

Partie 1 – Questionnaire (10 points)

1. Qu'est-ce que React, et quel est son objectif principal ?

React est un framework/bibliothèque JavaScript open-source développé par Facebook.

Il permet de créer des interfaces utilisateur interactives et dynamiques à partir de composants réutilisables.

Son objectif principal est de simplifier la création d'applications web modernes grâce à une approche déclarative et à la mise à jour efficace du DOM via le Virtual DOM.

2. Quelle est la différence entre une bibliothèque et un framework ?

- Une bibliothèque (comme React) fournit des outils qu'on appelle soi-même dans son code.
- Un framework (comme Angular ou Vue) impose une structure et un cycle d'exécution — c'est lui qui appelle votre code.

3. Explique le rôle du Virtual DOM dans React.

Le Virtual DOM est une représentation en mémoire du DOM réel.

React y fait les mises à jour en premier, puis compare ("diff") les deux versions de l'arbre pour n'appliquer au DOM réel que les changements nécessaires.

4. Quelle commande permet d'initialiser automatiquement un projet React prêt à l'emploi ?

```
npx create-react-app mon-projet
```

Cette commande configure automatiquement un projet React complet avec toutes les dépendances nécessaires (React, ReactDOM, scripts de build, etc.).

5. Quelle est la différence entre un fichier .ts et .tsx ?

- .ts → fichier TypeScript standard, sans JSX.
- .tsx → fichier TypeScript avec du JSX, donc utilisé pour les composants React.
Le compilateur TypeScript transforme les balises JSX en appels à `React.createElement`.

6. Que fait la fonction suivante :

```
ReactDOM.render(<App />, document.getElementById('root'));
```

Cette instruction monte le composant React `<App />` dans le DOM, à l'intérieur de l'élément HTML ayant l'identifiant `root`.

C'est le point d'entrée de l'application React : tout le rendu de l'app se fera dans cette balise.

7. À quoi sert le hook useState ?

useState est un Hook qui permet d'ajouter un état local à un composant fonctionnel.
Il retourne un tableau à deux éléments : la valeur de l'état et la fonction pour la modifier.

```
const [compteur, setCompteur] = useState(0);
```

Chaque appel à setCompteur entraîne un nouveau rendu du composant avec la valeur mise à jour.

8. Que fait le hook useEffect lorsqu'on fournit un tableau vide [] en second argument ?

useEffect exécute la fonction passée une seule fois après le premier rendu du composant, exactement comme componentDidMount dans les composants de classe.

Il sert souvent à exécuter du code d'initialisation (par exemple, une requête API ou un `document.title = ...`).

9. Quelle différence y a-t-il entre JavaScript et TypeScript dans un projet React ?

JavaScript	TypeScript
Langage dynamique (pas de vérification de types)	Langage à typage statique
Plus simple, pas de compilation	Vérifie les types à la compilation
Plus sujet aux erreurs	Détection d'erreurs avant exécution
Fichiers .js / .jsx	Fichiers .ts / .tsx
TypeScript facilite la maintenance, la documentation et la fiabilité du code.	

10. Que représente le terme SPA (Single Page Application) ?

Une Single Page Application est une application web qui ne recharge pas la page complète à chaque navigation.

Tout le contenu est chargé une seule fois, et les vues changent dynamiquement grâce à JavaScript et React.

Partie 2 – Mise en place du projet React SuperHeros App

Étape 1 – Création du projet

```
npm create vite@latest superheros-app -- --template react-ts
cd superheros-app
npm install
```

Étape 2 – Structure des fichiers

- index.html : page racine contenant `<div id="root"></div>`
- src/main.tsx : point d'entrée React (ReactDOM.createRoot)
- src/App.tsx : composant principal
- src/SuperHeros.json : données de super-héros

Étape 3 – Premier composant (App.tsx)



```
import React, { FunctionComponent } from 'react';
const App: FunctionComponent = () => {
  const name: string = 'React';
  return <h1>Avec React: Hello {name}!</h1>;};export default App;
```

Étape 4 – Ajout de l'état (useState)

```
import React, { useState } from 'react';
const App = () => {
  const [name, setName] = useState('World');
  return (
    <div><h1>Hello {name}!</h1>
    <button onClick={() => setName(name + '?')}>Ajoutez un?</button>
    <button onClick={() => setName('World')}>Remettre</button></div> );};export default App;
```

Étape 5 – Utilisation de useEffect

```
import React, { useState, useEffect } from 'react';
const App = () => {
  const [count, setCount] = useState(0);
  useEffect(() => {document.title = `Compteur: ${count}`;}, [count]);
  return (
    <div><p>Vous avez cliqué {count}fois</p>
    <button onClick={() => setCount(count + 1)}>Cliquez moi</button></div>
  );};export default App;
```

	Technologies Web	
	TD REACT	

Partie 3 – Affichage des SuperHéros

Lien GitHub : <https://github.com/blaironarthur-coder/TD-React.JS.git>