**Complete React Developer in 2021**

30th Dec ‘20

React-Saga: a scalable way to handle asynchronous actions in the app

Jest library for testing

MVC style started with Angular in 2010 from google inc.

What is AJAX? AJAX = Asynchronous JavaScript and XML. AJAX is a technique for creating fast and dynamic web pages.

React was written by Facebook in 2013

In imperative way directly changes the various parts of the app in response to user events.

1. Don’t touch the DOM

React uses a declarative approach to development. We are not interacting with the DOM, which React takes care. According to the state changes, React updates the DOM

1. Build websites like lego blocks

React components are functions / classes that receives input as props process it and return.

1. Unidirectional data-flow

Data flows in one direction from top to bottom

1. UI library

React Native for mobile

React Electron from Desktop apps

React Blas for Terminals

React 360 for VR

2 main libraries: React, ReactDOM

Good React developers are good at: Decide on Components, Decide the state and where it lives, What changes when state changes.

Nvm is node version manager. Nvm install <version>

<https://codesandbox.io/s/new> -- is react in the browser.

Npx allows to run app without installing it. Npx create-react-app my-app

Reactjs.org

For git bash, install from gitforwindows.org

Yarn is created by facebook

ReactDOM is the robot that interacts with the real DOM

A good rule of thumb as to when to use the key attribute you saw in the previous video, is this: Anytime you use the map() function inside of render, or you have a list of the same jsx elements one after another, they need a key attribute

useState and useEffect hooks has to be used only in custom hooks or at root of the function.

useEffect is to manage side effects.

The function inside the useEffect runs after every render cycle.

useCallback(function, [dependency]) caches the function thus prevents re-rendering.

useRef used to create a reference to a DOM element.

Using useReducer, whenever the state changes, component will rerender.

useMemo calls the function only when the dependencies change.

Through custom hooks, we can share the logic to components, not the data. Each component will have an independent implementation of the hook.

When we import class Component in a class, we get the life-cycle methods.

Both the class/function components has to be written in Capital case

One of the properties of the object ‘props’ are the children.

Learn grid and flex in css (card list is grid and card is flex?) transition->/transform->translate

Child components receive state as ‘props’.

Input element has a ‘search’ type.

Synthentic event is even on the browser

setState is an asynchronous function in JS. So, if want to do something after the setState, run it in the 2nd argument of the setState function.

Functional components takes some props and returns some html

If not using state, life-cycle methods, then use functional components instead of class components.

When defining method on a class, bind them inside the class constructor as – this.handleChange = this.handleChange.bind(this)

If we set an arrow function e6, js automatically binds that method to the class, so no need to bind.

‘this’ refers to context. Context is important.

The life-cycle methods in React is provided by the ‘Component’ class.

Video 47 for deploying project to github.

Yarn upgrade, npm update

setState is an asynchronous function.

In update phase, newporps, setState, forceUpdate before shouldComponentUpdate

<https://github.com/zhangmyihua> -- for code reference

npx create-react-app crwn-clothing --template cra-template-pwa

rmdir <folder> /s /q ------ to remove the folder along with its contents

sass – is an upgraded version of css. Saas-lang.com

yarn add node-sass (npm install // yarn add)

SASS (Syntactically Awesome Style Sheets) is a pre-processor scripting language that will be compiled or interpreted into CSS. SassScript is itself a scripting language whereas SCSS is the main syntax for the SASS which builds on top of the existing CSS syntax. It makes use of semicolons and brackets like CSS (Cascaded Style Sheets).

SASS and SCSS can import each other. Sass actually makes CSS more powerful with math and variable support.

Let’s list down the main difference between SAAS and SCSS.

SASS is used when we need a original syntax, code syntax is not required for SCSS.

SASS follows strict indentation, SCSS has no strict indentation.

SASS has a loose syntax with white space and no semicolons, the SCSS resembles more to CSS style and use of semicolons and braces are mandatory.

SAAS file extension is .sass and SCSS file extension is .scss.

SASS has more developer community and support than SCSS.

backgroundImage requires ‘url’ as property.

Npm update –D to update all package dependencies.

Because of the history api, now spa’s can move back and forth in the browser.

If there is a conflict of version while installing apps, in package.json, enter as ‘resolutions’ { package: version}

When router components are wrapped inside ‘switch’ when react finds the routed component, it loads and stops there.

In 2 ways we can navigate in a page, using Link and history.push()

Location pathname tell us where we are currently.

withRouter is a HOC(function) an hoc takes another component as argument and returns a modified component. withRouter(component)

SVG stands for Scalable Vector Graphics.

SVG defines vector-based graphics in XML format.

In order to use Firebase in our project, yarn add firebase

Import firbase from ‘firebase/app’

Import firebase/firestore

Import firebase/auth

Const config = {}

Firebase.initializeApp(config)

In firebase, a collection will have documents and documents will have their own collection and documents

Firestore returns 2 types of objects: reference and snapshots. Of these collection and document.

documentRef.get, set, update, delete. collectionRef.add

documentRef returns a documentSnapshot object

collectionRef returns a querySnapshot object.

Redux is a library that makes state management easier.

Redux uses flux architectural pattern - action -> dispatcher -> store -> view

Store UI based data in the State.

Redux don’t mutate data

Provider, connect is from react-redux. combineReducers, createStore, applyMiddleware is from ‘redux’

Middlewares receives action, do something with it and passes to the reducer

Import logger from ‘redux-logger’ – is a middleware

Javascript contexts: global, function, eval. Window is a global object

Using ‘this’ keyword inside a function to assign arguments to value, an object can be creted

Function createObject(a,b,c){

This.a = a,

This.b = b,

This.c = c

}

Cont ob = new createObject(1,2,3)

--

Reselect library helps memorization. Reselect will allow us to memoize and not re-render a component if the state value does not change.

