

Angular 15 Coding Challenges: Multiple Choice Questions (MCQs)

1. What command is used to create a new Angular project?

- A) `ng generate`
- B) `ng new`
- C) `ng create`
- D) `ng start`
- **Answer:** B) `ng new`

2. How can you generate a new component in Angular using CLI?

- A) `ng generate component <component-name>`
- B) `ng add component <component-name>`
- C) `ng create component <component-name>`
- D) `ng new component <component-name>`
- **Answer:** A) `ng generate component <component-name>`

3. Which decorator is used to define a component in Angular?

- A) `@NgModule`
- B) `@Component`
- C) `@Injectable`
- D) `@Directive`
- **Answer:** B) `@Component`

4. What is the purpose of the `ngOnInit` lifecycle hook?

- A) To initialize component styles
- B) To initialize component templates
- C) To perform component initialization logic after Angular has set input properties
- D) To destroy the component
- **Answer:** C) To perform component initialization logic after Angular has set input properties

5. How do you bind a property from the component to the template in Angular?

- A) `{{property}}`
- B) `[property]="value"`
- C) `[(property)]="value"`
- D) `@property="value"`
- **Answer:** A) `{{property}}`

6. Which of the following is used for two-way data binding in Angular?

- A) `{{property}}`
- B) `[property]="value"`
- C) `[(ngModel)]="value"`
- D) `@property="value"`
- **Answer:** C) `[(ngModel)]="value"`

7. What is the correct syntax for event binding in Angular?

- A) `(event)="function"`
- B) `[event]="function"`
- C) `{{event}}="function"`
- D) `@event="function"`
- **Answer:** A) `(event)="function"`

8. Which command is used to run an Angular application locally?

- A) `ng run`
- B) `ng serve`
- C) `ng start`
- D) `ng launch`
- **Answer:** B) `ng serve`

9. In Angular, how can you create a service?

- A) `ng create service <service-name>`
- B) `ng add service <service-name>`
- C) `ng new service <service-name>`

- D) `ng generate service <service-name>`
- **Answer:** D) `ng generate service <service-name>`

10. Which Angular directive is used to display a list of items?

- A) `*ngFor`
- B) `*ngIf`
- C) `*ngList`
- D) `*ngEach`
- **Answer:** A) `*ngFor`

11. How can you conditionally display elements in Angular?

- A) `*ngFor`
- B) `*ngIf`
- C) `*ngShow`
- D) `*ngDisplay`
- **Answer:** B) `*ngIf`

12. What is Angular's default change detection strategy?

- A) `OnPush`
- B) `Default`
- C) `Manual`
- D) `Async`
- **Answer:** B) `Default`

13. How can you catch and handle errors in Angular services?

- A) Using `try-catch` blocks
- B) Using `errorHandler`
- C) Using `catchError` operator in RxJS
- D) Using `onError` method
- **Answer:** C) Using `catchError` operator in RxJS

14. Which method in Angular's HttpClient is used to send a GET request?

- A) `httpGet()`
- B) `fetch()`
- C) `get()`
- D) `request()`
- **Answer:** C) `get()`

15. What is the purpose of Angular's `NgModule` decorator?

- A) To define a component
- B) To define a directive
- C) To define an Angular module
- D) To define a service
- **Answer:** C) To define an Angular module

16. What is the correct way to include a third-party library in an Angular project?

- A) Use `npm install <library>`
- B) Add the library URL to the index.html
- C) Use `ng add <library>`
- D) Import the library in `app.module.ts`
- **Answer:** A) Use `npm install <library>`

17. Which Angular module is commonly used for routing?

- A) `HttpClientModule`
- B) `FormsModule`
- C) `RouterModule`
- D) `CommonModule`
- **Answer:** C) `RouterModule`

18. How do you import a module into an Angular application?

- A) `@NgModule`
- B) `@Module`

- C) ``import { ModuleName } from 'module-path';``
- D) ``import { ModuleName } from 'angular-core';``
- ****Answer:**** C) ``import { ModuleName } from 'module-path';``

