



# Spread operator

How to Use in different  
Context

---

Shubham Goyal





## Definition

---

The **spread operator (...)** in React (and JavaScript) is used to **spread out elements of an iterable (like an array or object)**. It is commonly used for working with props, state updates, and component rendering in React.



# Spread Operator with Arrays

---



```
const arr1 = [1, 2, 3];  
const arr2 = [...arr1, 4, 5]; // Combines arr1  
and additional values  
console.log(arr2); // Output: [1, 2, 3, 4, 5]
```



# Spread Operator with Objects



---

```
const obj1 = { a: 1, b: 2 };  
const obj2 = { ...obj1, c: 3 }; // Adds a new  
key-value pair  
console.log(obj2); // Output: { a: 1, b: 2, c: 3 }
```



# Passing Props in React

---



The spread operator is widely used to pass all properties of an object as props to a component.

```
const user = { name: 'John', age: 25 };
```

```
function UserProfile(props) {  
  return <h1>{props.name} is {props.age} years old.</h1>;  
}
```

```
<UserProfile name="John" age={25} />
```



# Updating State with Spread Operator

---

## Example: Adding to an Array in State

```
const [items, setItems] = React.useState([1, 2, 3]);

const addItem = () => {
  setItems([...items, 4]);
};
```

## Example: Updating an Object in State

```
const [user, setUser] = React.useState({ name: 'John', age: 25 });

const updateUser = () => {
  setUser({ ...user, age: 26 });
};
```

When updating state in React, we often use the spread operator to ensure immutability (create a new object or array rather than modifying the original one).



# Combining Multiple Objects



You can merge multiple objects, which is useful when combining props or configurations.

```
const defaultSettings = { theme: 'light', layout: 'grid' };  
const userSettings = { layout: 'list', showSidebar: true };  
  
const settings = { ...defaultSettings, ...userSettings };
```