Yarn add reselect

Import { createSelector } from ‘reselect’

In session storage, even if we refresh the page, session will persist. But once we close he tab/page, data will lose.

Window.sessionStorage // window.localStorage

localStorage will persist until we clear out

const obj = {name:"jaiosn"}

window.localStorage.setItem("mykey",JSON.stringify(obj))

window.localStorage.getItem("mykey")

const retr = window.localStorage.getItem('mykey')

JSON.parse(retr)

Import persistStore from ‘redux-persist’

Import persistReducer from ‘redux-persist’

Import storage from ‘persist-storage/lib/storage’

Import { PersistGate } from ‘redux-persist/integration/react’

Stripe is a platform that allows us to handle online payments.

Yarn add react-stripe-checkout

Import StripeCheckout from react-stripe-checkout

Css all share one single global namespace

BEM block element modifier

Css issue is styles leaking across components because of namespace sharing.

Styled-components is a css library. The styled components will get a unique class name

Import styled, {css} from ‘styled-components’

Const Text = styled.div`

Color: red;

Font-size: 12 `

Block css code can be written inside the css template and interpolate inside the div template

A hoc is like a class factory in python. It takes a component as argument, modifies it and return.

Function { class render return } return class

Observable: subscribe to the observable events, do something on next action, do something on error, unsubscribe when done.

Redux Thunk cares about only functions, not objects

Reducers fire first, then sagas receive the action. From there, sagas can fire off new actions which in turn hit the reducers and other sagas as well!

React hooks was introduced in Feb 2019 with React 16.8

Hooks can be used only inside the function components, not inside the class components.

useState is used whenever a function needs internal state, and not when any life-cycle methods.

useEffect fire side effects inside the functional components.

Node is an environment outside of browser to run JS

Express is a framework for node

By simply entering ‘yarn’ will install all the dependencies from package.json

**Apollo Server is an**[**open-source**](https://github.com/apollographql/apollo-server)**, spec-compliant GraphQL server** that's compatible with any GraphQL client, including [Apollo Client](https://www.apollographql.com/docs/react). It's the best way to build a production-ready, self-documenting GraphQL API that can use data from any source.

Prisma helps app developers build faster and

make fewer errors with an open source ORM for PostgreSQL, MySQL and SQLite.

The 3 most important parts of a progressive web app: 1) HTTPS, 2)App Manifest 3) Service worker

Any client request to the server should be secured.

Lets Encrypt, CloudFlare - for encryption

Service worker is a script runs in the background separate from the web-app

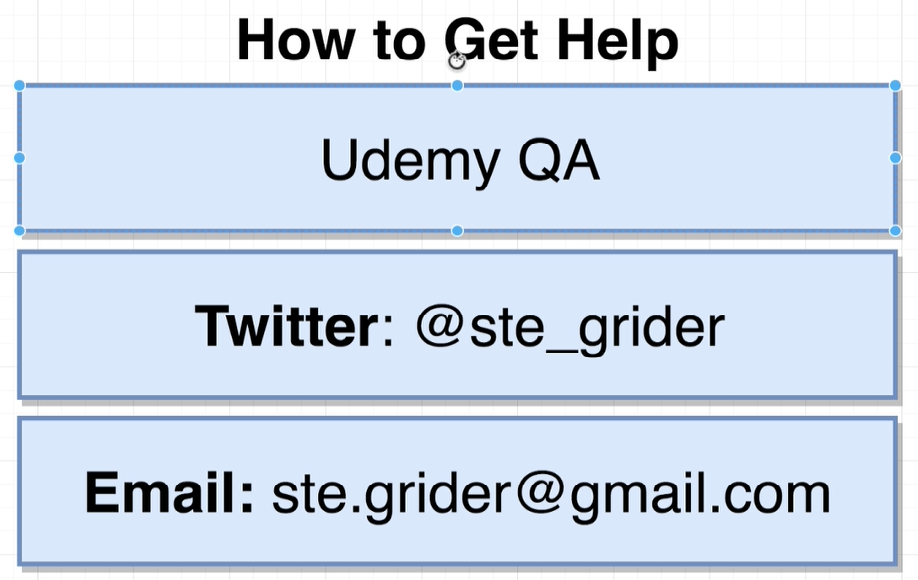
When a user opens the page of a pwa, the request 1st goes to the service worker.

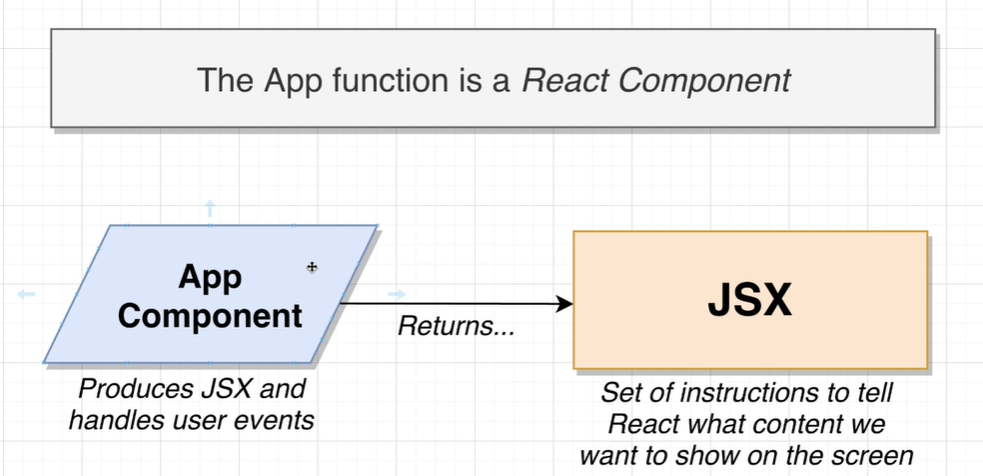
Gatsby is a framework for rendering static pages in a website. The pages are already loaded ono the server in the build stage.