19. What command is used to build an Angular project for production?

- A) ``ng build --prod``
- B) ``ng build``
- C) ``ng production``
- D) ``ng deploy``
- ****Answer:**** A) ``ng build --prod``

20. Which operator in RxJS is used to transform the items emitted by an Observable?

- A) ``map``
- B) ``filter``
- C) ``catchError``
- D) ``debounce``
- ****Answer:**** A) ``map``

21. How can you subscribe to an Observable in Angular?

- A) ``observable.subscribe()``
- B) ``observable.listen()``
- C) ``observable.observe()``
- D) ``observable.watch()``
- ****Answer:**** A) ``observable.subscribe()``

22. What is the role of the ``@Input`` decorator in Angular?

- A) To pass data from parent to child components
- B) To pass data from child to parent components
- C) To bind an event in a component
- D) To create a service
- ****Answer:**** A) To pass data from parent to child components

23. Which lifecycle hook is called after the component's view has been fully initialized?

- A) `ngOnInit`
- B) `ngAfterViewInit`
- C) `ngAfterContentInit`
- D) `ngOnChanges`
- **Answer:** B) `ngAfterViewInit`

24. What is the purpose of Angular's `RouterLink` directive?

- A) To define a route
- B) To navigate to a route
- C) To protect a route
- D) To lazy load a route
- **Answer:** B) To navigate to a route

25. How can you perform form validation in Angular?

- A) Using form controls and validators
- B) Using JavaScript functions
- C) Using CSS classes
- D) Using HTML attributes
- **Answer:** A) Using form controls and validators

26. Which RxJS operator is used to combine multiple Observables into one?

- A) `map`
- B) `merge`
- C) `filter`
- D) `concat`
- **Answer:** B) `merge`

27. How do you define a route in Angular?

- A) Using `@RouteConfig`

- B) Using `RouteModule`
- C) Using `RouterModule.forRoot()`
- D) Using `@RouterConfig`
- **Answer:** C) Using `RouterModule.forRoot()`

28. What is Angular's `FormBuilder` used for?

- A) To create form controls and form groups
- B) To validate forms
- C) To submit forms
- D) To reset forms
- **Answer:** A) To create form controls and form groups

29. How do you access a child component's methods from a parent component?

- A) Using `@Output` decorator
- B) Using `@ViewChild` decorator
- C) Using `@Input` decorator
- D) Using `@Injectable` decorator
- **Answer:** B) Using `@ViewChild` decorator

30. Which Angular service is used for making HTTP requests?

- A) `Http`
- B) `HttpClient`
- C) `HttpService`
- D) `HttpRequest`
- **Answer:** B) `HttpClient`

31. What is Angular's `async` pipe used for?

- A) To handle asynchronous operations in templates
- B) To transform data
- C) To format dates
- D) To handle errors

- **Answer:** A) To handle asynchronous operations in templates

32. Which Angular CLI command is used to update Angular and its dependencies?

- A) `ng update`

- B) `ng upgrade`

- C) `ng install`

- D) `ng new`

- **Answer:** A) `ng update`

33. How can you protect a route in Angular?

- A) Using `CanActivate` guard

- B) Using `CanDeactivate` guard

- C) Using `Resolve` guard

- D) Using

`RouteGuard`

- **Answer:** A) Using `CanActivate` guard

34. What is the purpose of Angular's `Injector`?

- A) To manage dependency injection

- B) To manage component lifecycle

- C) To manage routing

- D) To manage state

- **Answer:** A) To manage dependency injection

35. How do you create a custom directive in Angular?

- A) Using `@Directive` decorator

- B) Using `@Component` decorator

- C) Using `@Pipe` decorator

- D) Using `@NgModule` decorator

- **Answer:** A) Using `@Directive` decorator

36. What is the correct way to unsubscribe from an Observable?

- A) `observable.unsubscribe()`
- B) `subscription.unsubscribe()`
- C) `observer.unsubscribe()`
- D) `subject.unsubscribe()`
- **Answer:** B) `subscription.unsubscribe()`

