Cycle de vie des composants

Les composants React avancé

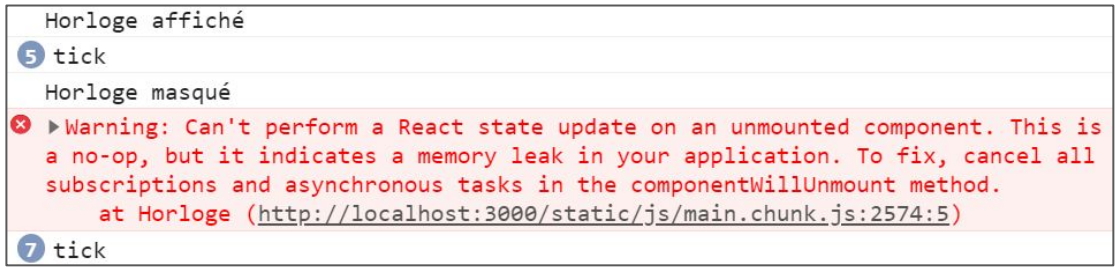
But du cycle de vie d'un composant

C'est un mécanisme qui permet d'exécuter du code lors d'une actualisation du rendu.

Elles permettent, par exemple de :

- Activer / désactiver les abonnements (méthode setInterval, requête Ajax,...)
- Éviter les erreurs provoqués par la modification du state d'un élément inexistant.

Exemple d'erreur lors du déclenchement d'un « setState » d'un composant retiré du DOM.



Le Hook d’effet

Les composants React avancé

Le Hook d’effet

Le Hook d’effet permet au composant React d'exécuter un bloc de code lors de l’ajout, la mise à jour et le retrait d’un composant du DOM.

Il est conseillé de créer un Hook d’effet pour chaque effet de bord du composant.

Il existe deux grands types d’effets de bord dans les composants React :

➔ Ceux qui ne nécessitent pas de traitement après leurs exécutions.

➔ Ceux qui nécessitent d’un nettoyage.

Le Hook d’effet - Effets sans nettoyage

Le Hook d’effet sans nettoyage permet d'exécuter du code uniquement lors de l’ajout et la mise à jour du composant.

Celui-ci peut être utilisé par exemple pour effectuer :

- Une requête réseau
- Des modifications du DOM

```
import React, { useState, useEffect } from 'react';

const DemoEventHook = function() {
  const [count, setCount] = useState(0);

  useEffect(() => {
    document.title = `Vous avez cliqué ${count} fois`;
  });

  return (
    <div>
      <button onClick={() => setCount(c => c + 1)}>
        Compteur : {count}
      </button>
    </div>
  );
}

export default DemoEventHook;
```

Le Hook d'effet - Effets avec nettoyage

Le Hook d'effet avec nettoyage permet d'ajouter du code à exécuter lors du retrait du composant.

Pour cela, il faut renseigner la fonction de nettoyage dans le retour du hook.

Le nettoyage permet de :

- D'annuler une/des requête(s)
- Retirer un abonnement

```
import React, { useEffect } from 'react';

const DemoEventHookCleanup = function() {

    useEffect(() => {
        console.log("Lancement de l'effet");

        return () => {
            console.log("Nettoyage de l'effet");
        }
    });

    return (
        <div>Demo d'un effet avec nettoyage</div>
    );
}

export default DemoEventHookCleanup;
```

Le Hook d'effet - Optimiser les performances

Il est possible d'indiquer à React de sauter l'exécution d'un effet.

Pour cela, il faut donner une liste en deuxième argument du Hook d'effet.
L'effet sera déclenché, uniquement si au moins une des valeurs du tableau a changé.

```
useEffect(() => {
  document.title = `Vous avez cliqué ${count} fois`;
}, [count]);
```

Ce mécanisme permet également de déclencher un effet uniquement lors du premier rendu. Pour cela, il suffit de lui donner un tableau vide « [ ] » comme argument.

Exercice • Cycle de vie des composants

Les composants React avancé

Exercice

- Créer un composant « Horloge »
  - Une balise “p” affiche le temps du chrono au format : **« hh : mm : ss »**.
  - L’horloge s'actualise automatiquement toutes les 200 millisecondes.

- Créer un composant « DateDuJour »
  - Une balise “p” affiche la date du jour en francais au format : **« dd mois yyyy »**.

- Ajouter les 2 composants dans l’application
  - Permettre d’alterner entre l’affichage des 2 composants, à l’aide d’un bouton.