React – Debugging with React-devtools

In this lab, you will debug a react component using the react-dev-tools embedded inside the Chrome browser.

# Objectives

In this lab, you will

* Install the react-devtools extension to Chrome
* Debug a Component from an existing project

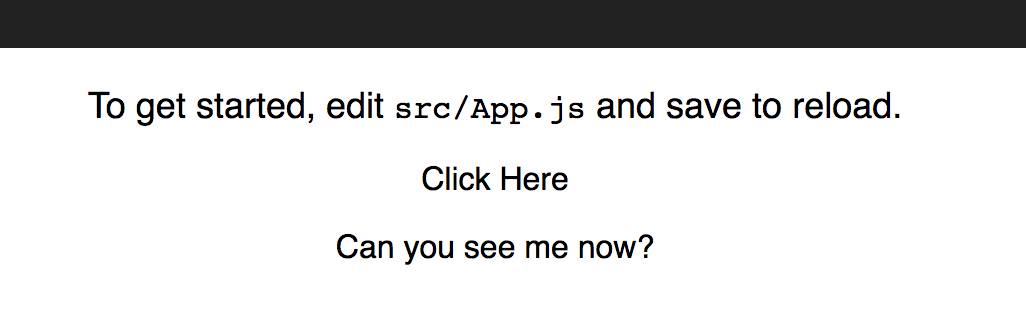
# Examine the Project

1. Open the solution/debugging folder and start the application with:

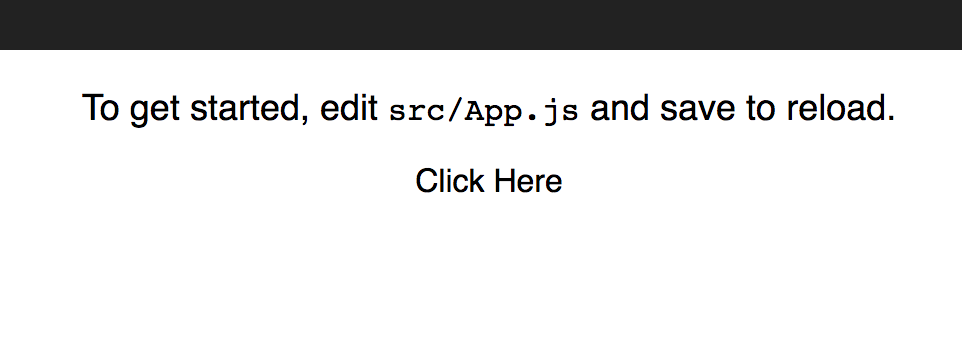
yarn install

yarn start

1. Open the browser to <http://localhost:3000/> and see the below:



1. Click where indicated and notice the line below it disappears!



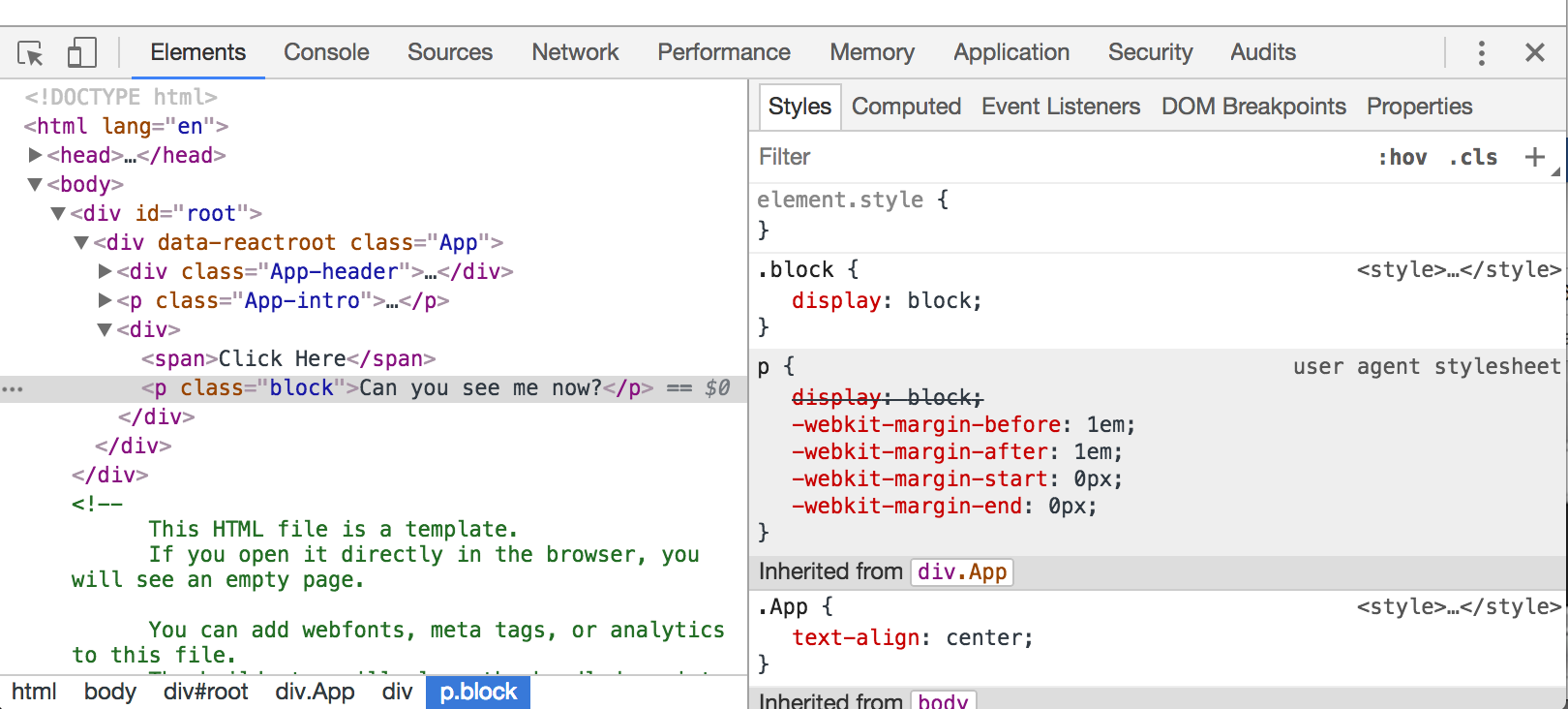
1. The <Hidden> component displays the messages and handles the click as shown below.



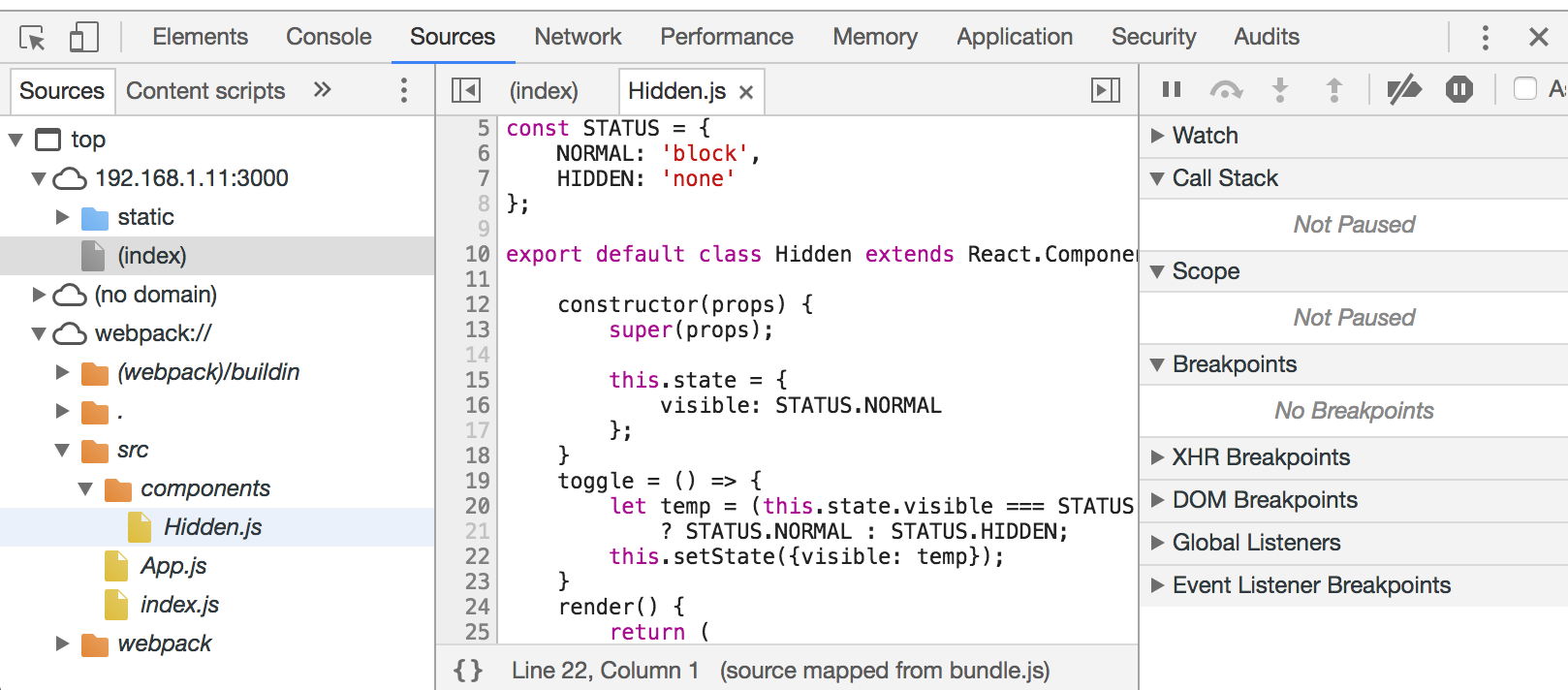
1. Let’s put a breakpoint at line 21 using the Chrome Developer Tools.