37. How can you share data between sibling components in Angular?

- A) Using a shared service
- B) Using `@Input` decorator
- C) Using `@Output` decorator
- D) Using `@ViewChild` decorator
- **Answer:** A) Using a shared service

38. What is Angular's `HttpInterceptor` used for?

- A) To handle HTTP errors
- B) To transform HTTP requests and responses
- C) To cache HTTP responses
- D) To retry failed HTTP requests
- **Answer:** B) To transform HTTP requests and responses

39. Which command is used to test an Angular application?

- A) `ng test`
- B) `ng serve`
- C) `ng build`
- D) `ng run`
- **Answer:** A) `ng test`

40. How do you create a custom pipe in Angular?

- A) Using `@Pipe` decorator

- B) Using `@Component` decorator
- C) Using `@Directive` decorator
- D) Using `@NgModule` decorator
- **Answer:** A) Using `@Pipe` decorator

41. What is the role of Angular's `NgZone` service?

- A) To manage change detection
- B) To manage routing
- C) To manage HTTP requests
- D) To manage state
- **Answer:** A) To manage change detection

42. How can you optimize change detection in Angular?

- A) Using `OnPush` change detection strategy
- B) Using `Default` change detection strategy
- C) Using `Manual` change detection strategy
- D) Using `Async` change detection strategy
- **Answer:** A) Using `OnPush` change detection strategy

43. What is the purpose of Angular's `ElementRef`?

- A) To access the DOM element
- B) To manage component state
- C) To manage routing
- D) To manage HTTP requests
- **Answer:** A) To access the DOM element

44. How do you enable lazy loading in Angular?

- A) Using `loadChildren` in route configuration
- B) Using `loadComponent` in route configuration
- C) Using `loadModule` in route configuration
- D) Using `lazyLoad` in route configuration

- **Answer:** A) Using `loadChildren` in route configuration

45. How can you handle errors globally in Angular?

- A) Using `ErrorHandler` class
- B) Using `catchError` operator
- C) Using `try-catch` blocks
- D) Using `onError` method

- **Answer:** A) Using `ErrorHandler` class

46. Which Angular directive is used to dynamically add or remove DOM elements?

- A) `*ngIf`
- B) `*ngFor`
- C) `*ngSwitch`
- D) `*ngClass`

- **Answer:** A) `*ngIf`

47. How do you update the state in Angular's `OnPush` change detection strategy?

- A) By changing the reference of the object
- B) By modifying the object directly
- C) By using `setState`
- D) By using `updateState`

- **Answer:** A) By changing the reference of the object

48. What is Angular's `Renderer2` used for?

- A) To safely manipulate the DOM
- B) To manage component state
- C) To manage HTTP requests
- D) To manage routing

- **Answer:** A) To safely manipulate the DOM

49. How can you create a reactive form in Angular?

- A) Using `FormBuilder`
 - B) Using `FormControl`
 - C) Using `FormGroup`
 - D) All of the above
- **Answer:** D) All of the above

50. How do you define a child route in Angular?

- A) Using `children` property in route configuration
 - B) Using `childRoutes` property in route configuration
 - C) Using `subRoutes` property in route configuration
 - D) Using `nestedRoutes` property in route configuration
- **Answer:** A) Using `children` property in route configuration

51. Which Angular module is required for form handling?

- A) `FormsModule`
 - B) `HttpClientModule`
 - C) `RouterModule`
 - D) `CommonModule`
- **Answer:** A) `FormsModule`

52. How do you handle asynchronous operations in Angular templates?

- A) Using `async` pipe
 - B) Using `sync` pipe
 - C) Using `asyncAwait` pipe
 - D) Using `promise` pipe
- **Answer:** A) Using `async` pipe

