Preguntas de Angular

Explicar la diferencia[[1]](#footnote-1) entre Angular[[2]](#footnote-2) y Angularjs[[3]](#footnote-3).

Qué hay que instalar en la PC para comenzar a desarrollar una aplicación de frontend en Angular.

Cuáles son los patrones de diseño relevantes en Angular[[4]](#footnote-4).

Qué es TypeScript[[5]](#footnote-5), [[6]](#footnote-6), [[7]](#footnote-7).

Qué es RxJS[[8]](#footnote-8), [[9]](#footnote-9).

Qué significa SPA[[10]](#footnote-10), [[11]](#footnote-11).

Qué hace la extensión vscode-icons de Visual Studio Code.

Qué hace la extensión ESLint de Visual Studio Code.

Qué significa que una aplicación sea responsiva (responsive[[12]](#footnote-12) en inglés).

Explicar la arquitectura de una aplicación en Angular[[13]](#footnote-13).

Qué es un componente en Angular[[14]](#footnote-14).

Qué es un template en Angular[[15]](#footnote-15).

Como se usan los componentes y templates en Angular[[16]](#footnote-16).

Cómo se genera un componente nuevo en Angular.

Qué archivos forman el componente generado en Angular.

Qué es interpolación en Angular[[17]](#footnote-17).

Qué son template statements en Angular[[18]](#footnote-18).

Qué hace el data binding en Angular[[19]](#footnote-19).

Cómo se usa data binding en HTML en Angular[[20]](#footnote-20).

Explicar los tipos de data binding que existen en Angular[[21]](#footnote-21).

Explicar la relación entre los tipos de data binding y sus targets en Angular[[22]](#footnote-22).

Qué es property binding en Angular[[23]](#footnote-23).

Qué hace el método ngOnInit()[[24]](#footnote-24).

Qué es binding de atributo, clase y estilo en Angular[[25]](#footnote-25).

Qué es binding de eventos en Angular[[26]](#footnote-26).

Qué es el binding bidireccional en Angular[[27]](#footnote-27).

Qué son las directivas en Angular[[28]](#footnote-28).

Qué son las variables de template en Angular[[29]](#footnote-29).

Qué son y para qué sirven los decoradores @[Input](https://angular.io/api/core/Input)() y @[Output](https://angular.io/api/core/Output)() en Angular[[30]](#footnote-30).

Qué es la directiva NgModel en Angular[[31]](#footnote-31), [[32]](#footnote-32).

Diferencia entre [(ngModel)] y [ngModel] para ligar un estado a una propiedad[[33]](#footnote-33).

Qué hace la directiva ngFor[[34]](#footnote-34).

Explicar los tipos primitivos de datos string, number y boolean[[35]](#footnote-35).

Explicar cómo se especifica el tipo array[[36]](#footnote-36).

Explicar el tipo any[[37]](#footnote-37).

Explicar el flag de compilador noImplicitAny[[38]](#footnote-38).

Explicar las anotaciones de tipo en variables[[39]](#footnote-39).

Qué son las funciones en JavaScript[[40]](#footnote-40), [[41]](#footnote-41), [[42]](#footnote-42), [[43]](#footnote-43), [[44]](#footnote-44).

Qué son las anotaciones de tipo de parámetros de una función[[45]](#footnote-45).

Qué son las anotaciones de tipo de retorno de una función [[46]](#footnote-46).

Qué es la asignación contextual de tipos de parámetros en las funciones anónimas[[47]](#footnote-47).

Explicar los tipos Object[[48]](#footnote-48).

Qué es una propiedad opcional de un objeto y cómo se especifica[[49]](#footnote-49).

Qué son los tipos de unión[[50]](#footnote-50).

Cómo y dónde se especifica el punto de entrada de la aplicación, es decir por dónde arranca[[51]](#footnote-51).

Qué es Angular Material[[52]](#footnote-52).

Qué es el operador de navegación segura, u operador de encadenamiento opcional en JavaScript[[53]](#footnote-53).

