

Q1. Pourquoi utiliser un backend RESTful dans une SPA Angular ?

→ C. Pour permettre une communication stateless et découplée avec le front-end

Q2. Différence entre ngIf et [hidden] ?

→ B. ngIf supprime l'élément du DOM, [hidden] le cache visuellement

Q3. Pourquoi ChangeDetectionStrategy.OnPush améliore les performances ?

→ C. Évite les recalculs inutiles si les références n'ont pas changé

Q4. Que permet le typage strict de TypeScript ?

→ B. Il génère des erreurs si les types ne sont pas précisés

Q5. Rôle du fichier polyfills.ts ?

→ B. Inclure des fonctionnalités JS modernes sur navigateurs anciens

Q6. Intérêt des Lazy Modules ?

→ C. Charger certaines parties de l'app seulement si nécessaire

Q7. Directive pour appliquer des classes CSS conditionnelles ?

→ B. [ngClass]

Q8. Avantage de ngOnInit() sur constructor() ?

→ D. Il est appelé après injection des dépendances

Q9. Que fait ce setter set price(val: number) ?

→ B. Il stocke le prix uniquement si val > 0

Q10. Comportement de {{ product?.title }} si product est undefined ?

→ C. Affiche vide sans planter

Q11. Rôle de constructor(@Inject(HttpClient) private http: HttpClient) ?

→ C. Injecte une instance de HttpClient

Q12. Que fait type Color = 'red' | 'green'; let c: Color = 'yellow'; ?

→ B. Erreur de type

Q13. Résultat de let price = "100"; price * 2 ?

→ A. 200

Q14. Que fait ngOnChanges() ?

→ B. Appelé si une @Input() change

Q15. Signification de [ngStyle]="{ color: getColor(price) }" ?

→ B. Applique dynamiquement un style inline

Q16. Résultat de forEach((item, i) => setTimeout(() => console.log(i), 0)) ?

→ A. 0 1 2

Q17. Effet de "optimization": true, "sourceMap": false ?

→ C. Optimise le build et cache le code source

Q18. Rôle de providedIn: 'root' ?

→ C. Rend le service globalement injectable une fois

Q19. Résultat de Object.freeze(obj); obj.a = 2 ?

→ B. 1

Q20. Que fait *ngIf="products.length; else empty" si products = [] ?

→ B. Affiche "Aucun produit"

Q21. constructor(private service: ProductService) {} sert à quoi ?

→ B. Injecter une instance du service

Q22. Rôle de @NgModule ?

→ B. Fournir les métadonnées pour compiler un module

Q23. Rôle de import { Component } from '@angular/core' ?

→ C. Utiliser le décorateur @Component

Q24. let i = index dans *ngFor sert à ?

→ A. Récupérer l'indice de la boucle

Q25. Que se passe-t-il si une propriété @Input() n'est pas initialisée ?

→ B. TypeScript génère une erreur sauf si ! est utilisé

Q26. Que fait list.map(x => x * 3); sans assignation ?

→ B. Crée un nouveau tableau mais ne l'utilise pas

Q27. Que se passe-t-il si img est undefined dans [src]="img" ?

→ D. L'image ne se charge pas, mais la page reste stable

Q28. Rôle de Angular CLI ?

→ B. Générer fichiers et structure automatiquement

Q29. À quoi sert le slice pipe ?

→ C. Extraire une portion d'une chaîne ou d'un tableau

Q30. Pourquoi BrowserModule est obligatoire dans AppModule ?

→ B. Pour afficher les composants dans le navigateur

Q31. Quel fichier contient la config globale du projet Angular ?

→ Réponse : angular.json

Q32. Commande pour créer un composant ?

→ Réponse : ng generate component

Q33. Directive pour afficher selon condition ?

→ Réponse : *ngIf

Q34. Nom du mécanisme de liaison de données dans Angular ?

→ Réponse : Data Binding

Q35. Classe pour appels HTTP ?

→ Réponse : `HttpClient`

Q36. Décorateur pour recevoir des données d'un parent ?

→ Réponse : `@Input()`

Q37. Rôle du dossier `assets/` ?

→ Réponse : Stocker les fichiers statiques

Q38. Type générique en TypeScript ?

→ Réponse : `any`

Q39. Rôle de `[ngStyle]` ?

→ Réponse : Appliquer dynamiquement un style inline

Q40. Quand est appelée `ngOnInit()` ?

→ Réponse : Lors de l'initialisation du composant

Q41. Rôle de `main.ts` ?

→ Réponse : Démarrer l'application Angular

Q42. Type de directive `*ngFor` ?

→ Réponse : Structurelle

Q43. Nom du système de transformation de données dans les templates ?

→ Réponse : Pipe

Q44. Fonction JS pour convertir "123" en nombre ?

→ Réponse : `Number()` ou `parseInt()`

Q45. Opérateur logique OU ?

→ Réponse : `||`

Q46. Rôle d'`AppComponent` ?

→ Réponse : Composant racine

Q47. Mot-clé pour connaître le type d'une variable ?

→ Réponse : `typeof`

Q48. Signification de `NullInjectorError` ?

→ Réponse : Service non fourni à Angular

Q49. Dossier contenant les composants ?

→ Réponse : `src/app/`

Q50. Syntaxe pour appliquer une classe CSS si condition vraie ?