53. What is the purpose of Angular's `ngClass` directive?

- A) To conditionally add or remove CSS classes
- B) To conditionally add or remove HTML elements
- C) To dynamically change the element's content

- D) To bind data to the template
- **Answer:** A) To conditionally add or remove CSS classes

54. How can you pass data from a child component to a parent component?

- A) Using `@Output`` decorator and `EventEmitter``
- B) Using `@Input`` decorator
- C) Using `@ViewChild`` decorator
- D) Using `@Child`` decorator
- **Answer:** A) Using `@Output`` decorator and `EventEmitter``

55. Which Angular directive is used for structural changes in the DOM?

- A) `*ngIf``
- B) `ngStyle``
- C) `ngClass``
- D) `ngBind``
- **Answer:** A) `*ngIf``

56. How can you implement pagination in an Angular application?

- A) Using `*ngFor`` with slice pipe
- B) Using `*ngIf``
- C) Using `ngClass``
- D) Using `ngSwitch``
- **Answer:** A) Using `*ngFor`` with slice pipe

57. What is Angular's `TemplateRef`` used for?

- A) To reference a template
- B) To reference a component
- C) To reference a directive
- D) To reference a service
- **Answer:** A) To reference a template

58. Which Angular directive is used for dynamic style binding?

- A) `ngStyle`
- B) `ngClass`
- C) `ngBind`
- D) `ngSwitch`
- ****Answer:**** A) `ngStyle`

59. How can you prefetch data before navigating to a route in Angular?

- A) Using `Resolve` guard
- B) Using `CanActivate` guard
- C) Using `CanDeactivate` guard
- D) Using `Prefetch` guard
- ****Answer:**** A) Using `Resolve` guard

60. What is the purpose of Angular's `ng-content`?

- A) To project content into a component
- B) To bind data to the template
- C) To conditionally add or remove CSS classes
- D) To dynamically change the element's content
- ****Answer:**** A) To project content into a component

61. How do you enable Angular Material in a project?

- A) Using `ng add @angular/material`
- B) Using `ng install @angular/material`
- C) Using `ng generate @angular/material`
- D) Using `ng build @angular/material`
- ****Answer:**** A) Using `ng add @angular/material`

62. How do you implement route guards in Angular?

- A) By creating a service that implements `CanActivate` or `CanDeactivate` interfaces
- B) By creating a component that implements `CanActivate` or `CanDeactivate` interfaces

- C) By creating a directive that implements `CanActivate` or `CanDeactivate` interfaces
- D) By creating a pipe that implements `CanActivate` or `CanDeactivate` interfaces
- **Answer:** A) By creating a service that implements `CanActivate` or `CanDeactivate` interfaces

63. What is Angular's `ViewContainerRef` used for?

- A) To create and insert views dynamically
- B) To access the component's view
- C) To access the component's template
- D) To access the component's styles
- **Answer:** A) To create and insert views dynamically

64. Which Angular lifecycle hook is called when the component is destroyed?

- A) `ngOnDestroy`
- B) `ngAfterViewInit`
-
- C) `ngAfterContentInit`
- D) `ngOnChanges`
- **Answer:** A) `ngOnDestroy`

65. How can you update the URL in Angular without reloading the page?

- A) Using `Location` service
- B) Using `Router` service
- C) Using `HttpClient` service
- D) Using `ActivatedRoute` service
- **Answer:** B) Using `Router` service

66. What is the purpose of Angular's `RendererFactory2`?

- A) To create a `Renderer2` instance
- B) To create a `ViewContainerRef` instance
- C) To create a `TemplateRef` instance

- D) To create a ChangeDetectorRef instance
- **Answer:** A) To create a Renderer2 instance

67. How do you handle form submission in Angular?

- A) Using `<ngSubmit>` directive
- B) Using `<ngForm>` directive
- C) Using `<ngModel>` directive
- D) Using `<ngControl>` directive
- **Answer:** A) Using `<ngSubmit>` directive