# Start the Chrome dev tools

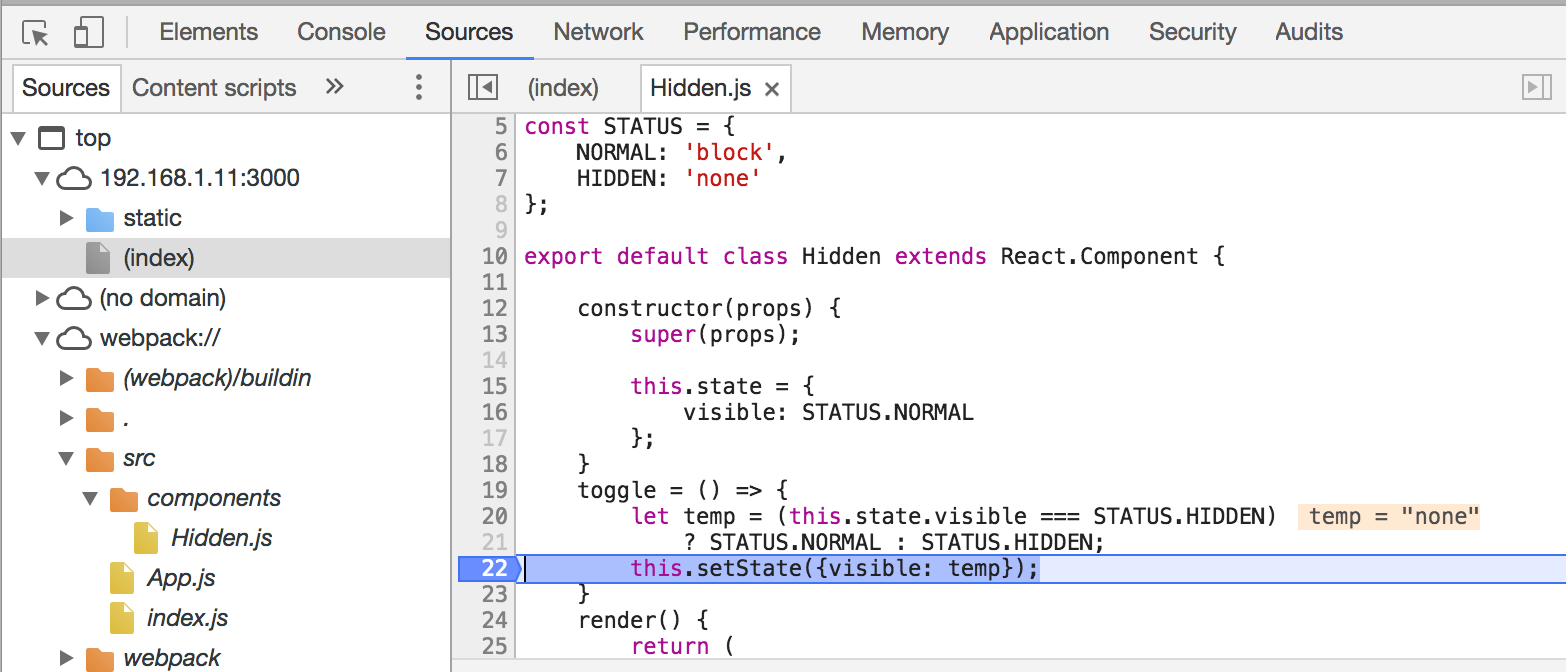
1. In the browser, right click and select ‘inspect’. You should see the developer tools console as shown below:



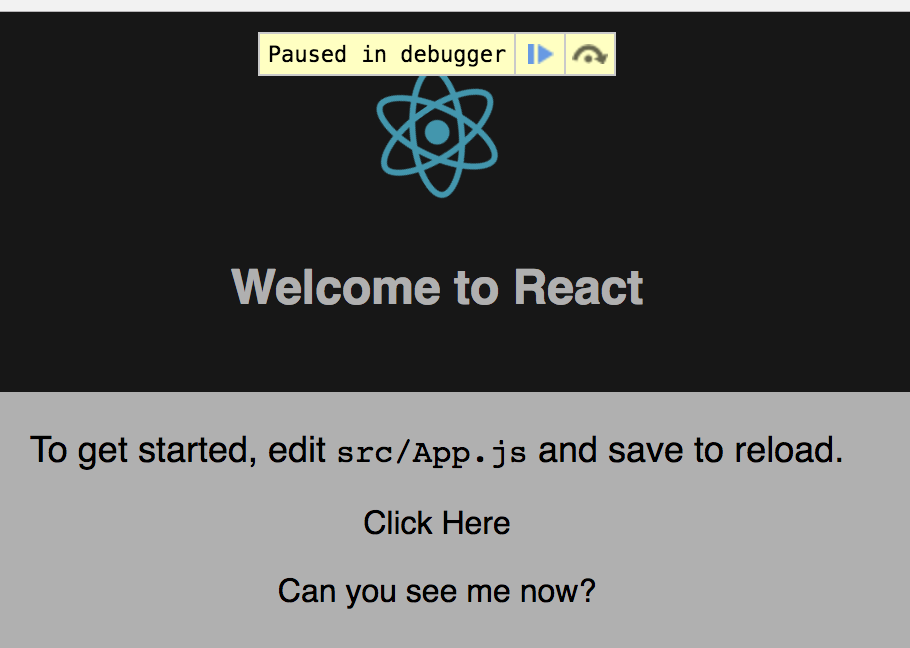
1. This is used by all Chrome applications and allows you to debug JavaScript applications.
2. Let’s look at the ‘Sources” tab



1. Under the ‘webpack://’ folder, you can see all the source code generated in our app. Because we created the app using ‘create-react-app’, webpack generates the bundle with links to the source code.
2. Click on line number 21 above to establish a breakpoint.
3. In the browser, click on the ‘Click Here’ in the app.
4. This stops the app at the breakpoint shown below:



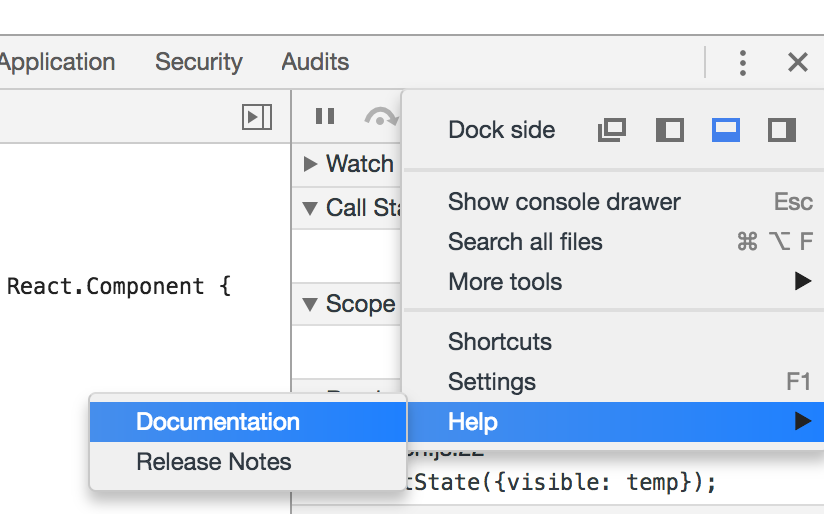
1. Notice the value of temp is also displayed.
2. In addition, there is an indicator in the browser window that shows the debugger has stopped the app.



