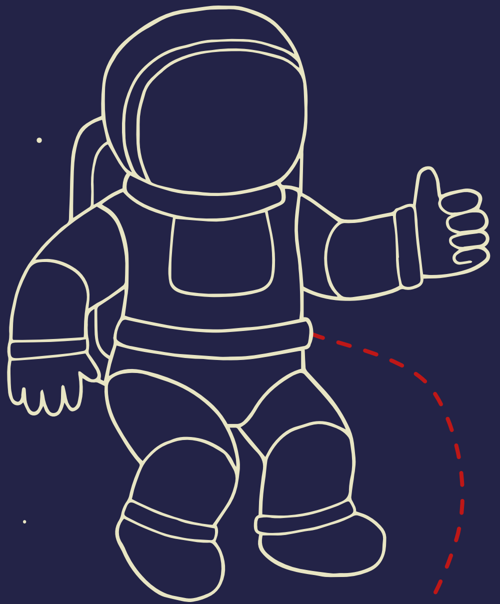
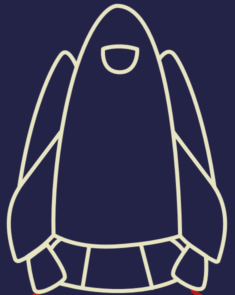
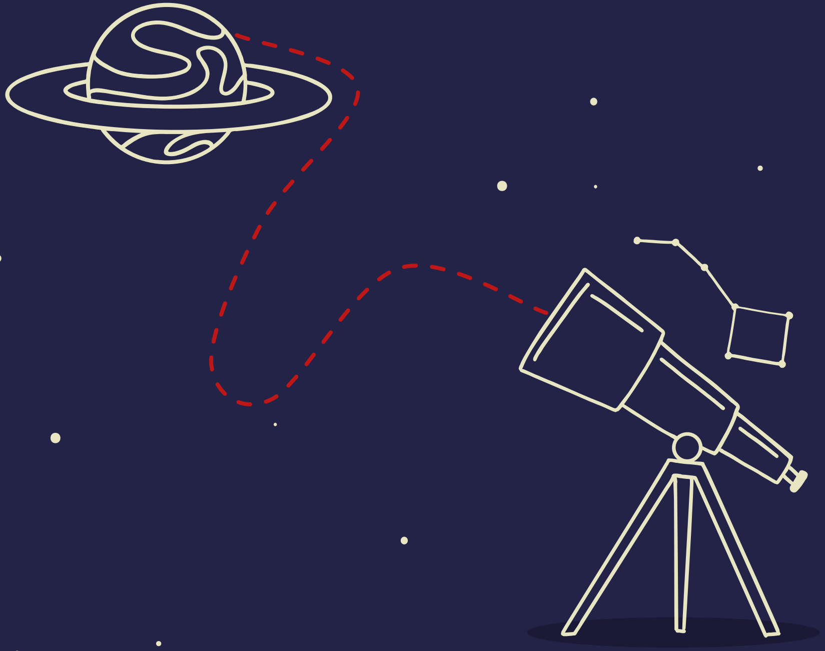


