




ReactTube

 <https://github.com/learn-co-curriculum/react-hooks-state-events-pairing>  <https://github.com/learn-co-curriculum/react-hooks-state-events-pairing/issues/new>


Learning Goals

- Train yourself to [Think in React](https://reactjs.org/docs/thinking-in-react.html)  <https://reactjs.org/docs/thinking-in-react.html>
- Design components based on a mockup
- Pass down data to components via props
- Handle user interaction via events
- Use **state** to trigger changes in the DOM

Setup

Fork and clone this lab. Then, `cd` into the lab directory and run:

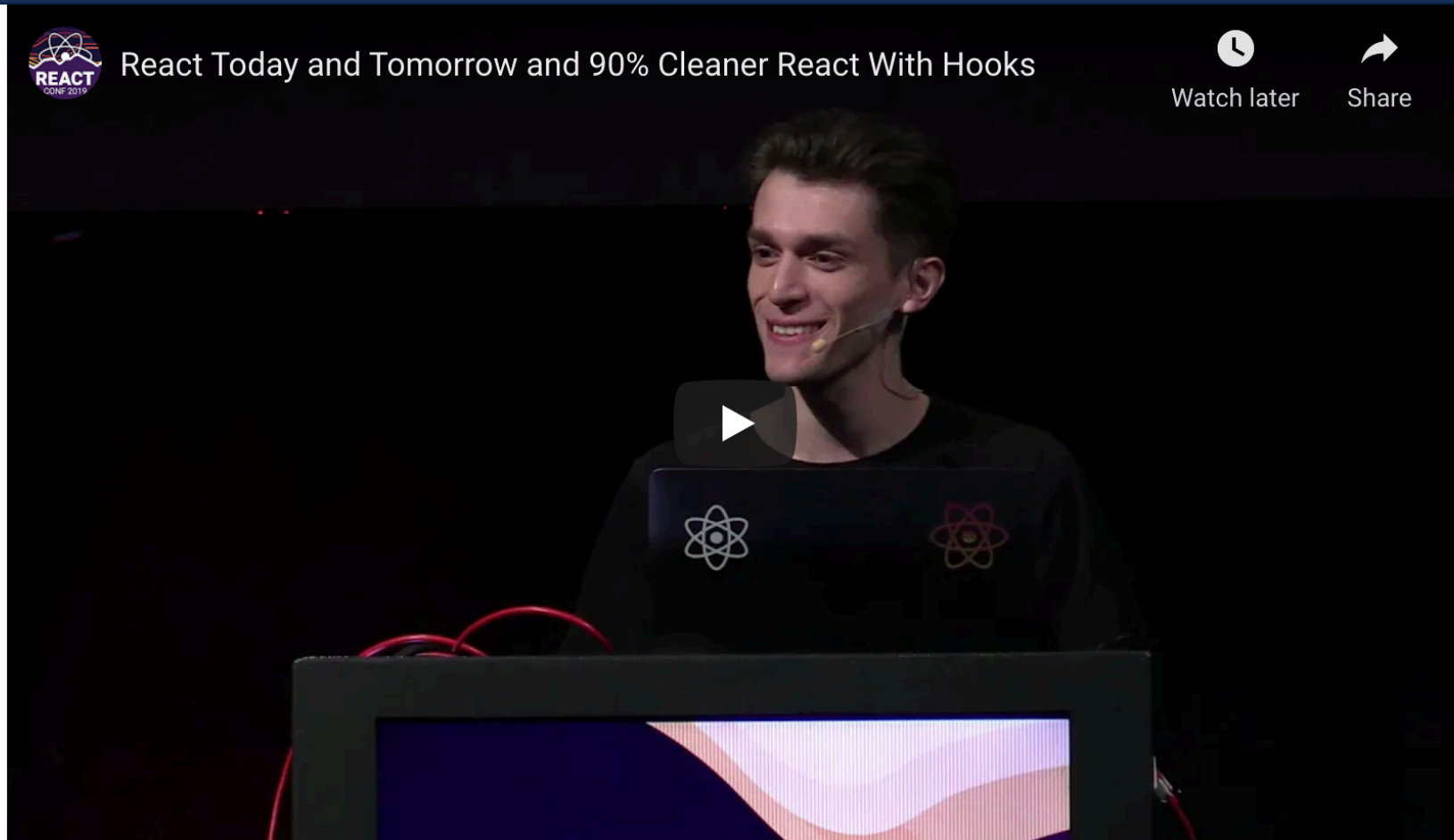
```
$ npm install  
$ npm start
```

This will install all the dependencies and run the application in your browser at <http://localhost:3000>  <http://localhost:3000>.

There are no tests for this lab, so you'll need to check your work by running your application in the browser and trying to match the demo below.

Deliverables

Your goal is to build this app from scratch:



React Today and Tomorrow and 90% Cleaner React With Hooks

730707 Views | Uploaded Oct 26, 2018

9210



185



Hide Comments

2 Comments

duanebot

first!

gaeron

What a great tutorial!

The application has the following features:

- When the app loads, display a video along with its details and a list of all the comments
- When a user clicks on the "👍" button, the number of upvotes for the video should increase
- When a user clicks on the "👎" button, the number of downvotes for the video should increase
- When the user clicks the "Hide Comments" button, the comments should be removed from the page, and the button text should change to "Show Comments". When the button is clicked again, it should toggle back to showing the comments.

The data you'll need for the app can be found in the `data/video.js` file.

Start by planning out what components you'll need and draw out your component hierarchy. Then, build out your components and pass down any data they need via props using the `video` object that's been imported in the `App` component.

Consider what components need to use **state** as you're building out these features.

There's an example of how to embed a video using an `<iframe>` in the `App` component. Other than that, you're on your own in terms of how to design each part of the application. Don't worry too much about getting the styling to match the demo. Focus on functionality.

Good luck!

Bonus

If you finish early and want to get more practice with state and events, here are some ideas for additional features:

- Add upvote/downvote buttons to each comment

- Add a search bar to search comments by username
- Add a "Remove Comment" button to delete comments from the page
- Add a sorting feature to sort the comments