Content

[1. Debug reactjs component life cycle 1](#_Toc5566)

[2. Transform html to Reactjs 3](#_Toc14201)

[1.3 Define Components 4](#_Toc7995)

[1.4 replace index.html 5](#_Toc26763)

[1.5 add router 5](#_Toc5253)

[1.6 add link 6](#_Toc7107)

[2 Add Nodejs login to React 6](#_Toc15284)

[2.1 Alter Form 6](#_Toc30030)

[2.2 Add events Handler 7](#_Toc3306)

[2.3 Add bind 7](#_Toc1734)

[2.4 handle form post request 8](#_Toc27987)

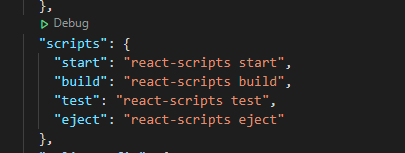
[2.5 add route context 8](#_Toc15035)

[2.6 add auth 9](#_Toc5896)

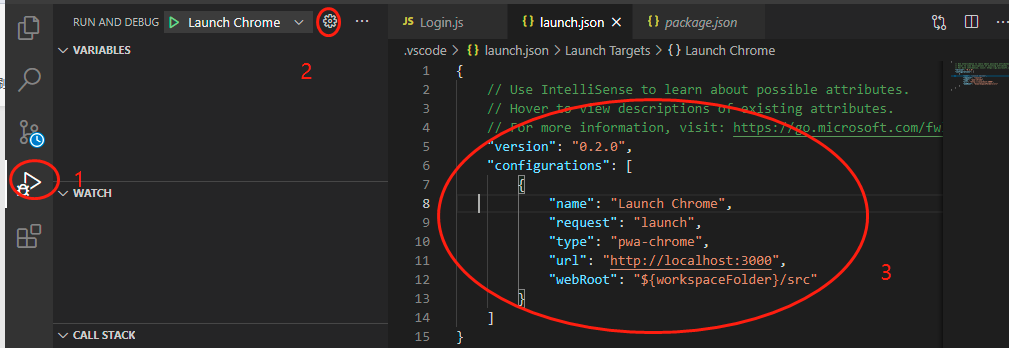
# Debug reactjs component life cycle

Download this repo: <https://github.com/onebit256/react-demo>

Step 1: add scripts to package.json

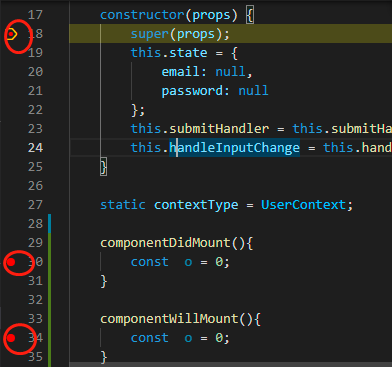


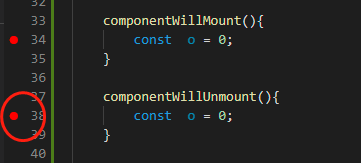
Step2: config launch.json



Step3: npm start

Step 4: put break points on source/component/login







Step 5：click 2 launch in step 2

Step 6 go to browser and type 127.0.0.1:3000/login to trigger the debugger

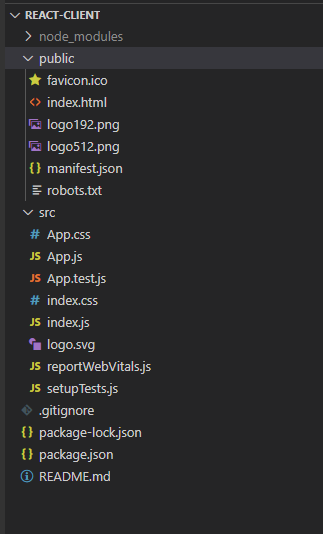
You will be able to see the component life cycle

# Transform html to Reactjs

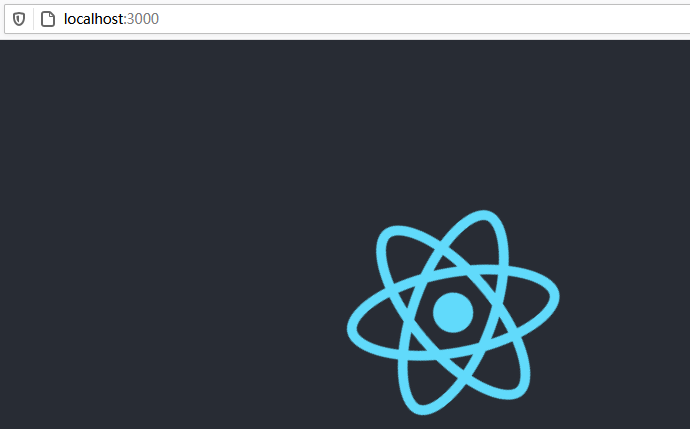
Step1: init project

npx create-react-app .

