

React: Controlled Components

Phase 2 | Week 1, Lesson 3

Sakib Rasul | Updated January 31, 2024 | Created August 2, 2023



First, a question!

Today's Objectives

Today, we'll answer the following questions:

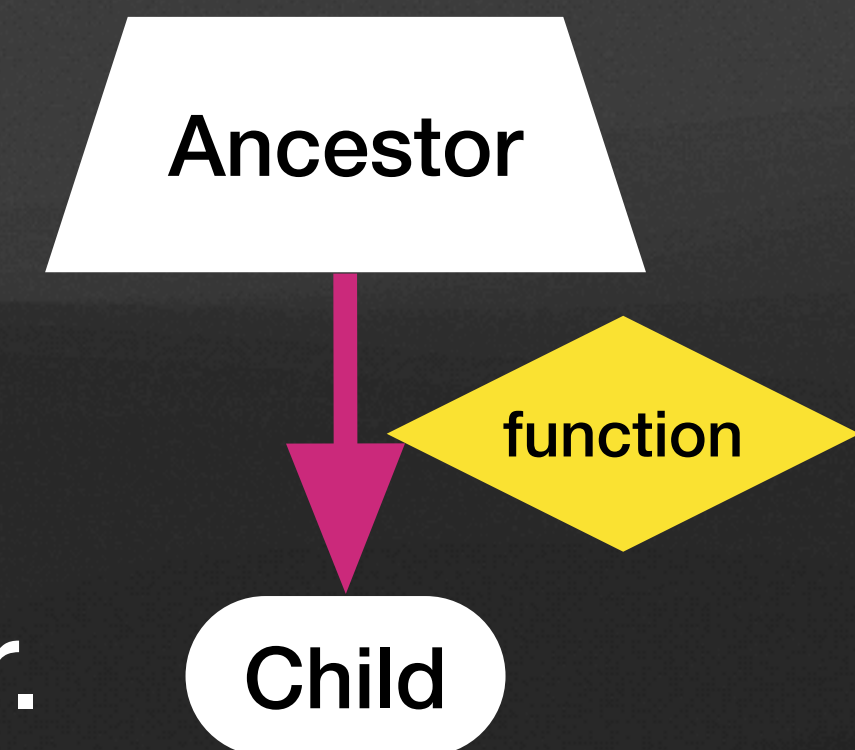
1. How can a child component ask its parent to execute some code?
 - A. When should we grant that kind of **control** to parents?

```
function Child ({ function }) { function(input) }
```


Passing Functions as Props

```
function Child ({ function }) { function(input) }
```

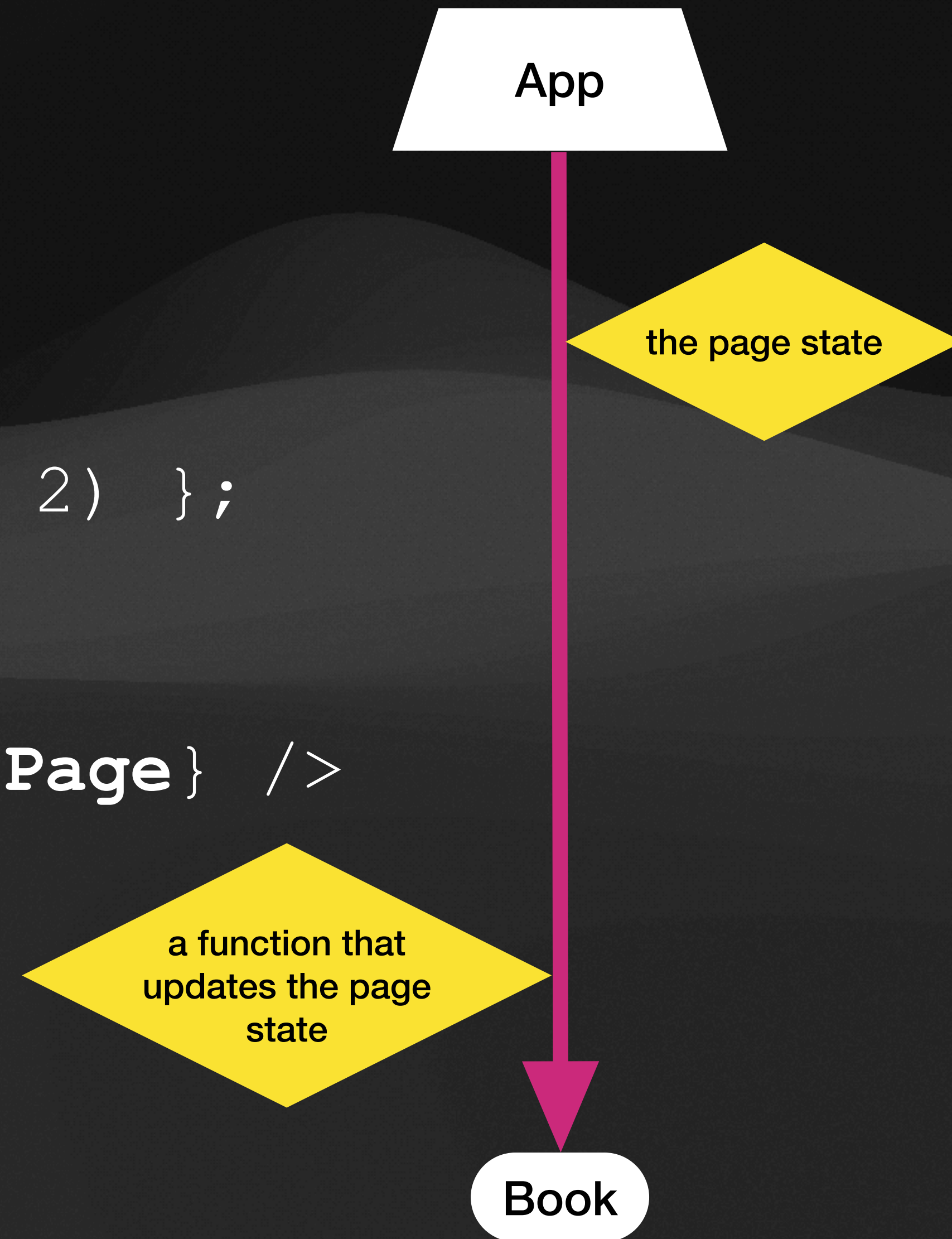
- We can pass any values as props, including objects, arrays, and functions.
- If a parent defines a function and passes it to its child as a prop, we say that:
 - The parent **controls** its child.
 - The child is a **controlled component**.
 - Some of the child's behavior is dictated by an ancestor.
 - The child is driven by props, as opposed to being driven by state*.
- In practice, no component is completely controlled — but it's a useful term!



*When we say “driven by state”, we mean driven by things internal to a component, including information not managed with `useState()`.

Example: A Controlled Book

```
function App() {  
  const [page, setPage] = useState(1);  
  function turnThePage() { setPage(page + 2) };  
  return (  
    <Book page={page} turnThePage={ turnThePage } />  
  );  
};
```



Let's try it



Thanks!

Let me know if you have any questions.