

Now we are going to tell you that in class component how you can use state and props

In the previous lesson we talked about class component how you can make it and use it and everything so teacher is going to continue from there

We will use the class component of previous lesson and there pass the state and props update it ,define it

So first of all we will do is that we will define a state in class component and will print that value and how to update it on event and then we will see how to pass the props there and in a class component How receive it and display it.

Many interview question will be cleared

Now a question arises when super keyword is used you will know this in current lesson which is a very important interview question in js, typescript and OOPs it is often asked.

Now how to define state in a class component whenever you want to define a state in class

Then you need a constructor.

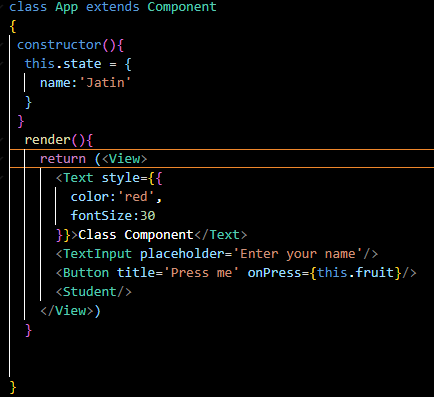
So in our class we write constructive inside which we write

This.<statename> = {}

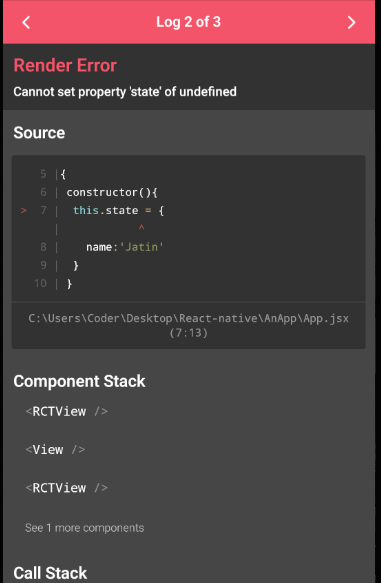
Now whatever key we will put inside the object ( that is made by curly braces ) will be treated as state and its value will be set as default you can give any value that is undefined or null even

So by this way we create a state name

As shown in the code below



Now when we refresh our program we see an error as shown below



Because Until you call super(), this is **undefined** because child class is built on parent class sp we need to define parent by super and then child is defined this practise is followed in java also but in java if super() is not included thn it still work because compiler be default adds in our code

According to teacher state is defined in parent class so we need to call super so that we can have access to the state from parent that is Component

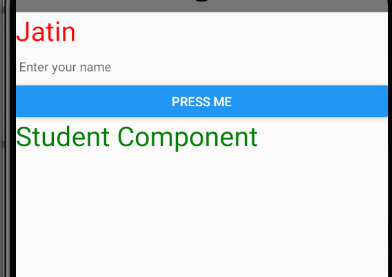
So to call parent constructor we use super this concept is similar in almost all language like Java, Typescript

You save the file now no error will be shown

Now if we use the state we made in constructor



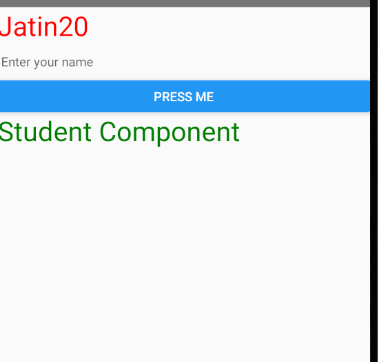
Now the output is as shown below



Whenever we use class or component property function then you have to “this”

Now we add another key age in the state and then we use it with our name as shown below





Now suppose if there is a TextInput box then in its on change whatever we type it should be set in state name how could we do this

For just testing we do such that we add an event on it that is we pass prop onChangeText to InputTextBox and we pass it callback function

And then inside the class we make another function name it anything you want which align with the purpose of the function we call it updateName

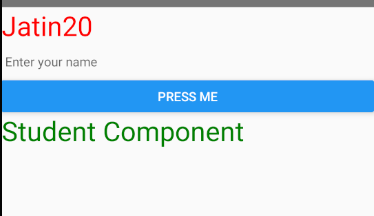
Now inside it we call the setState which is a by default function inside class component which update our state we pass it an object inside which you pas the key value pairs the key should be equal to the state name you want to modify and then value should be the new value you want for the state for now we keep the value as “Hello” and now we call the update name function in the call back of onChangetext in InputbBox