This will auto generates the folder structure

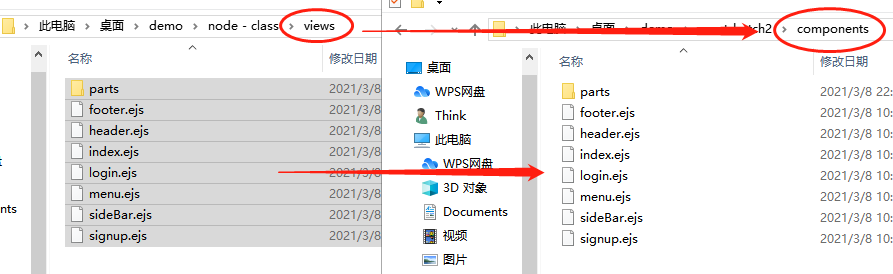


Npm start to test reactjs, you will see

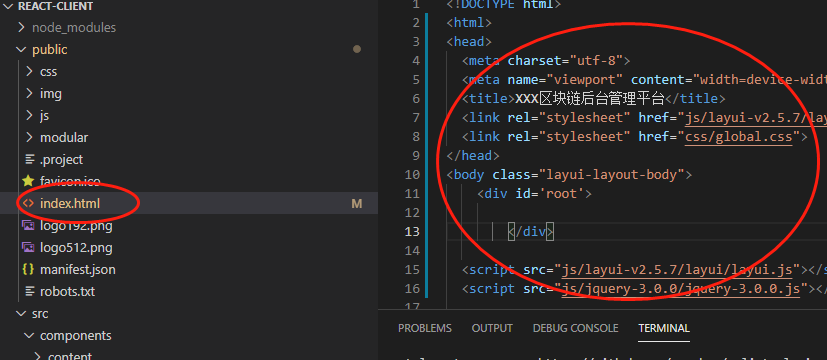


Step2 convert all ejs template to class components

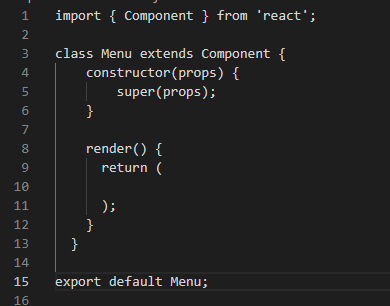
1. Creating empty folder called components
2. Copy all the ejs from lab5 to components folder



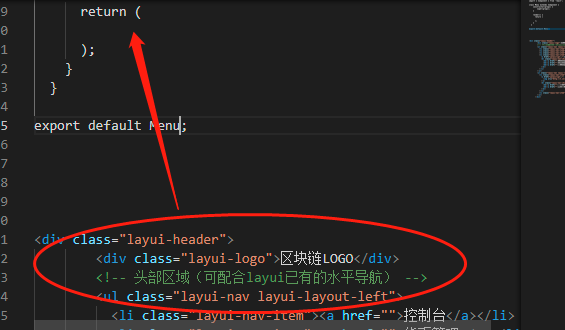
1. copy header.ejs and footer.ejs to index.html, write it as follow



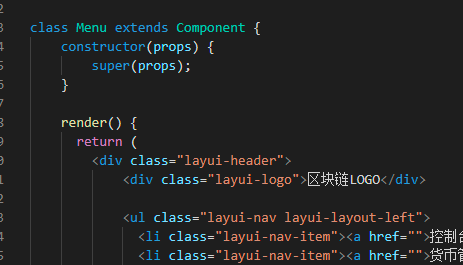
1. Convert ejs to react component: copy and past this template to each file and change the file extension to js



