

## QCM TYPESCRIPT

**1. Quel est le type de base pour les nombres en TypeScript ?**

- A) number
- B) int
- C) float
- D) double

**2. Comment déclarer une variable de type chaîne de caractères en TypeScript ?**

- A) let name: string = 'John';
- B) var name: str = 'John';
- C) string name = 'John';
- D) let name = 'John';

**3. Quelle est la syntaxe correcte pour définir un tableau de nombres en TypeScript ?**

- A) let list: number[] = [1, 2, 3];
- B) let list: array<number> = [1, 2, 3];
- C) let list = [1, 2, 3];
- D) let list: []number = [1, 2, 3];

**4. Comment définir une fonction retournant un nombre en TypeScript ?**

- A) function add(x: number, y: number): number { return x + y; }
- B) func add(x: number, y: number) -> number { return x + y; }
- C) function add(x, y): number = x + y;
- D) let add = function(x: number, y: number): number { return x + y; }

**5. Quelle est la syntaxe correcte pour définir une énumération en TypeScript ?**

- A) enum Color {Red, Green, Blue}
- B) enumeration Color {Red, Green, Blue}
- C) enum Color = {Red, Green, Blue}
- D) enum Color {Red = 1, Green, Blue}

**6. Comment rendre un paramètre de fonction optionnel en TypeScript ?**

- A) function buildName(firstName: string, lastName?: string) {}
- B) function buildName(firstName: string, lastName: string?) {}
- C) function buildName(firstName: string, optional lastName: string) {}
- D) function buildName(firstName: string, lastName = 'Smith') {}

**7. Quel est le mot-clé utilisé pour créer une classe en TypeScript ?**

- A) class
- B) new
- C) object
- D) create

**8. Comment créer une instance d'une classe en TypeScript ?**

- A) let obj = new MyClass();
- B) let obj = MyClass();
- C) let obj: MyClass = new MyClass();
- D) let obj = create MyClass();

## QCM TYPESCRIPT

**9. Quel modificateur d'accès rend une propriété accessible uniquement à l'intérieur de sa classe ?**

- A) public
- B) private**
- C) protected
- D) internal

**10. Quelle est la syntaxe pour implémenter une interface en TypeScript ?**

- A) class MyClass implements MyInterface {}**
- B) class MyClass extends MyInterface {}
- C) interface MyClass implements MyInterface {}
- D) class MyClass uses MyInterface {}

**11. Comment déclarer une constante en TypeScript ?**

- A) let constant = "value";
- B) const constant = "value";
- C) var constant = "value";
- D) constant = « value";

**12. Quel est le résultat de l'assertion null === undefined en TypeScript ?**

- A) true
- B) false**
- C) null
- D) undefined

**13. Quelle est la différence principale entre let et var en TypeScript ?**

- A) let a une portée de fonction, tandis que var a une portée de bloc.
- B) var a une portée de fonction, tandis que let a une portée de bloc.**
- C) Il n'y a aucune différence.
- D) var est obsolète et let est la nouvelle syntaxe.

**14. Comment annoter une variable avec un type union en TypeScript ?**

- A) let variable: string | number = 'value';**
- B) let variable: string || number = 'value';
- C) let variable: [string | number] = 'value';
- D) let variable: (string | number) = 'value';

**15. Quelle est la syntaxe pour définir une propriété de classe comme étant en lecture seule en TypeScript ?**

- A) readonly propName: string;**
- B) const propName: string;
- C) private propName: string;
- D) public readonly propName: string;

**16. Comment TypeScript gère-t-il le typage dynamique ?**

- A) En utilisant le type any**
- B) En utilisant le type dynamic
- C) En convertissant automatiquement les types
- D) TypeScript ne supporte pas le typage dynamique

## QCM TYPESCRIPT

**17. Quelle est l'utilité du mot-clé extends en TypeScript ?**

- A) Pour étendre une interface
- B) Pour hériter d'une classe
- C) A et B
- D) Pour ajouter des fonctionnalités à des fonctions

**18. Comment déclarer un tuple en TypeScript ?**

- A) `let tuple: [string, number] = ['hello', 10];`
- B) `let tuple: array<string, number> = ['hello', 10];`
- C) `let tuple = ['hello', 10];`
- D) `let tuple: tuple<string, number> = ['hello', 10];`

**19. Quel est le but du mot-clé as en TypeScript ?**

- A) Pour déclarer le type d'une variable
- B) Pour la conversion de type
- C) Pour spécifier une valeur par défaut pour un paramètre
- D) Pour créer un alias de type

**20. Quelle est la différence entre les types unknown et any ?**

- A) unknown est plus strict que any
- B) any est plus strict que unknown
- C) Il n'y a aucune différence
- D) unknown est un nouveau type introduit pour remplacer any