1. Enable the browser to continue by clicking the blue arrow above.
2. Click the ‘Click Here’ again and watch it pause again.

# Getting Help

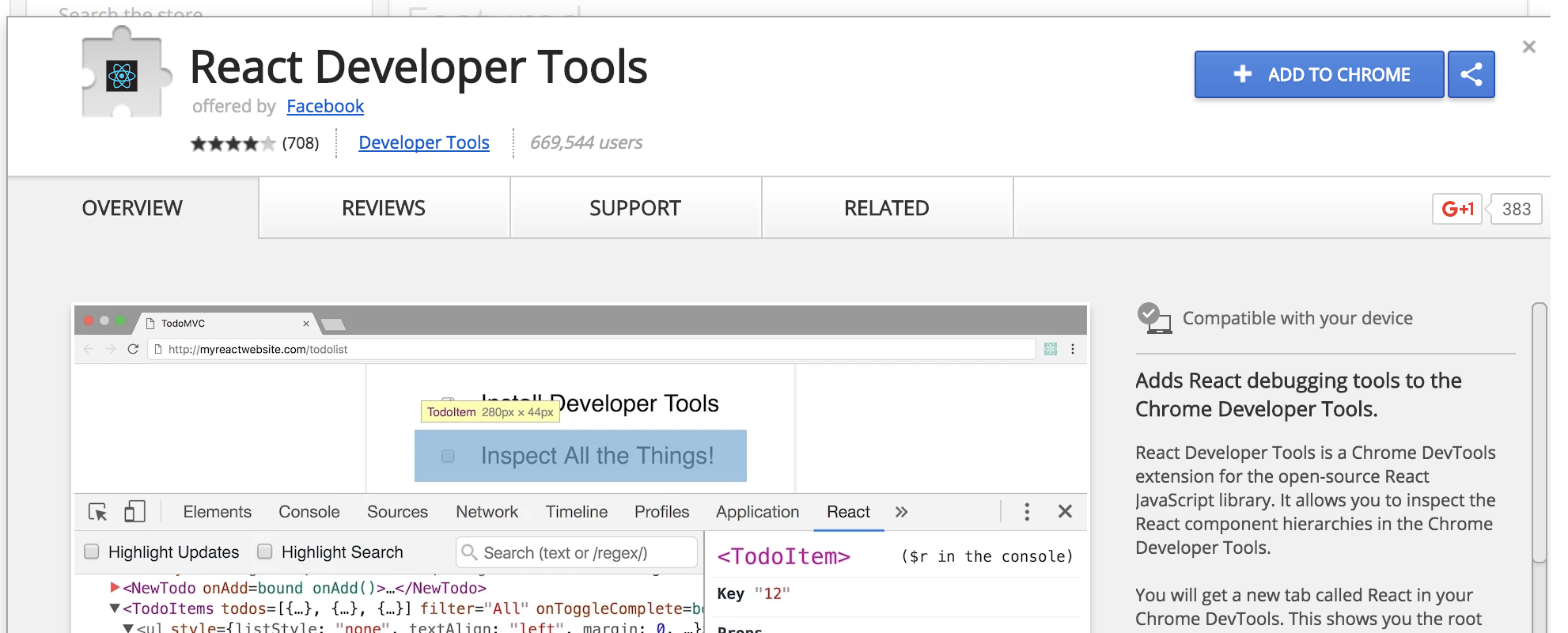
1. Click on the vertical dots at the right hand side of the tab line and select Help -> Documentation



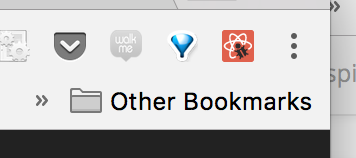
1. This directs you to the Chrome help page for the dev tools. The documentation is good and allows you to do many measurements and examine many parts of the JavaScript engine. Feel free to examine this at your leisure.