1. <https://www.campusmvp.es/recursos/post/las-10-principales-diferencias-entre-angularjs-y-angular.aspx> [↑](#footnote-ref-1)
2. <https://angular.io> [↑](#footnote-ref-2)
3. <https://angularjs.org/> [↑](#footnote-ref-3)
4. <https://angular-enterprise.com/en/ngpost/courses/design-patterns/> [↑](#footnote-ref-4)
5. <https://www.typescriptlang.org/> [↑](#footnote-ref-5)
6. <https://en.wikipedia.org/wiki/TypeScript> [↑](#footnote-ref-6)
7. <https://profile.es/blog/que-es-typescript-vs-javascript/> [↑](#footnote-ref-7)
8. <https://rxjs.dev/> [↑](#footnote-ref-8)
9. <https://angular.io/guide/rx-library> [↑](#footnote-ref-9)
10. <https://developer.mozilla.org/en-US/docs/Glossary/SPA> [↑](#footnote-ref-10)
11. <https://es.wikipedia.org/wiki/Single-page_application> [↑](#footnote-ref-11)
12. <https://en.wikipedia.org/wiki/Responsive_web_design> [↑](#footnote-ref-12)
13. <https://angular.io/guide/architecture> [↑](#footnote-ref-13)
14. <https://angular.io/guide/glossary#component> [↑](#footnote-ref-14)
15. <https://angular.io/guide/glossary#template> [↑](#footnote-ref-15)
16. <https://angular.io/guide/architecture-components> [↑](#footnote-ref-16)
17. <https://angular.io/guide/interpolation> [↑](#footnote-ref-17)
18. <https://angular.io/guide/template-statements> [↑](#footnote-ref-18)
19. <https://angular.io/guide/binding-syntax> [↑](#footnote-ref-19)
20. <https://angular.io/guide/binding-syntax#data-binding-and-html> [↑](#footnote-ref-20)
21. <https://angular.io/guide/binding-syntax#types-of-data-binding> [↑](#footnote-ref-21)
22. <https://angular.io/guide/binding-syntax#binding-types-and-targets> [↑](#footnote-ref-22)
23. <https://angular.io/guide/property-binding> [↑](#footnote-ref-23)
24. <https://angular.io/api/core/OnInit> [↑](#footnote-ref-24)
25. <https://angular.io/guide/attribute-binding> [↑](#footnote-ref-25)
26. <https://angular.io/guide/event-binding> [↑](#footnote-ref-26)
27. <https://angular.io/guide/two-way-binding> [↑](#footnote-ref-27)
28. <https://angular.io/guide/built-in-directives> [↑](#footnote-ref-28)
29. <https://angular.io/guide/template-reference-variables> [↑](#footnote-ref-29)
30. <https://angular.io/guide/inputs-outputs> [↑](#footnote-ref-30)
31. <https://angular.io/api/forms/NgModel#description> [↑](#footnote-ref-31)
32. <https://www.acontracorrientech.com/angular-entendiendo-la-ngmodel-directive/> [↑](#footnote-ref-32)
33. <https://stackoverflow.com/questions/42504918/difference-between-ngmodel-and-ngmodel-for-binding-state-to-property> [↑](#footnote-ref-33)
34. <https://angular.io/guide/built-in-directives#ngFor> [↑](#footnote-ref-34)
35. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#the-primitives-string-number-and-boolean> [↑](#footnote-ref-35)
36. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#arrays> [↑](#footnote-ref-36)
37. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#any> [↑](#footnote-ref-37)
38. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#noimplicitany> [↑](#footnote-ref-38)
39. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#type-annotations-on-variables> [↑](#footnote-ref-39)
40. <https://www.w3schools.com/js/js_functions.asp> [↑](#footnote-ref-40)
41. <https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Functions> [↑](#footnote-ref-41)
42. <https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/Function> [↑](#footnote-ref-42)
43. <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Functions> [↑](#footnote-ref-43)
44. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#functions> [↑](#footnote-ref-44)
45. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#parameter-type-annotations> [↑](#footnote-ref-45)
46. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#return-type-annotations> [↑](#footnote-ref-46)
47. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#anonymous-functions> [↑](#footnote-ref-47)
48. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#object-types> [↑](#footnote-ref-48)
49. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#optional-properties> [↑](#footnote-ref-49)
50. <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#union-types> [↑](#footnote-ref-50)
51. <https://angular.io/guide/bootstrapping> [↑](#footnote-ref-51)
52. <https://material.angular.io/> [↑](#footnote-ref-52)
53. <https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Operators/Optional_chaining> [↑](#footnote-ref-53)