React Series

React Hooks
By Mithilesh Sir
Fascoon Eduversity

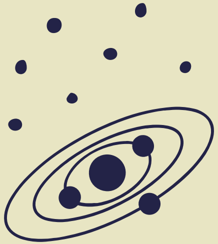


Creating Functional Components in React

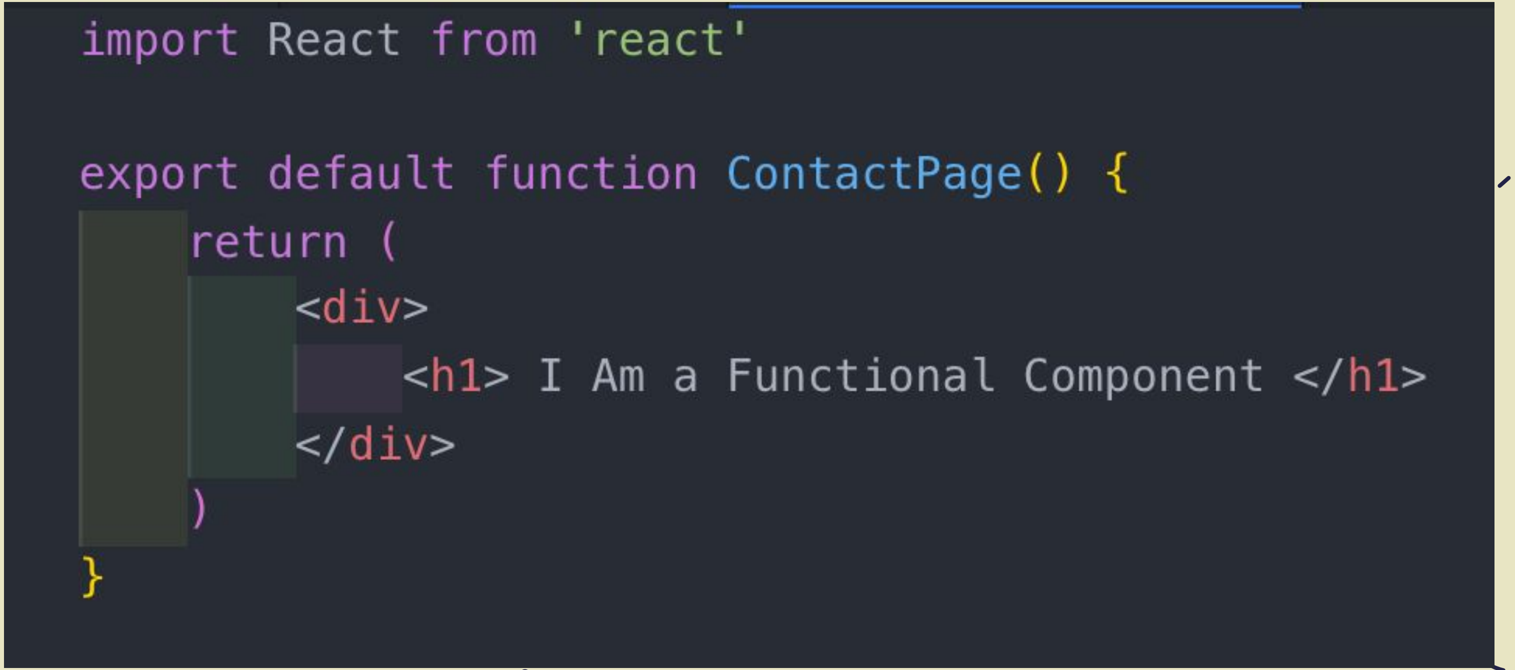


What are Functional Components in React

- Functional Components in React are also called dumb components.
- These Components do not have its own state.
- They are dependent on Props from other Components.



Example of Functional Component in React.

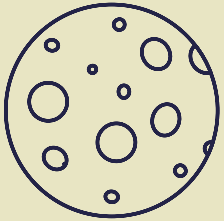


```
import React from 'react'

export default function ContactPage() {
    return (
        <div>
            <h1> I Am a Functional Component </h1>
        </div>
    )
}
```

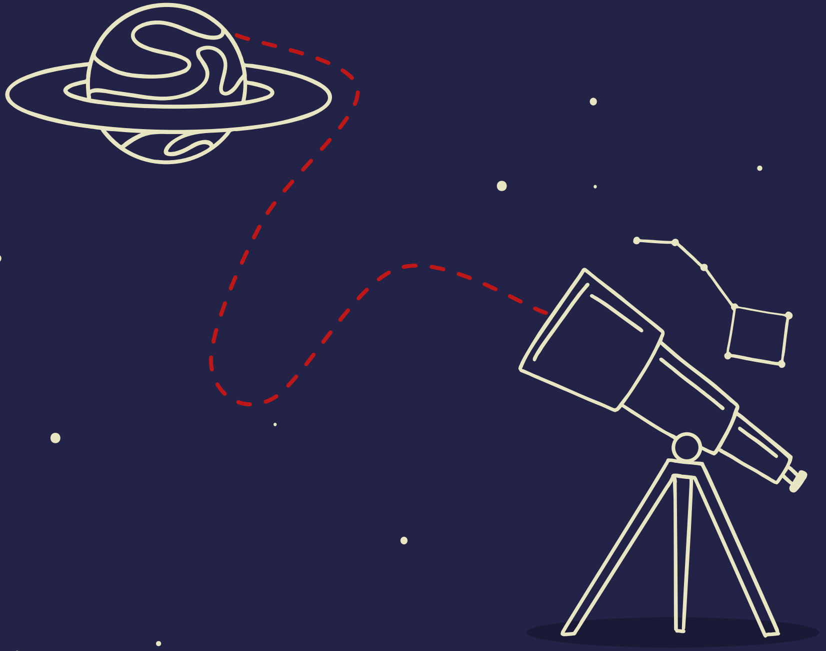
Can Functional Component have its Own State?