→ Réponse : `[ngClass]="{'ma-classe': condition}"`

Q51. À quoi sert le décorateur `@Output()` dans un composant Angular ?

→ Réponse : À émettre des événements vers le composant parent

Q52. Quel type d'objet Angular utilise-t-on pour écouter les événements @Output() ?

→ **Réponse :** EventEmitter

Q53. Que permet le two-way binding avec [(ngModel)] ?

→ **Réponse :** Liaison bidirectionnelle entre le template et la classe

Q54. Quel module faut-il importer pour utiliser [(ngModel)] ?

→ **Réponse :** FormsModule

Q55. Que fait la méthode .subscribe() dans Angular ?

→ **Réponse :** Elle permet de réagir aux valeurs émises par un Observable

Q56. Que retourne la méthode http.get() du HttpClient ?

→ **Réponse :** Un Observable

Q57. Quelle méthode utilise-t-on pour envoyer des données à un backend ?

→ **Réponse :** http.post()

Q58. Quel mot-clé TypeScript permet de définir un type facultatif ?

→ **Réponse :** ? (point d'interrogation)

Q59. À quoi sert la méthode .toUpperCase() sur une chaîne ?

→ **Réponse :** À convertir la chaîne en majuscules

Q60. Quelle est la syntaxe correcte pour une interpolation en Angular ?

→ **Réponse :** {{ variable }}

Q61. Dans quel fichier configure-t-on les routes Angular ?

→ **Réponse :** app-routing.module.ts

Q62. Quelle propriété Angular permet de rediriger une route vide ?

→ **Réponse :** redirectTo

Q63. Que fait `[routerLink]='[/home']` ?

→ **Réponse :** Crée un lien vers la route `/home`

Q64. Quelle directive est utilisée pour activer une route dans un `<a>` ?

→ **Réponse :** `routerLinkActive`

Q65. Quel mot-clé TypeScript permet de définir une variable constante ?

→ **Réponse :** `const`

Q66. Que fait la méthode `.filter()` sur un tableau ?

→ **Réponse :** Crée un nouveau tableau avec les éléments qui passent un test

Q67. Que signifie l'opérateur `===` en JavaScript/TypeScript ?

→ **Réponse :** Comparaison stricte (valeur + type)

Q68. Quelle est la valeur initiale d'un tableau vide `.length` ?

→ **Réponse :** 0

Q69. Que fait le pipe `date` dans Angular ?

→ **Réponse :** Formate une date selon un modèle (ex : `shortDate`, `longDate`...)

Q70. Quelle propriété CSS est utilisée pour cacher un élément ?

→ **Réponse :** `display: none;`

Q71. Quel attribut HTML est utilisé pour afficher une image ?

→ **Réponse :** `src`

Q72. Quelle méthode JS convertit un nombre flottant en chaîne avec 2 décimales ?

→ **Réponse :** `.toFixed(2)`

Q73. Que fait `console.log(typeof [])` ?

→ **Réponse :** `"object"`

Q74. Quelle balise Angular permet de définir du contenu alternatif avec `*ngIf` ?

→ **Réponse :** `<ng-template>`

Q75. Quel est le rôle du fichier `environment.ts` ?

→ **Réponse :** Contenir les variables d'environnement (API URL, etc.)

Q76. Quelle méthode JavaScript permet de fusionner deux tableaux ?

→ **Réponse :** `.concat()`

Q77. Quel est le comportement de `[disabled]="true"` sur un bouton ?

→ **Réponse :** Le bouton devient inactif

Q78. Comment commenter une ligne en TypeScript ?

→ **Réponse :** `//` commentaire

Q79. Que retourne une fonction `void` en TypeScript ?

→ **Réponse :** Rien

Q80. Quel opérateur permet d'éviter une erreur si une propriété est `null` ou `undefined` ?

→ **Réponse :** `?.` (opérateur de chaînage optionnel)

Q81. Que permet le décorateur `@Injectable()` ?

→ **Réponse :** Rendre un service injectable via le système de dépendances Angular

Q82. Quel mot-clé est utilisé pour déclarer une fonction en TypeScript ?

→ **Réponse :** `function`

Q83. Que fait `[value]="product.title"` dans un `<input>` ?

→ **Réponse :** Préremplit le champ avec la valeur `title` du produit

Q84. Comment empêcher le comportement par défaut d'un événement JS ?

→ **Réponse :** `event.preventDefault()`

Q85. Quelle est la valeur de `"5" + 2` en JS ?

→ **Réponse :** `"52"`

Q86. Quelle méthode Angular permet de supprimer un élément d'un tableau ?

→ **Réponse :** `.splice()`

Q87. Quel est le rôle du `trackBy` dans `*ngFor` ?

→ **Réponse :** Optimiser le rendu en identifiant chaque élément par une clé

Q88. Que permet l'attribut `[checked]="isSelected"` dans un `<input type="checkbox">` ?

→ **Réponse :** Coche dynamiquement la case selon la valeur de `isSelected`

Q89. Que permet `ng build --prod` ?

→ **Réponse :** Compiler l'application Angular en version optimisée pour la production

Q90. Quelle est la différence entre `let` et `const` en TypeScript ?

→ **Réponse :** `const` ne peut pas être réaffecté après initialisation