

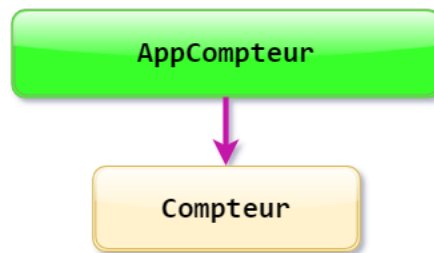
Module : M202 – Développement front-end
Filière : Développement digital – option web full stack

TP 5

Objectifs :

- Cycle de vie d'un composant

Exercice 1



Le composant **Compteur** contient un compteur qu'on peut l'incrémenter et décrémenter
Le composant **AppCompteur** contient deux boutons pour charger et décharger le composant Compteur

Le composant **AppCompteur** est utilisé seulement pour vérifier la méthode *componentWillUnmount*

Monter Démonter

composant Compteur

compteur:0

Incrementer Decrementer

Rendu console au premier chargement

```

constructeur
Render
Component Did mount
-----
  
```

Rendu console au premier chargement

```

Render
Component Did update
-----
  
```

Quand je click sur le bouton Démonter :

```

Component est démonté
  
```

Exercice 2 :

Question 1 :

Après avoir exécuté ce code, combien de console.log vous attendriez-vous à voir, et quand les verriez-vous ?

```
import React, { useEffect } from "react";
import ReactDOM from "react-dom";
const App = () => {
  useEffect(() => {
    console.log("TEST!");
  }, []);
  return <div>test composant</div>;
};
ReactDOM.render(<App />, document.querySelector("#root"));
```

- a- Je vais voir un log de 'TEST', il est affiché après le composant est rendu
- b- Je ne vais voir aucun log
- c- Je vais voir un log de 'TEST', il est affiché avant le composant est rendu
- d- Je vais voir deux log de 'TEST', ils sont affichés après le composant est rendu

Question 2 :

On suppose qu'on a exécuté le code suivant et qu'un utilisateur a cliqué trois fois sur l'élément bouton. Combien d'instructions de log 'TEST' vous attendriez-vous à voir imprimées ?

```
import React, { useEffect, useState } from "react";
import ReactDOM from "react-dom";
const App = () => {
  const [count, setCount] = useState(0);
  useEffect(() => {
    console.log("TEST!");
  }, []);
  const onClick = () => {
    setCount(count + 1);
  };
  return (
    <div>
      <h1>Count: {count}</h1> <button onClick={onClick}>Click me!</button>{" "}
    </div>
  );
};
ReactDOM.render(<App />, document.querySelector("#root"));
```

- a- Je vais voir trois console.log 'TEST'
- b- Je vais voir quatre console.log 'TEST'
- c- Je vais voir un console.log 'TEST'

Question 3 :

Ce code est identique à la dernière question, sauf que maintenant useEffect a un tableau avec un seul argument à l'intérieur.

Combien de fois vous attendez-vous à voir l'instruction de console.log 'TEST' après qu'un utilisateur a cliqué trois fois sur le bouton ?

```
import React, { useEffect, useState } from "react";
import ReactDOM from "react-dom";
const App = () => {
  const [count, setCount] = useState(0);
  useEffect(() => {
    console.log("TEST!");
  }, [count]);
  const onClick = () => {
    setCount(count + 1);
  };
  return (
    <div>
      <h1>Count: {count}</h1> <button onClick={onClick}>Click me!</button>
    </div>
  );
};
ReactDOM.render(<App />, document.querySelector("#root"));
```

- a- Je vais voir un console.log 'TEST'
- b- Je vais voir deux console.log 'TEST'
- c- Je vais voir trois console.log 'TEST'
- d- Je vais voir quatre console.log 'TEST'

Exercice 3 : (composant fonctionnel)



nom: prénom: age: 0

Salutation

salut --- --- vote age est:0

On souhaite que le titre de la page prenne les valeurs nom et le prénom

1. Au démarrage.
2. A chaque modification des propriétés d'états nom et prénom
3. A chaque modification du nom ou prénom ou âge