68. Which Angular module provides support for reactive forms?

- A) `ReactiveFormsModule`
- B) `FormsModule`
- C) `HttpClientModule`
- D) `RouterModule`
- **Answer:** A) `ReactiveFormsModule`

69. How can you track changes in a list of items in Angular?

- A) Using `trackBy` function in `*ngFor`
- B) Using `*ngIf`
- C) Using `<ngClass>`
- D) Using `<ngStyle>`
- **Answer:** A) Using `trackBy` function in `*ngFor`

70. What is the purpose of Angular's `ChangeDetectorRef`?

- A) To control change detection
- B) To control routing
- C) To control HTTP requests
- D) To control state management
- **Answer:** A) To control change detection

71. How can you implement internationalization (i18n) in Angular?

- A) Using Angular's i18n tools and libraries
 - B) Using third-party i18n libraries
 - C) Using custom i18n services
 - D) All of the above
- **Answer:** D) All of the above

72. Which Angular directive is used to dynamically switch between templates?

- A) `*ngIf``
 - B) `ngClass``
 - C) `ngStyle``
 - D) `*ngSwitch``
- **Answer:** D) `*ngSwitch``

73. How do you create a dynamic component in Angular?

- A) Using `ComponentFactoryResolver``
 - B) Using `NgModule``
 - C) Using `@Directive``
 - D) Using `Renderer2``
- **Answer:** A) Using `ComponentFactoryResolver``

74. What is the purpose of Angular's `Injector``?

- A) To provide dependency injection
 - B) To manage change detection
 - C) To manage routing
 - D) To manage HTTP requests
- **Answer:** A) To provide dependency injection

75. How can you debounce user input in Angular?

- A) Using RxJS `debounceTime`` operator
- B) Using `setTimeout``

- C) Using `Promise`
- D) Using `async` pipe
- **Answer:** A) Using RxJS `debounceTime` operator

76. What is Angular's `@ViewChildren` decorator used for?

- A) To query multiple child elements or directives
- B) To query a single child element or directive
- C) To query parent elements
- D) To query sibling elements
- **Answer:** A) To query multiple child elements or directives

77. How can you detect changes manually in Angular?

- A) Using `ChangeDetectorRef.detectChanges()`
- B) Using `NgZone.run()`
- C) Using `ApplicationRef.tick()`
- D) All of the above
- **Answer:** D) All of the above

78. What is Angular's `@HostBinding` decorator used for?

- A) To bind a property to a host element
- B) To bind an event to a host element
- C) To bind a class to a host element
- D) To bind a style to a host element
- **Answer:** A) To bind a property to a host element

79. Which Angular directive is used for dynamically applying CSS styles?

- A) `ngStyle`
- B) `ngClass`
- C) `ngBind`
- D) `ngSwitch`
- **Answer:** A) `ngStyle`

80. How can you pass data to a service in Angular?

- A) Using service methods
- B) Using `@Input`` decorator
- C) Using `@Output`` decorator
- D) Using `@ViewChild`` decorator
- ****Answer:**** A) Using service methods

81. What is Angular's `@ContentChild`` decorator used for?

- A) To query a projected content child element
- B) To query a projected content parent element
- C) To query a projected content sibling element
- D) To query a projected content host element
- ****Answer:**** A) To query a projected content child element

82. How can you configure multiple environments in Angular?

- A) Using `angular.json`` file
- B) Using `environment.ts`` files
- C) Using `app.module.ts`` file
- D) Using `main.ts`` file
- ****Answer:**** B) Using `environment.ts`` files

83. How can you create a singleton service in Angular?

- A) By providing the service in the ``root`` injector
- B) By providing the service in the ``AppModule``
- C) By providing the service in the ``NgModule``
- D) All of the above
- ****Answer:**** A) By providing the service in the ``root`` injector

84. What is the purpose of Angular's `@ContentChildren`` decorator?

- A) To query multiple projected content child elements

- B) To query a single projected content child element
- C) To query multiple projected content parent elements
- D) To query multiple projected content sibling elements
- ****Answer:** A) To query multiple projected content child elements**

