

Another aspect that I mentioned in the state management lecture was derived. State sounds complicated, but it's actually pretty straightforward. So essentially derived state is simply state that is computed from another existing piece of state or also from props. But let's look at some actual code. So here we have three pieces of state, as we can see by the three use state function calls. However, if we analyze these states, it actually doesn't make much sense that all of them exist because num items and total price depend entirely on the card. So num items is simply the number of items in the cart and total price is the sum of all the prices in the cart. And so all the data for these two pieces of state is actually already in the cart. So there is no need to create these additional state variables and doing so is actually quite problematic first because now we have to keep all these states in sync. So we need to be careful to always update them together. So in this situation, whenever we update the cart, we would also need to manually update the number of items and the total price. Otherwise, our states would get out of sync. But updating these three states separately creates a second problem because that will then rerender the component three times, which is absolutely unnecessary in this example. Instead, we can simply derive the Num items and total price state from the cart and therefore solve all these problems because the cart already contains all the data that we need. So here we simply calculate num items as the cart length and total price as the sum of all prices and store them in regular variables. There is no use state required here which will cause no unnecessary rerenders. The Cart state acts as a single source of truth for these related pieces of state, making sure that everything will always stay in sync. And this works because updating the cart will rerender the component, which means that the function is called again. And so then as all the code is executed again, num items and total price will also automatically get recalculated. Now, of course, most of the time we cannot derive state, but whenever you have a situation like this one where one state can easily be computed from another, always prefer derived state. So don't create two state variables if you actually only need one. That's a very common beginner mistake, but now you will be able to avoid it.