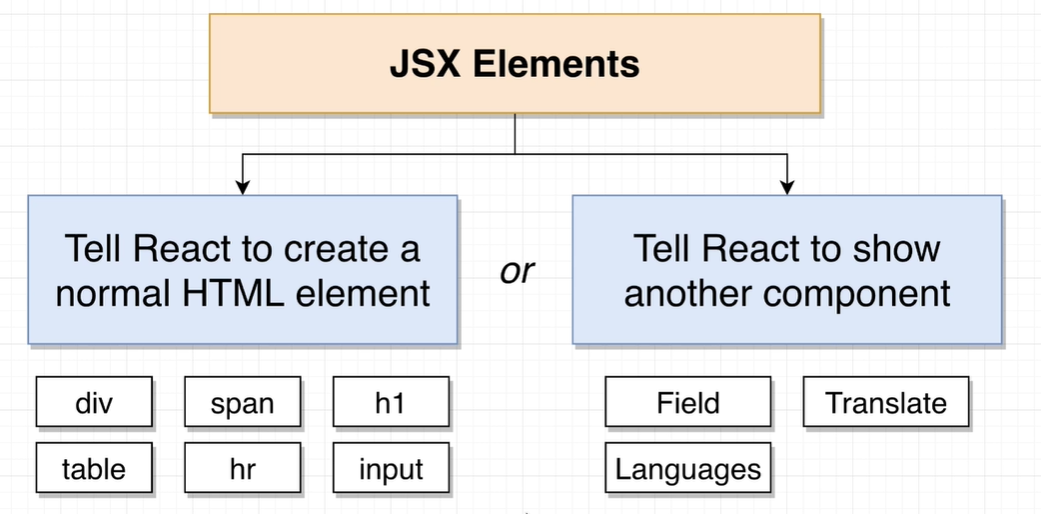
3 things in react: NextJS for sever side script, Gatsby, createreactapp

