30-Day Angular Learning Roadmap (Frontend-Only)

Week 1: Angular Basics and Component Architecture

Day 1

- Concept to Prepare:

Install Angular, understand project structure

- How to Implement in Project:

Install Angular CLI, create new project with routing and SCSS. Explore Angular folder structure.

Day 2

- Concept to Prepare:

Components and Templates

- How to Implement in Project:

Create Header and Sidebar components using Angular CLI. Bind static data using interpolation.

Day 3

- Concept to Prepare:

Structural Directives

- How to Implement in Project:

Use *ngIf and *ngFor in TaskListComponent to display list of mock tasks.

Day 4

- Concept to Prepare:

Event Binding

- How to Implement in Project:

Add click events to toggle task status. Create button interactions.

Day 5

- Concept to Prepare:

Property Binding

- How to Implement in Project:

Disable submit button based on input. Create a basic task form.

Day 6

- Concept to Prepare:

Two-way Binding

- How to Implement in Project:

Use [(ngModel)] for binding input values. Import FormsModule.

Day 7

- Concept to Prepare:

Input/Output

- How to Implement in Project:

Use @Input and @Output to create TaskItemComponent and emit delete actions.

Week 2: Services, Routing, Reactive Forms

Day 8

- Concept to Prepare:

Angular Routing

- How to Implement in Project:

Set up routes and create TaskDetailsComponent with RouterModule.

Day 9

- Concept to Prepare:

Services and Dependency Injection

- How to Implement in Project:

Create TaskService to manage tasks. Inject it in components.

Day 10

- Concept to Prepare:

In-Memory Web API

- How to Implement in Project:

Install and configure in-memory-web-api for CRUD simulation.

Day 11

- Concept to Prepare:

Reactive Forms - Part 1

- How to Implement in Project:

Build Reactive Form using FormGroup and FormControl for task creation.

Day 12

- Concept to Prepare:

Reactive Forms - Part 2

- How to Implement in Project:

Add form validation and error messages.

Day 13

- Concept to Prepare:

HttpClient Integration

- How to Implement in Project:

Use TaskService to perform CRUD operations using HttpClient.

Day 14

- Concept to Prepare:

Child Routing

- How to Implement in Project:

Set up nested route for editing task at /tasks/:id/edit.

Week 3: State Management, Filtering, UI Enhancements

Day 15

- Concept to Prepare:

Angular Pipes

- How to Implement in Project:

Use built-in and custom pipes for formatting task data.

Day 16

- Concept to Prepare:

Lifecycle Hooks

- How to Implement in Project:

Use ngOnInit to fetch data from service.

Day 17

- Concept to Prepare:

Modular Architecture

- How to Implement in Project:

Refactor shared/common parts to CoreModule and SharedModule.

Day 18

- Concept to Prepare:

State Management with RxJS

- How to Implement in Project:

Use BehaviorSubject in TaskStoreService to share task state.

Day 19

- Concept to Prepare:

Filtering/Search

- How to Implement in Project:

Implement search and filtering by task status.

Day 20

- Concept to Prepare:

Pagination and Sorting

- How to Implement in Project:

Add manual pagination and sorting tasks by priority.

Day 21

- Concept to Prepare:

Loading and Empty States

- How to Implement in Project:

Add loader and empty state UI to task list.

Week 4: Advanced Features and Deployment

Day 22

- Concept to Prepare:

Authentication and Guards

- How to Implement in Project:

Create login page and AuthService. Use CanActivate for route protection.

Day 23

- Concept to Prepare:

Lazy Loading

- How to Implement in Project:

Move workflows feature to separate lazy-loaded module.

Day 24

- Concept to Prepare:

Role-based Access

- How to Implement in Project:

Simulate roles and protect routes/components accordingly.

Day 25

- Concept to Prepare:

Interceptors

- How to Implement in Project:

Create HTTP Interceptor to add mock JWT token.

Day 26

- Concept to Prepare:

Global Error Handling

- How to Implement in Project:

Handle API errors globally and show toast messages.

Day 27

- Concept to Prepare:

Animations

- How to Implement in Project:

Add task entry/removal animation using Angular Animation API.

Day 28

- Concept to Prepare:

UI with Angular Material

- How to Implement in Project:

Use MatCard, MatToolbar, MatSnackBar to enhance UI.

Day 29

- Concept to Prepare:

Build and Deploy

- How to Implement in Project:

Use ng build --prod and deploy to Firebase/Netlify/Vercel.

Day 30

- Concept to Prepare:

Final Testing & Documentation

- How to Implement in Project:

Test full app, write README, push to GitHub, record walkthrough.