

Essential React Hooks Explanation (Darija)

1. useState

Had hook kayst3mlou bash nkhznou chi qima w n-3amlo biha f-component, u k-t3awwna bzzaf bach nbdlou les valeurs m3a kol interaction dyal user. B7al chi variable bghina y-tbdl m3a wa9t.

Kifash kaykhdm:

...

```
import { useState } from 'react';
```

```
function Counter() {
```

```
  const [count, setCount] = useState(0);
```

```
  return (
```

```
    <div>
```

```
      <p>Le 3ad l-7ali: {count}</p>
```

```
      <button onClick={() => setCount(count + 1)}>Zid wahd</button>
```

```
    </div>
```

```
  );
```

```
}
```

...

2. useEffect

Had hook kanst3mlou bash ndir chi haja mn ba3d ma ych7al component, b7al talb dyal API ola

nbddl chi 7aja kat3tamid 3la update m3in.

Kifash kaykhdm:

...

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

```
function DataFetcher() {
```

```
  const [data, setData] = useState(null);
```

```
  useEffect(() => {
```

```
    fetch('/api/data')
```

```
      .then((response) => response.json())
```

```
      .then((data) => setData(data));
```

```
  }, []);
```

```
  return <div>{data ? JSON.stringify(data) : 'Katsber chwya...'}</div>;
```

```
}
```

...

3. useContext

Had hook kay3awn bash n-sharkiwi l-qima bin les components bla ma nbdlo **props** f-kol marra. B7al `theme` ola l-lugha.

Kifash kaykhdm:

...

```
import { useContext, createContext } from 'react';
```

```
const ThemeContext = createContext('light');
```

```
function DisplayTheme() {  
  const theme = useContext(ThemeContext);  
  return <p>Theme l-7ali: {theme}</p>;  
}
```

```
function App() {  
  return (  
    <ThemeContext.Provider value="dark">  
      <DisplayTheme />  
    </ThemeContext.Provider>  
  );  
}  
...
```

4. useRef

Had hook kanst3mlou bash ndirou reference l-chi 7aja f-component, mthalan chi DOM element, u kay3awn bzzaf bash ndirou control f-7wayj bla ma y3awd ych7al component.

Kifash kaykhdm:

...

```
import { useRef } from 'react';
```

```

function FocusInput() {
  const inputRef = useRef(null);

  const handleFocus = () => {
    inputRef.current.focus();
  };

  return (
    <div>
      <input ref={inputRef} type="text" />
      <button onClick={handleFocus}>Rkez 3la input</button>
    </div>
  );
}
...

```

5. useReducer

Had hook b7al `useState` walakin kaynfe3 f-7alat ktr ta9ad, b7al ila kayn l-logique m3qd f-state. Kay3tina control aktr.

Kifash kaykhdm:

...

```
import { useReducer } from 'react';
```

```
const initialState = { count: 0 };
```

```
function reducer(state, action) {  
  switch (action.type) {  
    case 'increment':  
      return { count: state.count + 1 };  
    case 'decrement':  
      return { count: state.count - 1 };  
    default:  
      throw new Error();  
  }  
}
```

```
function Counter() {  
  const [state, dispatch] = useReducer(reducer, initialState);  
  
  return (  
    <div>  
      <p>Le 3ad l-7ali: {state.count}</p>  
      <button onClick={() => dispatch({ type: 'increment' })}>Zid wahd</button>  
      <button onClick={() => dispatch({ type: 'decrement' })}>Na9s wahd</button>  
    </div>  
  );  
}
```

6. useMemo

Had hook kay3awn bach n7fdo 3la natija dyal chi 7isab bach ma n3awdch ndir calcul ila ma tbddlch l-qima li kat-athr 3liha. Kay3awn bzzaf bach n7snou l-performance.

Kifash kaykhdm:

...

```
import { useMemo } from 'react';
```

```
function ExpensiveCalculation({ num }) {
```

```
  const result = useMemo(() => {
```

```
    return heavyCalculation(num);
```

```
  }, [num]);
```

```
  return <div>Natija: {result}</div>;
```

```
}
```

...