Modern React with Redux [2020 Update] - By Stephen Grider

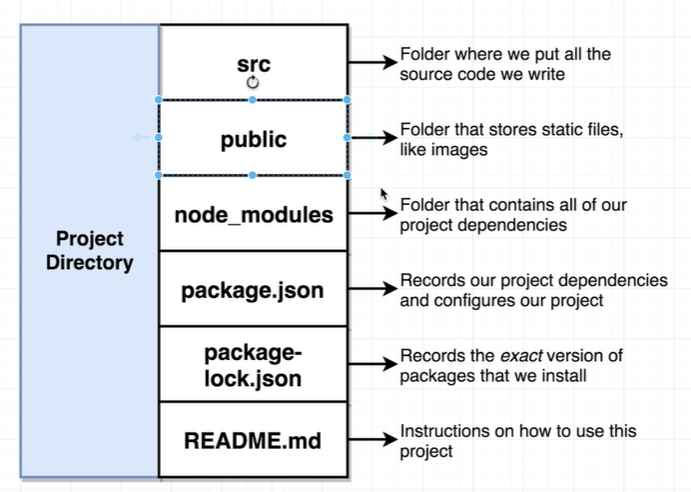
07-Mar-21

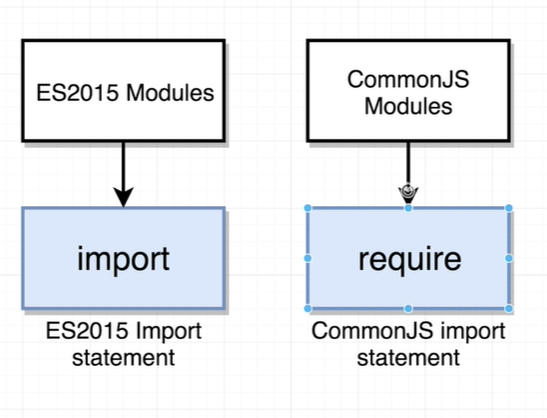




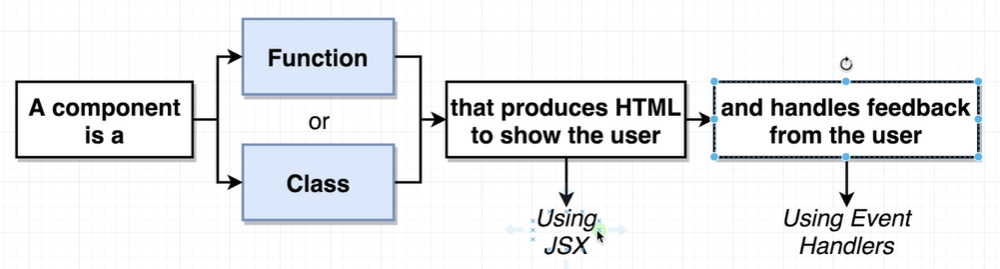






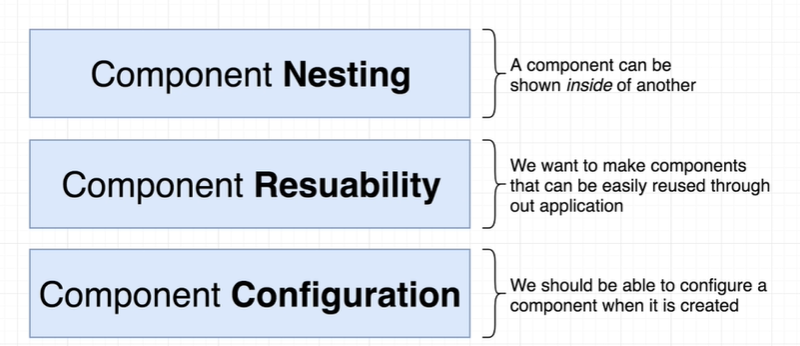


Modules are imported from node modules

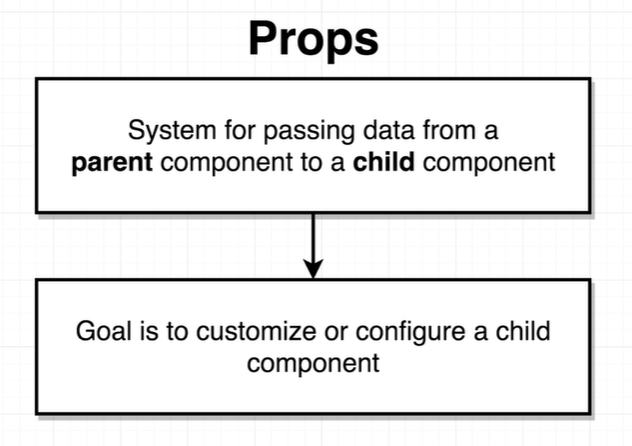


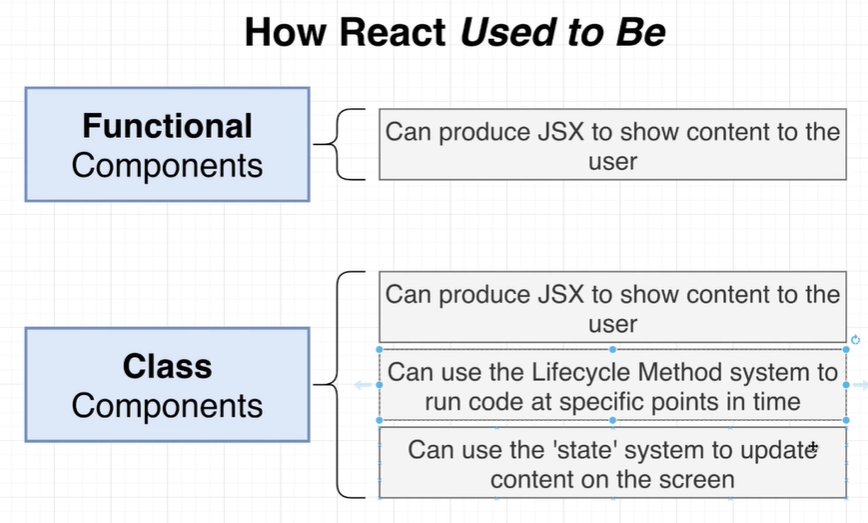
**React**. **createElement**() method takes the three **arguments** type , props ,children.

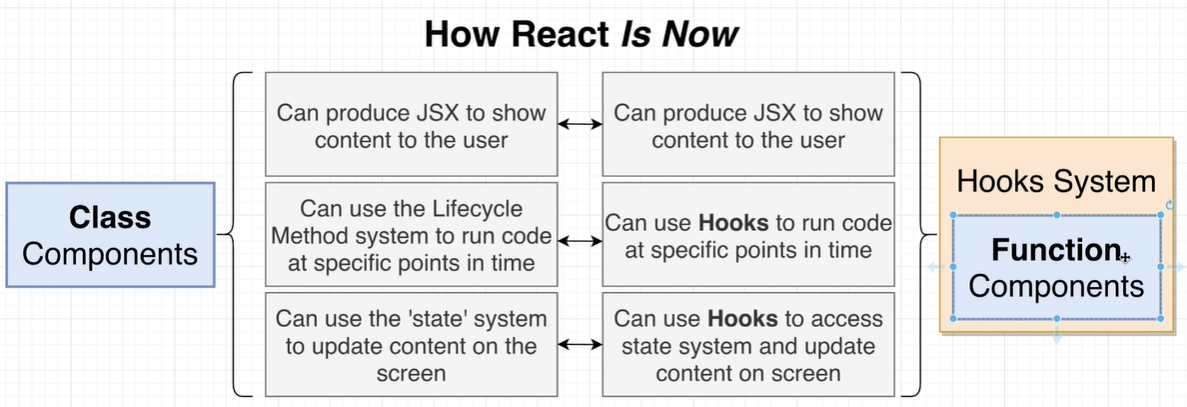
Tenets of a component:

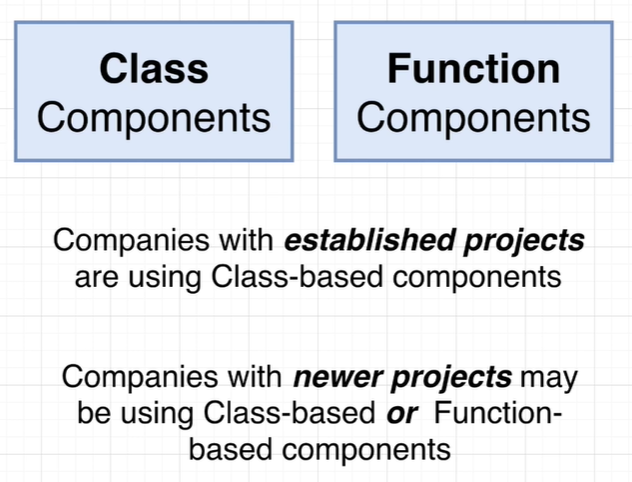


Faker.js // npm install faker // call faker.image.avatar as - src={fatker.image.avatar}