# React-devtools

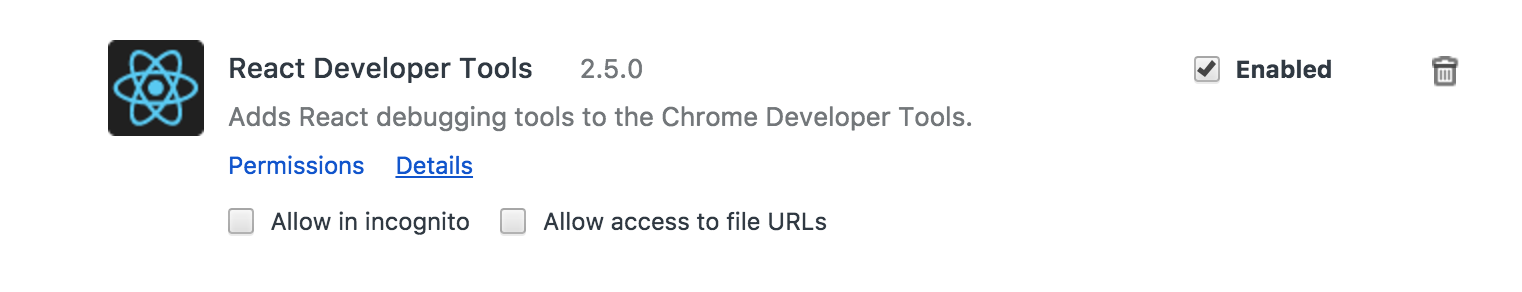
1. The react-devtools is an extension to Chrome that allows us to analyze the React components in the browser. It shows us the props and the state and allows us to modify them when debugging.
2. Install the extension from <https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi?hl=en>



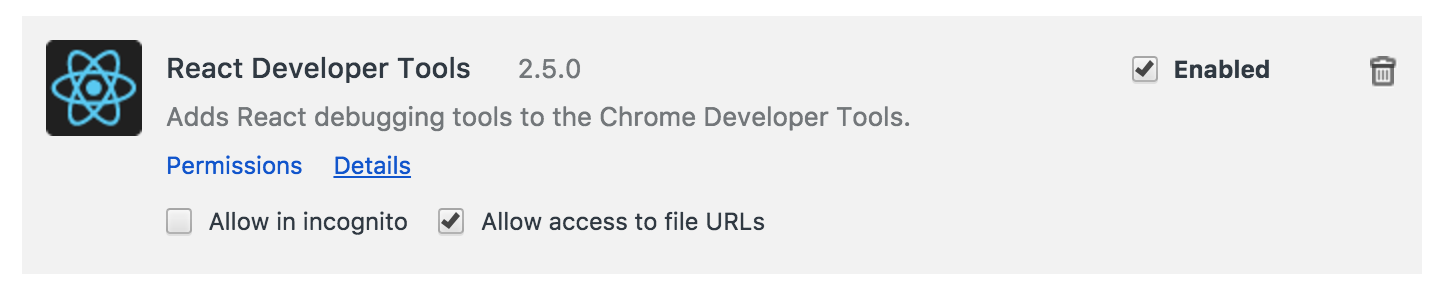
1. Click the Add to Chrome button. Tell it to add the extension.
2. It will generate a new icon on the address bar. This changes color when the app is a React app. Note, this one is red.



