ANGULAR – COMPLETE COURSE SYLLABUS (BEGINNER TO ADVANCED)

🔲 Module 1: Introduction to Angular

- What is Angular?
- Evolution from AngularJS → Angular 2+ → Angular 16+
- Key Features & Architecture
- Angular vs React vs Vue.js
- Understanding SPA (Single Page Applications)
- Installing Node.js & NPM
- Installing Angular CLI
- Creating your first Angular app
- Project structure walkthrough
- Running & building Angular apps

🔲 Module 2: TypeScript Fundamentals for Angular

- Introduction to TypeScript
- Variables, Data Types & Type Inference
- Functions & Arrow Functions
- Classes, Interfaces, Inheritance
- Modules & Namespaces
- Async/Await & Promises
- Enums, Generics
- Decorators & Metadata
- Strict TypeScript Mode in Angular

🔲 Module 3: Angular Components & Data Binding

- What are Components?
- Creating components using Angular CLI
- Component Decorator & Metadata
- Component Lifecycle Hooks
- Data Binding in Angular
  - Interpolation
  - Property Binding
  - Event Binding
  - Two-way Binding (ngModel)
- Working with Child & Parent Components (@Input, @Output)
- ViewChild & ViewChildren
- Content Projection (ng-content)

🔲 Module 4: Standalone Components (Angular 16+)

- What are standalone components?
- Creating standalone components, directives & pipes
- Removing NgModules
- Using imports array
- Routing with standalone components
- Migrating old apps to standalone

🔲 Module 5: Directives & Pipes

Directives

- Structural Directives: *ngIf, *ngFor, *ngSwitch
- Attribute Directives: [ngClass], [ngStyle]
- Writing Custom Directives

Pipes

- Built-in Pipes: DatePipe, CurrencyPipe, JsonPipe, SlicePipe etc.
- Chaining Pipes
- Creating Custom Pipes

🔲 Module 6: Services & Dependency Injection

- What are Services?
- Creating & Injecting Services
- Hierarchical Dependency Injection
- Using HttpClientModule
- Making API Requests (GET, POST, PUT, DELETE)
- Error Handling
- Interceptors (AuthInterceptor, LoggingInterceptor)
- Environment variables

🔲 Module 7: Angular Routing & Navigation

- Setting up Routing
- RouterModule & Route Configuration
- RouterLink, RouterLinkActive
- Route Params & Query Params
- Child Routes / Nested Routes
- Lazy Loading & Code Splitting
- Route Guards
  - CanActivate, CanDeactivate
  - Auth & Role Guards
- Handling 404 Pages & Redirection
- Modern Angular Routing (v17+)
  - Deferrable Views
  - Functional Guards

🔲 Module 8: Angular Forms

Template-Driven Forms

- FormsModule
- ngModel
- Validation & Error Handling

Reactive Forms

- ReactiveFormsModule
- FormGroup, FormControl, FormArray
- FormBuilder
- Custom Validators
- Async Validators
- Dynamic Forms

🔲 Module 9: RxJS & Reactive Programming

- What is RxJS?
- Observable vs Promise
- Subjects, BehaviorSubject, ReplaySubject
- Operators (map, filter, tap, switchMap, debounceTime, mergeMap, concatMap)
- Error Handling: catchError, retry
- Subscription Management
- Hot vs Cold Observables

Module 10: Angular Signals (Angular 16+)
- Introduction to Signals
- Writable, Computed & Effect Signals
- Signals vs RxJS
- Managing UI state with Signals
- Using Signals with Components & Templates
- Signals + Services
- Migration guide for Signals

Module 11: State Management (NgRx)
- Why State Management?
- Redux Principles
- NgRx Store
- Actions, Reducers
- Effects (side effects)
- Selectors
- NgRx Entity
- NgRx DevTools
- NgRx with Signals (latest pattern)

Module 12: HTTP, APIs & Interceptors
- HttpClient Deep Dive
- API Integration with real backend
- Global Error Handling
- Http Interceptors
- Token Interceptor (JWT)
- Retry & caching strategies
- File upload & download

Module 13: UI Development with Angular Material
- Installing Angular Material
- Material Themes
- Material Components:
  - Buttons, Cards, Forms, Tables, Dialog, Snackbar, Tabs
- Custom Styling
- Responsive Layout using Flex Layout
- Material Icon & Animations

Module 14: Authentication & Security
- JWT Authentication
- Login / Signup Implementation

- Storing and refreshing tokens
- Role-based Access Control
- Securing routes with Guards
- Preventing XSS, CSRF, Clickjacking
- Security best practices

Module 15: Performance Optimization
- Lazy Loading Modules & Components
- OnPush Change Detection
- trackBy in *ngFor
- Pure Pipes
- Optimizing API Calls using RxJS
- Preloading Strategies
- Angular Signals for performance
- Bundle Optimization

Module 16: Testing in Angular
- Unit Testing with Jasmine & Karma
- Component Testing
- Service Testing
- Mocking HttpClient
- Spy, TestBed & Fixtures
- Integration Testing
- End-to-End Testing with Cypress

Module 17: Angular PWA (Progressive Web Apps)
- Installing Angular PWA
- Service Workers
- Offline Support
- Web Manifest
- Add to Home Screen
- Background Sync

Module 18: Monorepo Development (Nx)
- Introduction to Nx Workspace
- Creating libraries
- Managing multiple Angular apps
- Code sharing across projects
- Module federation basics

Module 19: Deployment & DevOps (Optional)
- Building Angular for Production
- Environment Configurations
- Deploying on:
  - Firebase
  - Netlify
  - Vercel
  - AWS S3 + CloudFront
- Dockerizing Angular Applications
- Setting Up CI/CD Pipelines (GitHub Actions)

📘 **Module 20: Real-World Project (Capstone Project)**

Build a Complete Enterprise-Level Angular Application:

- Authentication + Authorization
- Role-based Dashboard
- CRUD Operations with API
- Pagination, Sorting & Filtering
- Angular Material UI
- NgRx / Signals based state management
- Route Guards
- Error Handling
- Deployment to Cloud
- Performance Optimization