>Exploring and setting up Microsoft Teams</p> <p>76</p>	 <p>COURSE</p> <p>Creating and managing teams & channels</p> <p>19</p>	 <p>COURSE</p> <p>Formatting, illustrating & reacting to messages</p> <p>13</p>	 <p>COURSE</p> <p>Using private messaging & call tools</p> <p>15</p>	 <p>COURSE</p> <p>Creating, joining, and managing meetings</p> <p>26</p>
 <p>COURSE</p> <p>Creating, finding & organizing files</p> <p>17</p>	 <p>COURSE</p> <p>Working with Tabs & Apps</p> <p>17</p>	 <p>COURSE</p> <p>Signing in & Setting Up Slack</p> <p>27</p>	 <p>COURSE</p> <p>Using Channels in Slack</p> <p>12</p>	 <p>COURSE</p> <p>Using Private Messaging & Communication Tools in...</p> <p>14</p>
 <p>COURSE</p> <p>Creating, Finding & Sharing Information in Slack</p> <p>8</p>	 <p>COURSE</p> <p>Configuring Slack</p> <p>7</p>	 <p>COURSE</p> <p>Creating & Setting Up Projects in Jira Cloud</p> <p>164</p>	 <p>COURSE</p> <p>Configuring & Managing Boards in Jira Cloud</p> <p>106</p>	 <p>COURSE</p> <p>Planning & Working on a Software Project in Jira...</p> <p>85</p>
 <p>COURSE</p> <p>Reporting in Jira Software</p> <p>83</p>	 <p>COURSE</p> <p>Signing in & Navigating within Spaces</p> <p>40</p>	 <p>COURSE</p> <p>Setting Up & Managing Spaces</p> <p>33</p>	 <p>COURSE</p> <p>Working with Space</p> <p>27</p>	 <p>COURSE</p> <p>Working with Team Members</p> <p>76</p>
 <p>COURSE</p> <p>Configuring Spaces</p> <p>20</p>				

Bookshelf Optional

 <p>BOOK</p> <p>Pro Angular 9: Build Powerful and Dynamic We...</p> <p>2</p>	 <p>BOOK</p> <p>Angular in Action</p> <p>18</p>	 <p>BOOK</p> <p>AngularJS in Action</p> <p>3</p>	 <p>BOOK</p> <p>Angular Development with Typescript, Second Edition</p> <p>16</p>	 <p>BOOK</p> <p>Angular for Material Design: Leverage Angular Material...</p> <p>1</p>
 <p>BOOK</p> <p>Angular and Machine Learning Pocket Primer</p> <p>1</p>	 <p>BOOK</p> <p>Angular Essentials: The Essential Guide to Learn...</p> <p>39</p>	 <p>BOOK</p> <p>Progressive Web Apps with Angular: Create Responsive...</p> <p>6</p>		

Business & Leadership for Enriched Web Development with Angular 11 Optional

 <p>COURSE</p> <p>Developing and Supporting an Agile Mindset</p> <p>1064</p>	 <p>COURSE</p> <p>Encouraging Team Communication and...</p> <p>1542</p>	 <p>COURSE</p> <p>The Essential Role of the Agile Product Owner</p> <p>276</p>	 <p>COURSE</p> <p>Using Strategic Thinking to Consider the Big Picture</p> <p>599</p>	 <p>COURSE</p> <p>Getting to the Root of a Problem</p> <p>926</p>
 <p>COURSE</p> <p>Unleashing Personal and Team Creativity</p> <p>844</p>	 <p>COURSE</p> <p>Contributing as a Virtual Team Member</p> <p>2636</p>	 <p>COURSE</p> <p>Developing a Growth Mindset</p> <p>2382</p>	 <p>COURSE</p> <p>Developing a Successful Team</p> <p>584</p>	 <p>COURSE</p> <p>Reaching Sound Conclusions</p> <p>275</p>