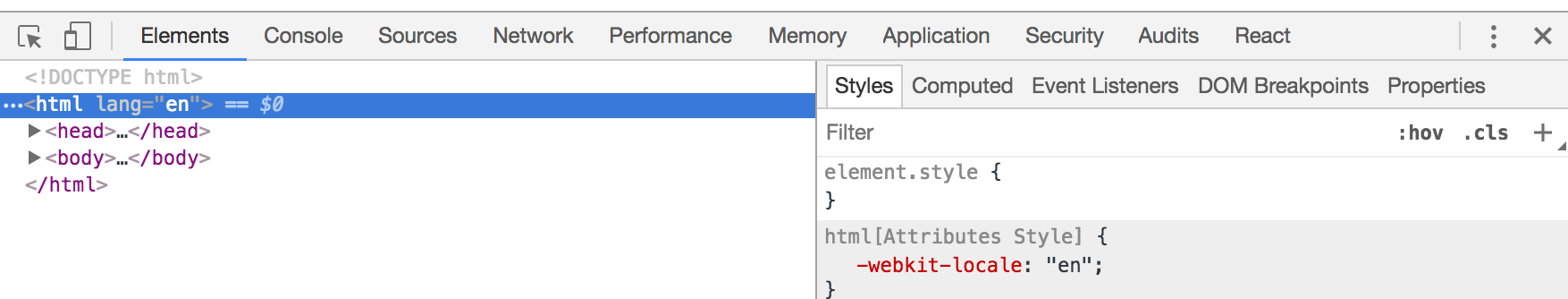
1. Let’s configure the extension. Click on the three vertical dots above and select More Tools 🡪 Extensions
2. Find the React Developer Tools shown below:



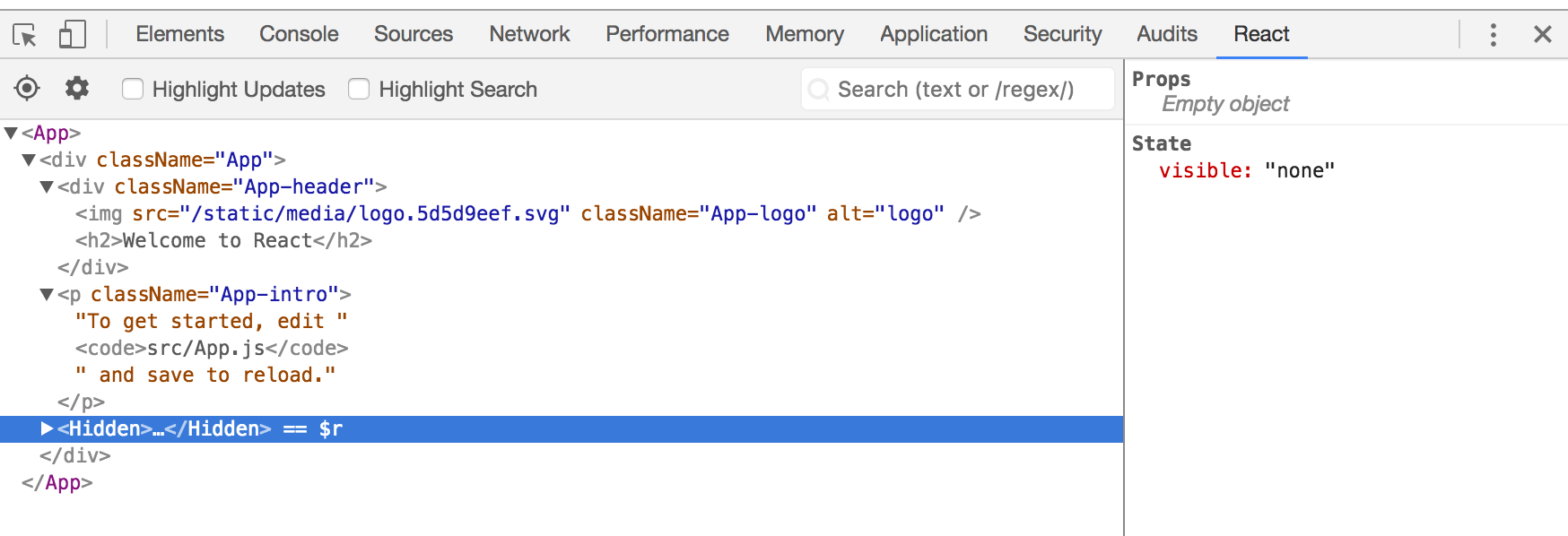
1. Click the ‘Allow access to File URLs’ above to get:



1. Close the debug console and reopen it. There should now be an additional tab in the nav bar.



1. Click on the ‘React’ tab and select the <Hidden> component as shown:



1. Notice the state data in the right pane. Click on the value of the state variable, visible, and change it to the string, ‘hello”. Is the string visible?
2. Change the value back to ‘none’. Is the string visible?
3. This makes it much easier to monitor the changes to the state and props as the application runs.
4. Have fun!

Congratulations. You have completed this lab.