

# Votre "Hello World" React

**Objectif :** Installer l'environnement React et afficher votre premier composant avant demain matin.

## L'Installation

Oubliez les configurations complexes. On utilise **Vite**, c'est l'outil le plus rapide du monde.

1. Ouvrez votre terminal dans votre dossier de projets.

2. Lancez la commande magique :

```
npm create vite@latest mon-premier-react -- --template react-ts
```

*(Cela crée un projet React avec TypeScript déjà configuré).*

3. Entrez dans le dossier et installez les dépendances :

```
cd mon-premier-react
```

```
npm install
```

4. Lancez le serveur de développement :

```
npm run dev
```

Cliquez sur le lien <http://localhost:5173> qui s'affiche. Vous devriez voir le logo React tourner.

## 2. Le Nettoyage

Vite nous donne un projet de démonstration un peu trop rempli. On va tout casser pour repartir d'une page blanche.

1. **Supprimez** tout le contenu du fichier src/App.css.
2. **Supprimez** le fichier src/index.css (ouitez-le).
3. Ouvrez src/App.tsx. C'est votre composant principal.

**Remplacez TOUT le code de src/App.tsx par ceci :**

```
// src/App.tsx
```

```
// Un composant React est juste une fonction qui retourne du "HTML"
function App() {
```

```
// Cette syntaxe (HTML dans du JS), c'est du JSX, plus précisément TSX ici.
return (
```

```
<div style={{ textAlign: 'center', marginTop: '50px' }}>
  <h1>🌐 Hello World !</h1>
  <p>Mon premier composant React fonctionne.</p>
</div>
);
}

export default App;
```

## Personnalisation

Maintenant que ça marche, essayez de modifier le texte.

Regardez votre navigateur : **il se met à jour instantanément sans recharger la page.** C'est la magie du "Hot Reload" de Vite.

### Défi rapide :

1. Créez une variable avant le return :  
const prenom = "VotrePrénom";
2. Utilisez cette variable dans le HTML avec des accolades {} :  
<h1>Hello {prenom} !</h1>