Now the code look like below

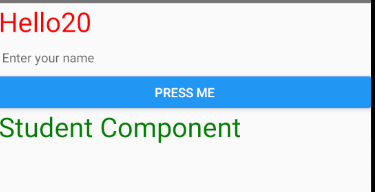


Now output will be like below

Before typing



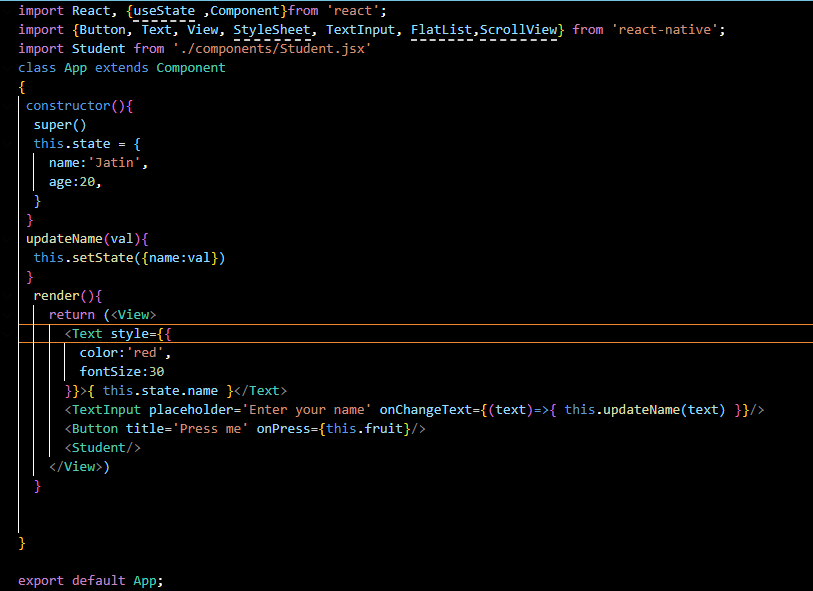
After typing



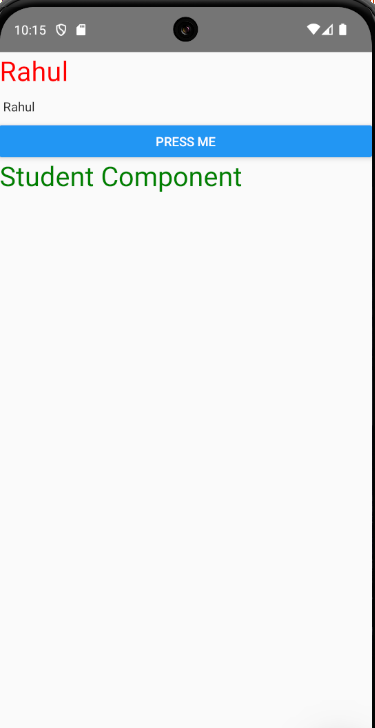
5:00

Now we want that the state should be equal to the textbox now we need a text parameter in our on Change Callback Now this parameter text contains all the text that is written in textfield and we pass it to updateName function and then this updateName set the name state equal to the string passed to the updateName method that was the text typed in the textbox

Now the code look like below



Now we type anything in the field it will be shown then before typing Jatin will be shown but after typing whatever we type then it will be shown in the title on the screen as shown below



So that’s how you can update your state in class component

Now we want that the Student component that we made in the last lesson Inside it we want to take the name meaning we want to take the name state inside the Student component and replace the   
“Student component” text in the screen above with our name

The thing is very simple that we pass props here that is our name and we pass it our anme state as shown below



Now we will go to Student component

Now we are going to use props in our render method we can use it another method

Now in render method we do console.warn method and then pass it the props and since our props contains name state the console will also print the newly text you typed in the textbox

Now if we typed it will look like below



As the console.warn or console.error is shown in Debugger

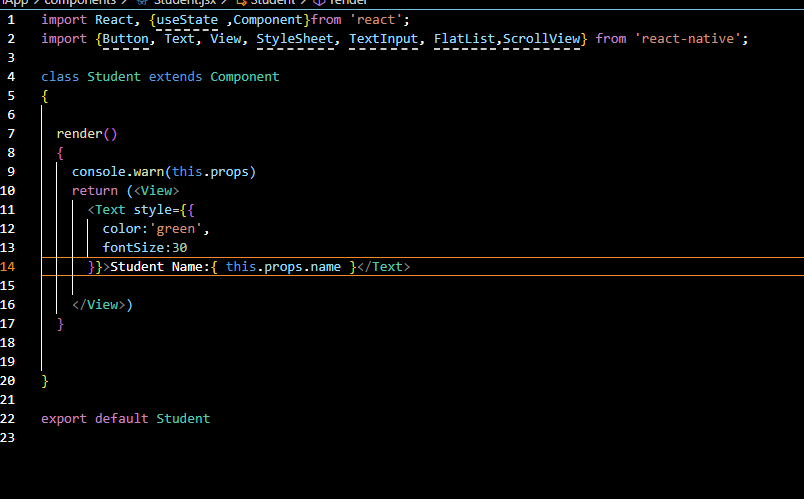
Now you can see whatever we type it will be shown in the debugger word by word

So now instead of writing console we will replace the text Student Component inside the Student component with the Student name: <name state passed as props>

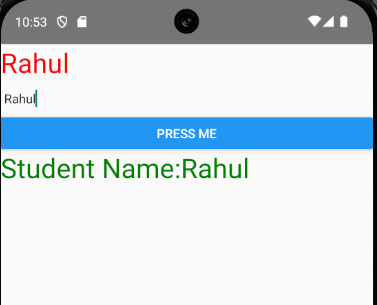
**Note:-**

We don’t have to user super as we use it inside the constructor to initialise the parent object because since we don’t write constructor here In Student Class it will be written by interpreter (who knows) and due to which props is easily available

Now our code that is of Student will look like below



Now output will look like below



Now whatever we type it will be shown on the render by Student component

Now in Interview question it could be asked that how to make a state inside class component and how to update a state, How to use props they could ask you use super keyword, how to create a constructor there are many question that could be formed

In the next lesson we will study about Life cycle Hooks

There will be 6-7 lessons here by this if we study life cycle hooks then you understand the core topic of react js like how to use state in life cycle methods what are life cycle methods and why to use it, What are Hooks and many things

Whenever you give interview then 50 percent are chances that question related to it could be asked from you because it is very important part that is life cycle of React or React Native , Angular js Technogen framework(who knows)