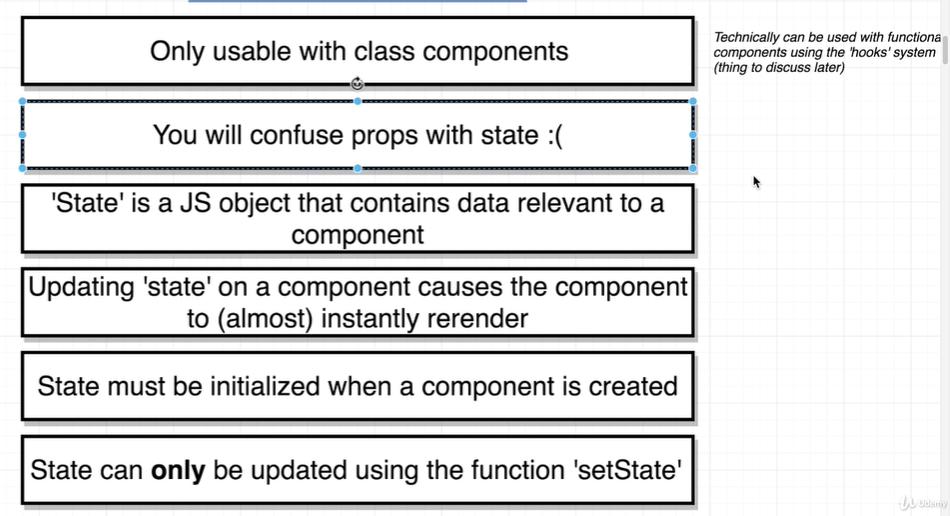




Window.navigator.geolocation.getCurrentPosition:

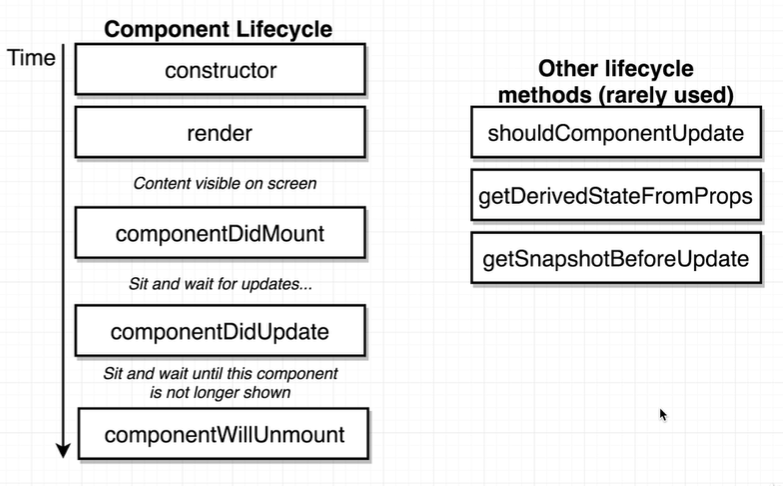


State rules:



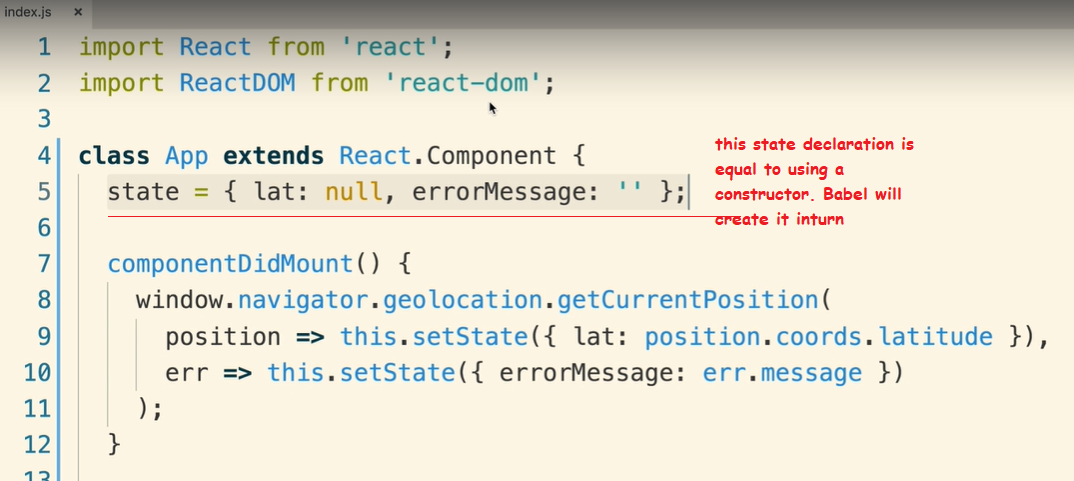
For each state modification, render is called whether that state reference is used inside the render or not.

Component life-cycle methods:





Do not do data loading inside a constructor function



npm install semantic-ui-css

Then, import the library into your root index.js component:

import "semantic-ui-css/semantic.min.css";

default props:

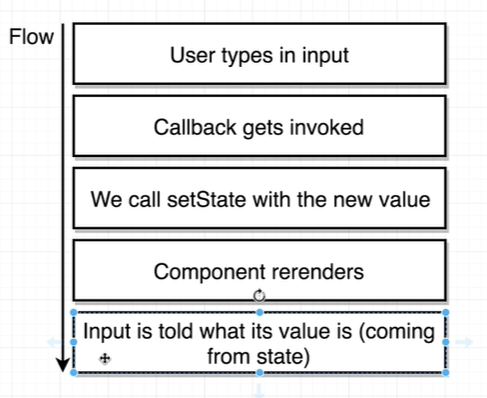


Avoid conditional in render

Controlled element:

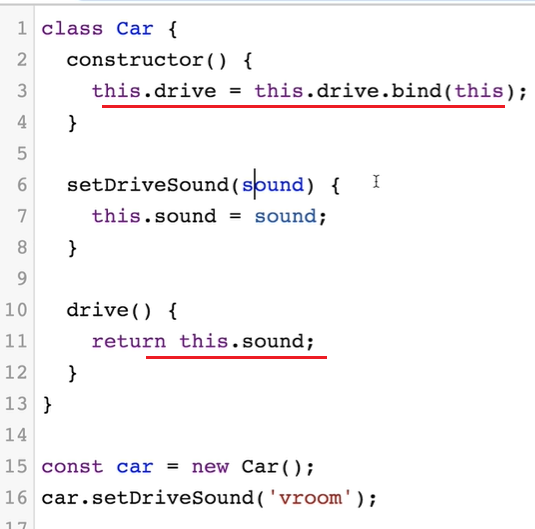


Controlled element work-flow:



Uncontrolled element will fetch its value from the DOM Html

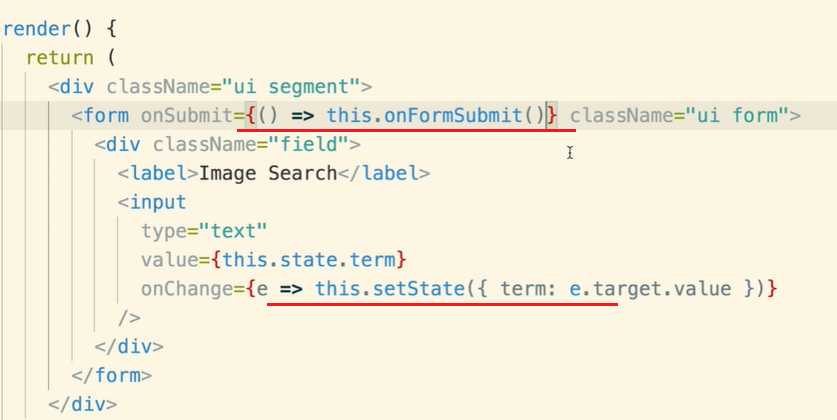
One way of binding ‘this’



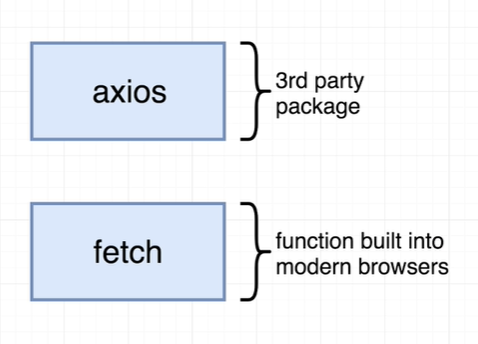
Another way of binding ‘this’ through of use arrow function:



Another way of using ‘this’ by arrow function inside: when using arrow function here, invoke the function by ‘()’



Getting data remotely: axios, fetch:





Steps:



An asynchronous function returns a promise

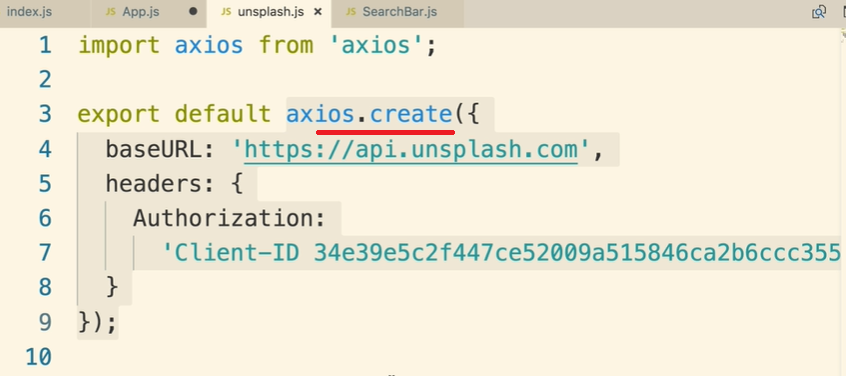
Async await syntax:



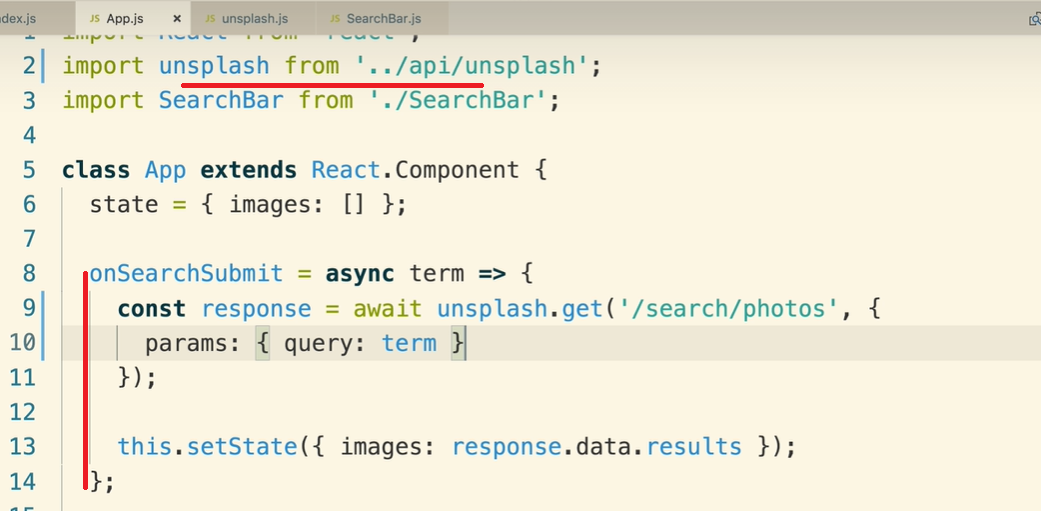
Promise syntax:



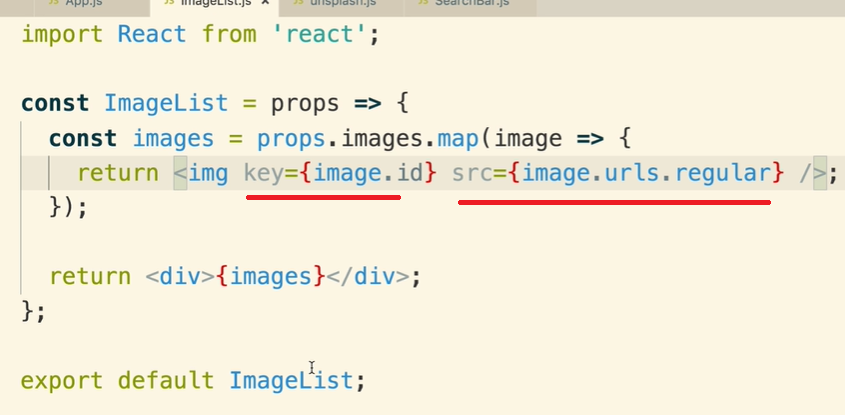
Axios.crate :



Remaining code for axios:



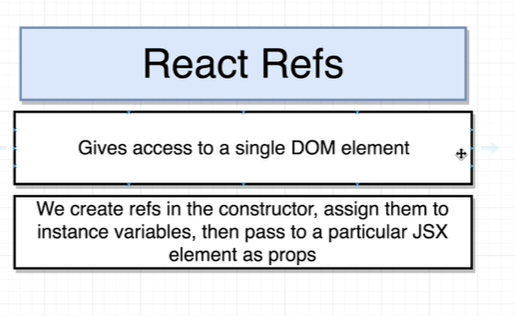
Display a list in a component:



Object destructuring:



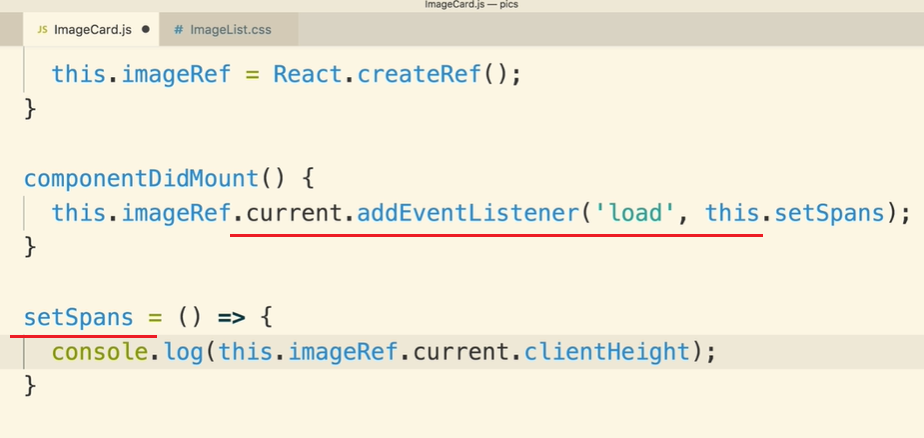
React Ref’s :

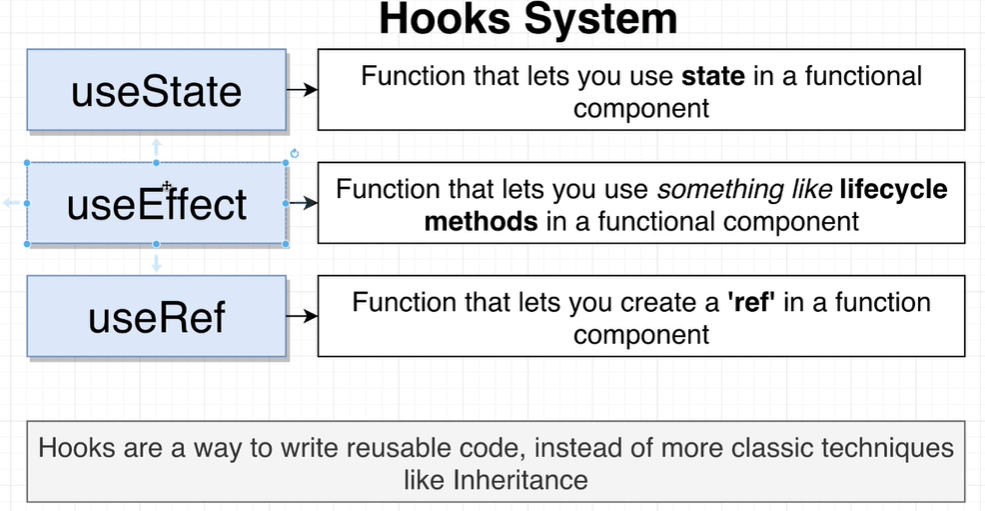


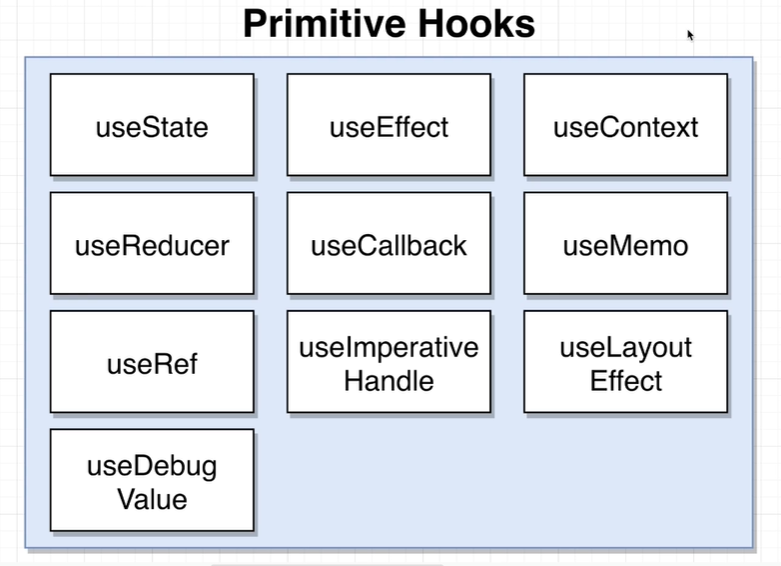
ref:



addEventListener:

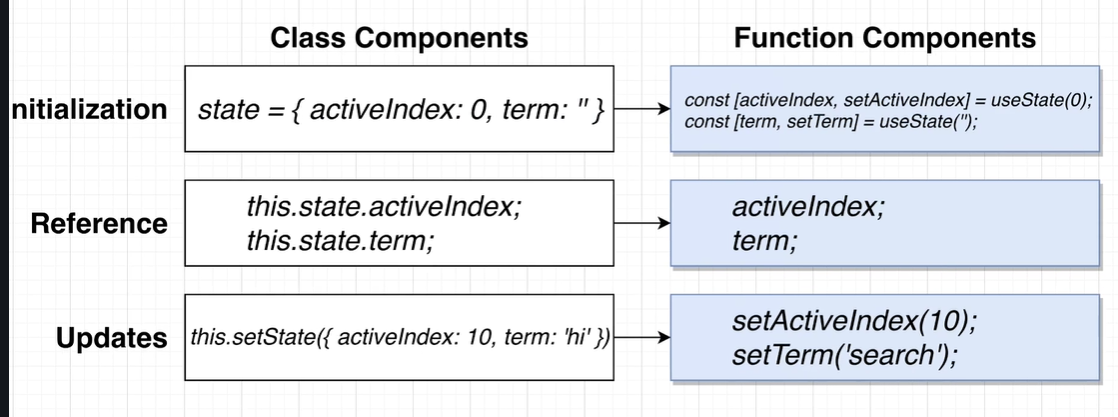






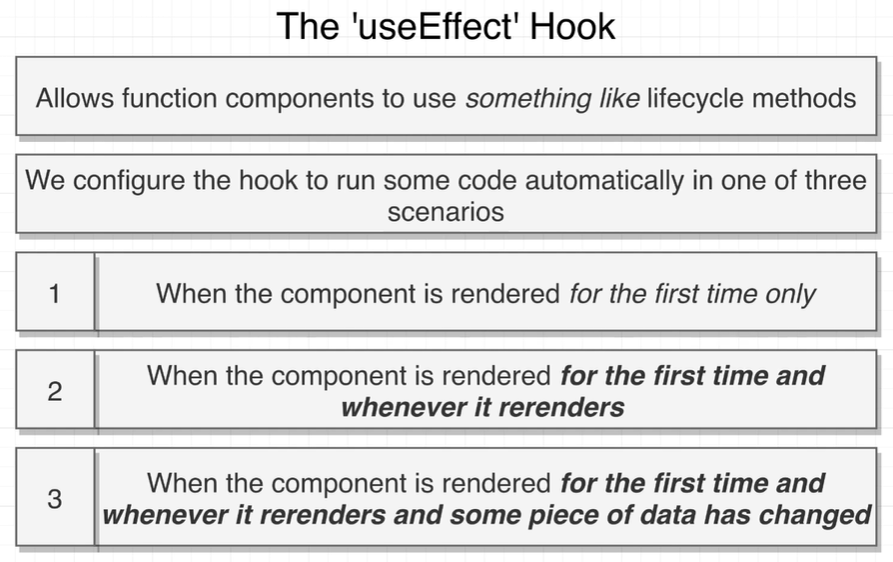
Accordion is like a question answer content

setState and useState:

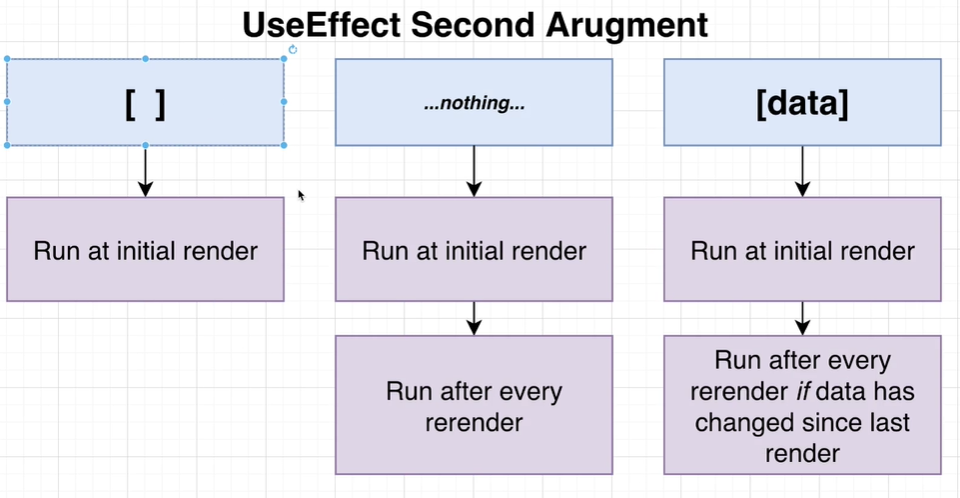


Semantic ui ‘active’ class causes the expansion and collapse of a question answer accordion.

useEffect:



useEfffect second argument:



\*starend