85. How can you handle animations in Angular?

- A) Using Angular's animation library
- B) Using CSS animations
- C) Using JavaScript animations
- D) All of the above
- ****Answer:** D) All of the above**

86. What is Angular's `BrowserAnimationsModule` used for?

- A) To enable animations in Angular applications
- B) To enable routing in Angular applications
- C) To enable HTTP requests in Angular applications
- D) To enable forms in Angular applications
- ****Answer:** A) To enable animations in Angular applications**

87. How can you optimize an Angular application for production?

- A) Using Angular CLI `ng build --prod`
- B) Using Angular CLI `ng serve --prod`
- C) Using Angular CLI `ng test --prod`
- D) Using Angular CLI `ng run --prod`
- ****Answer:** A) Using Angular CLI `ng build --prod`**

88. What is Angular's `@HostListener` decorator used for?

- A) To listen to events on the host element
- B) To listen to events on child elements
- C) To listen to events on parent elements
- D) To listen to events on sibling elements

- **Answer:** A) To listen to events on the host element

89. How can you test Angular components?

- A) Using Jasmine and Karma
- B) Using Jest
- C) Using Mocha and Chai
- D) All of the above

- **Answer:** D) All of the above

90. How can you handle forms in Angular?

- A) Using reactive forms
- B) Using template-driven forms
- C) Using JavaScript forms
- D) Both A and B

- **Answer:** D) Both A and B

91. How can you perform HTTP GET requests in Angular?

- A) Using `HttpClient.get()`
- B) Using `Http.get()`
- C) Using `HttpRequest.get()`
- D) Using `HttpClientRequest.get()`

- **Answer:** A) Using `HttpClient.get()`

92. What is Angular's `RouterLinkActive` directive used for?

- A) To add a CSS class to an active route link
- B) To navigate to a route
- C) To define a route
- D) To lazy load a route

- **Answer:** A) To add a CSS class to an active route link

93. How can you handle route parameters in Angular?

- A) Using `ActivatedRoute`
- B) Using `Router`
- C) Using `HttpClient`
- D) Using `Renderer2`
- **Answer:** A) Using `ActivatedRoute`

94. What is the purpose of Angular's `@Pipe` decorator?

- A) To create custom pipes
- B) To create custom directives
- C) To create custom components
- D) To create custom modules
- **Answer:** A) To create custom pipes

95. How can you dynamically create components in Angular?

- A) Using `ComponentFactoryResolver`
- B) Using `Renderer2`
- C) Using `ViewContainerRef`
- D) Both A and C

- **Answer:** D) Both A and C

96. How can you handle form validation errors in Angular?

- A) Using `FormControl` and `FormGroup`
- B) Using `NgModel`
- C) Using `NgForm`
- D) Using `NgControl`
- **Answer:** A) Using `FormControl` and `FormGroup`

97. How can you implement lazy loading in Angular?

- A) Using `loadChildren` property in route configuration

- B) Using `loadComponent` property in route configuration
- C) Using `loadModule` property in route configuration
- D) Using `lazyLoad` property in route configuration
- **Answer:** A) Using `loadChildren` property in route configuration

98. What is Angular's `HttpParams` used for?

- A) To create query parameters for HTTP requests
- B) To create headers for HTTP requests
- C) To create body for HTTP requests
- D) To create path variables for HTTP requests
- **Answer:** A) To create query parameters for HTTP requests

99. How can you unit test Angular services?

- A) Using Jasmine and TestBed
- B) Using Jest
- C) Using Mocha and Chai
- D) All of the above
- **Answer:** D) All of the above

100. How can you implement a search feature in Angular?

- A) Using `*ngFor` with filter pipe
- B) Using `*ngIf`
- C) Using `ngClass`
- D) Using `ngStyle`
- **Answer:** A) Using `*ngFor` with filter pipe