Angular Basics and MVC Architecture

KOENIG
step forward

Introduction to Angular and Its Ecosystem:

- Overview of Angular’s features and benefits
- Key concepts: Components, Directives, Services, and Modules
- Setting up an Angular project with the Angular CLI

Overview of Angular's features and benefits

Angular is an application design framework and development platform for creating efficient and sophisticated single page application.

Angular is a popular open-source web application framework developed by Google and a community of individual developers and corporations.

It is designed to simplify the process of building dynamic, single page web applications and provides a comprehensive set of tools and features for front-end web development.

Angular is a framework for building client applications using HTML, CSS, and JavaScript / TypeScript.

Overview of Angular's features and benefits

In short, Angular is…

- TypeScript-based, open-source framework developed and maintained by Google for building scalable and dynamic web applications.
- Complete front-end framework with built-in features like component-based architecture, dependency injection, routing, and state management.

Overview of Angular's features and benefits

Key Features of Angular

- Component-Based Architecture

  Angular applications are built using reusable components, making the code modular and maintainable.

- Two-Way Data Binding

  Synchronizes data between the UI and the component class, reducing boilerplate code.

- Dependency Injection (DI)

  Enhances modularity and testability by managing service dependencies efficiently.

- Directives & Templates

  Built-in and custom directives allow dynamic UI manipulation and code reuse.

- Routing & Navigation

  The Angular Router enables single-page applications (SPA) with lazy loading to improve performance.

Overview of Angular’s features and benefits

Key Features of Angular

- Reactive Forms & Template-Driven Forms

  Supports form validation and dynamic form controls with powerful form-handling techniques.

- HTTP Client Module

  Built-in module to handle API requests with Observables (RxJS).

- RxJS & State Management

  Uses Reactive Extensions (RxJS) for handling asynchronous operations like HTTP calls and event handling.

  State management solutions like NgRx, Akita, and NGXS provide centralized state control.

- Built-in Security & Performance Enhancements

  Sanitization for security against XSS (Cross-Site Scripting) attacks.

  Ahead-of-Time (AOT) Compilation improves loading speed.

Overview of Angular's features and benefits

Angular Ecosystem

- The Angular ecosystem consists of tools and libraries that enhance development efficiency

Tool/Library	Purpose
Angular CLI	Automates project setup, builds, and testing.
RxJS	Handles reactive programming and asynchronous operations.
Angular Material	Pre-built UI components for faster development.
NgRx	State management using Redux principles.
Jasmine & Karma	Unit testing framework and test runner.

Installation Procedure of Angular

The Angular installation process involves a few key steps, primarily centered around Node.js, npm (or yarn), and the Angular CLI.

1. Install Node.js and npm:

Node.js provides the JavaScript runtime environment.

npm (Node Package Manager) is used to install and manage Angular and its dependencies.

Procedure:

- Go to the official Node.js website: https://nodejs.org/
- Download the LTS (Long Term Support) version, which is generally recommended for stability.
- Run the installer and follow the on-screen instructions.
- Verify the installation by opening your terminal or command prompt and running the following commands:
  - node -v (to check the Node.js version)
  - npm -v (to check the npm version)

Installation Procedure of Angular

The Angular installation process involves a few key steps, primarily centered around Node.js, npm (or yarn), and the Angular CLI.

2. Install the Angular CLI:

The Angular CLI (Command Line Interface) is a tool that simplifies Angular development.

It allows you to create new Angular projects, generate components, run tests, and more.

Procedure:

- Open your terminal or command prompt.
- Run the following command to install the Angular CLI globally:
- npm install -g @angular/cli
- This command installs the latest stable version of the Angular CLI.
- Verify the installation by running:
- ng --version

Installation Procedure of Angular

The Angular installation process involves a few key steps, primarily centered around Node.js, npm (or yarn), and the Angular CLI.

3. Create a New Angular Project:

- In your terminal or command prompt, navigate to the directory where you want to create your project.
- Run the following command, replacing "my-app" with your desired project name:

  ng new my-app
- The CLI will prompt you with a few questions:
  - "Would you like to add Angular routing?" (Choose "y" for yes or "n" for no)
  - "Which stylesheet format would you like to use?" (Choose your preferred format, such as CSS, SCSS, or Sass)
- The CLI will then create a new Angular project with the specified name and configuration.

Installation Procedure of Angular

The Angular installation process involves a few key steps, primarily centered around Node.js, npm (or yarn), and the Angular CLI.

4. Run the Angular Application:

- Navigate to your project directory:

  cd my-app
- Run the following command to start the development server:

  ng serve --open or ng serve -o
- This command will build your application and open it in your default web browser at http://localhost:4200.
- Any changes you make to your code will be automatically reflected in the browser.

Key concepts: Components, Directives, Services, and Modules

- Angular is built on a modular and component-based architecture, allowing developers to create scalable applications.
- The four fundamental concepts in Angular are Components, Directives, Services, and Modules.
- Components
  - A Component is the basic building block of an Angular application. It controls a part of the UI and consists of:
    - Template (HTML): Defines the UI.
    - Class (TypeScript): Contains business logic and data.
    - Styles (CSS/SCSS): Defines the appearance.

Key concepts: Components, Directives, Services, and Modules

- Directives
  - Directives modify the behavior or appearance of elements in the DOM. They are classified into:
  - Structural Directives (Modify DOM structure)
    - *ngIf – Conditionally renders an element.
    - *ngFor – Loops through an array to generate elements.
    - *ngSwitch – Implements switch-case logic in templates.
  - Attribute Directives (Modify element appearance or behavior)
    - ngClass – Adds classes dynamically.
    - ngStyle – Applies styles dynamically.

Key concepts: Components, Directives, Services, and Modules

- Services
  - Services are used to share logic across components, such as data fetching, business logic, or API calls.
  - They are injected using Dependency Injection (DI).
- Modules
  - Modules help organize the application by grouping related components, directives, and services.
  - Every Angular app has at least one module: AppModule (app.module.ts).