Angular Basics:

Overview of Angular framework and its features.

Difference between AngularJS and Angular.

Angular CLI and its benefits.

Understanding Angular project structure and files.

Introduction to TypeScript and its benefits.

Key TypeScript features like static typing, interfaces, classes, and modules.

How TypeScript is used in Angular development.

Angular Components and Directives:

Creating and using components.

Component lifecycle hooks and their uses.

Understanding directives (structural and attribute) and their differences.

Templates and Data Binding:

Using interpolation and property binding.

Event binding and handling user input.

Two-way data binding using ngModel.

Understanding the concept of DI and its benefits.

Implementing DI in Angular using providers.

Interceptor

Directives (Custom Directives)

Pipes

Subjects

Promise vs Observable

Environment files

Component to component communication

Creating and using services for data sharing and business logic.

Making HTTP requests using Angular's HttpClient module.

Handling HTTP responses and error handling.

Setting up and configuring routes in Angular.

Lazy loading modules for improved performance.

Route guards and resolving data before navigation.

Template-driven forms and reactive forms.

Form controls, validators, and custom validation.

Handling form submission and displaying validation errors.

Introduction to state management and its importance.

Overview of Angular's built-in state management with RxJS.

Alternatives like NgRx and Redux for more complex state management

Writing unit tests and end-to-end (E2E) tests in Angular.

Testing tools like Jasmine, Karma, and Protractor.

Testing strategies and best practices.

Techniques for improving Angular application performance.

Lazy loading, code splitting, and preloading modules.

Change detection strategies and OnPush change detection.

Following Angular style guide and best practices.

Component architecture and code organization.

Separation of concerns and reusable components.