1. Convert ejs to react component: move the original html to return



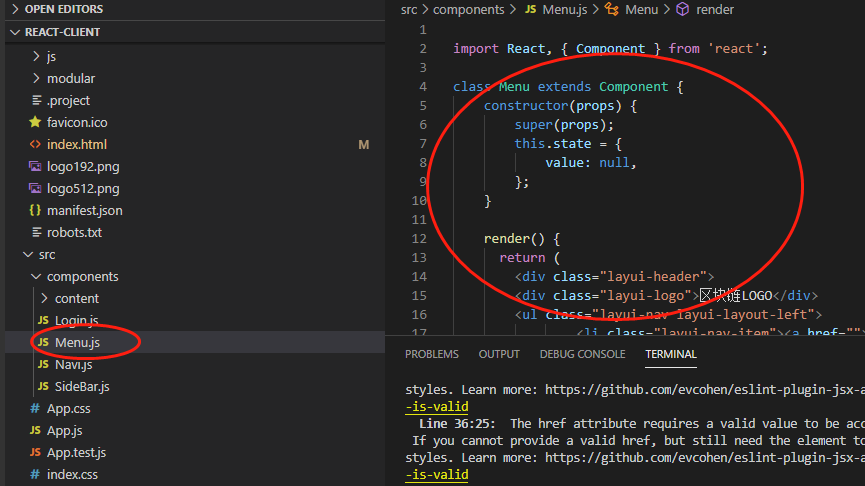
And it becomes



Do the same thing for sideBar Login and signup

For index.ejs move it to app.js

## 1.3 Define Components



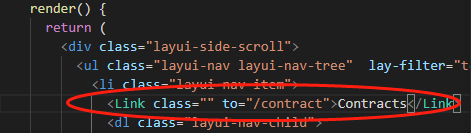
## 1.4 replace index.html

## 1.5 add router

Use exact keyword to do switch



## 1.6 add link

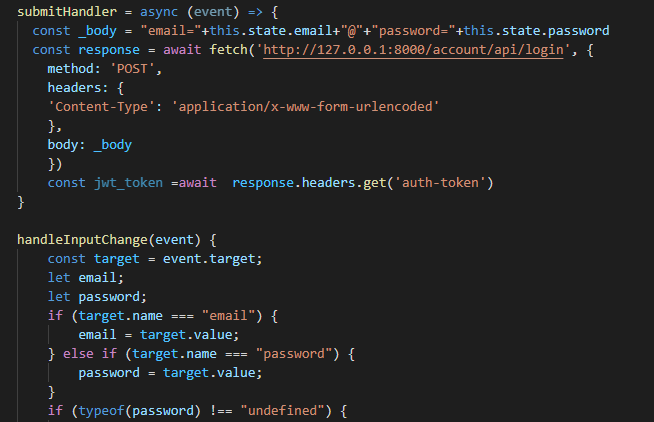


# 2 Add Nodejs login to React

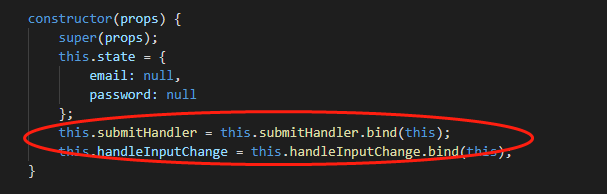
## 2.1 Alter Form



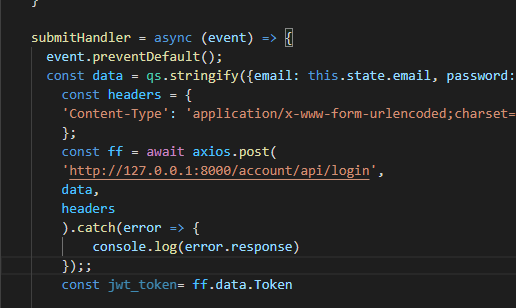
## 2.2 Add events Handler



## 2.3 Add bind



## 2.4 handle form post request

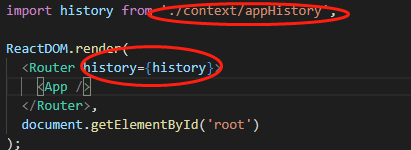


## 2.5 add route context

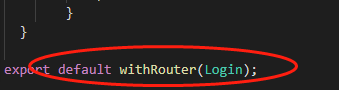
Create

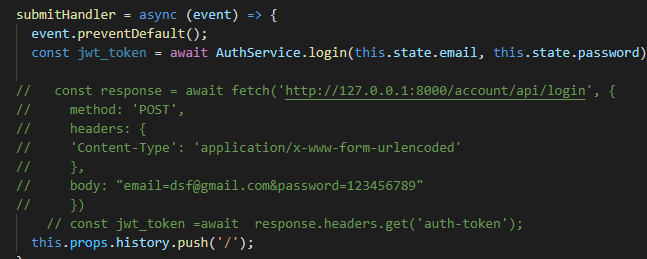


Add



Use context

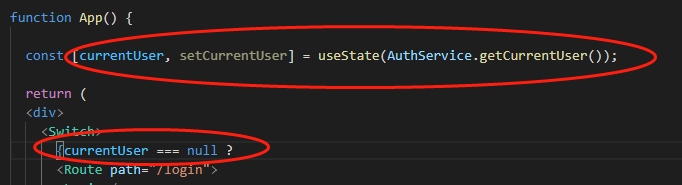




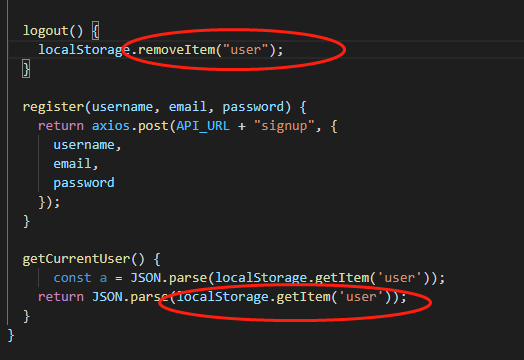
## 2.6 add auth

<https://bezkoder.com/react-jwt-auth/>

Add state to keep track of user login



Add localStorage in auth service



Add logout event handler