React 16.8 onwards

Functional Components are no more Dumb Components and can now hold its own state.

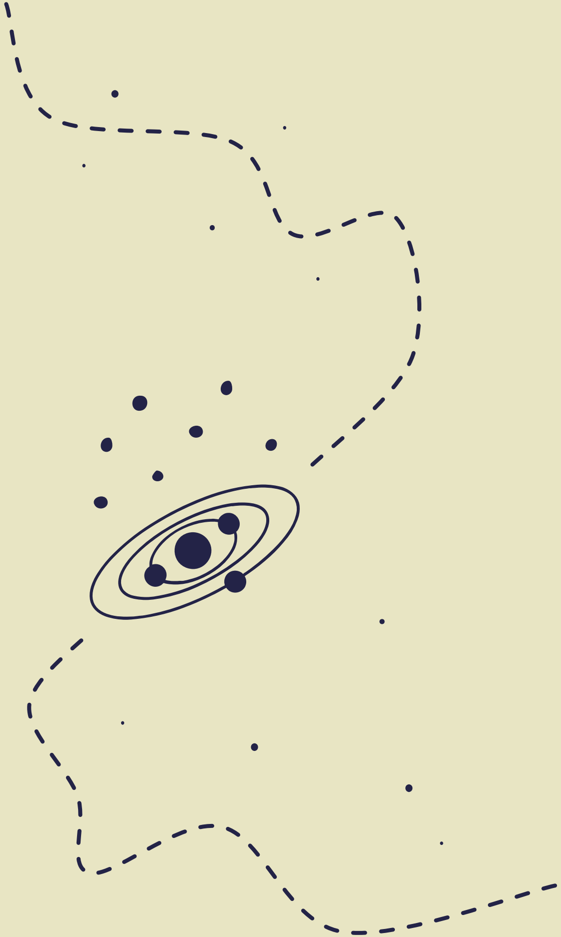


Introducing React Hooks



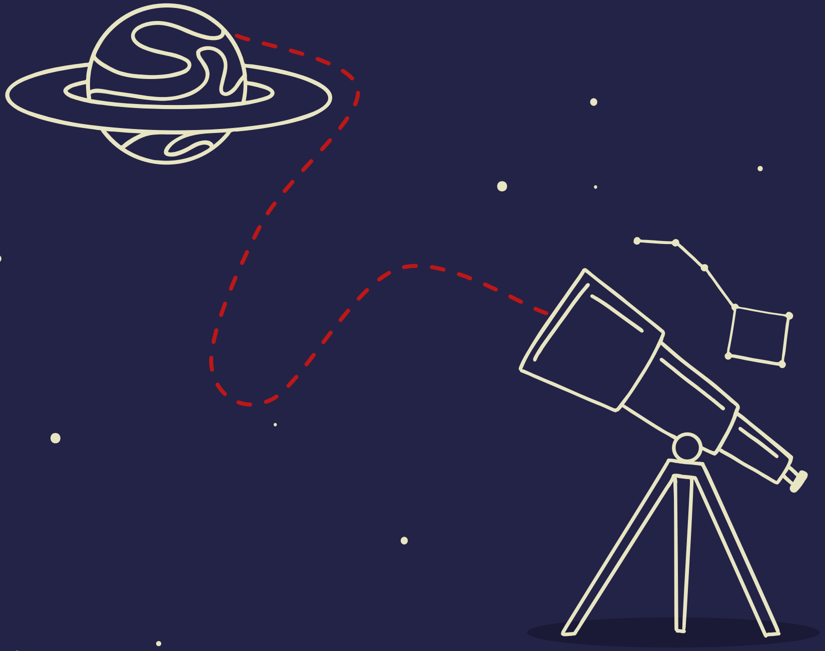
React Hooks are a GameChanger!

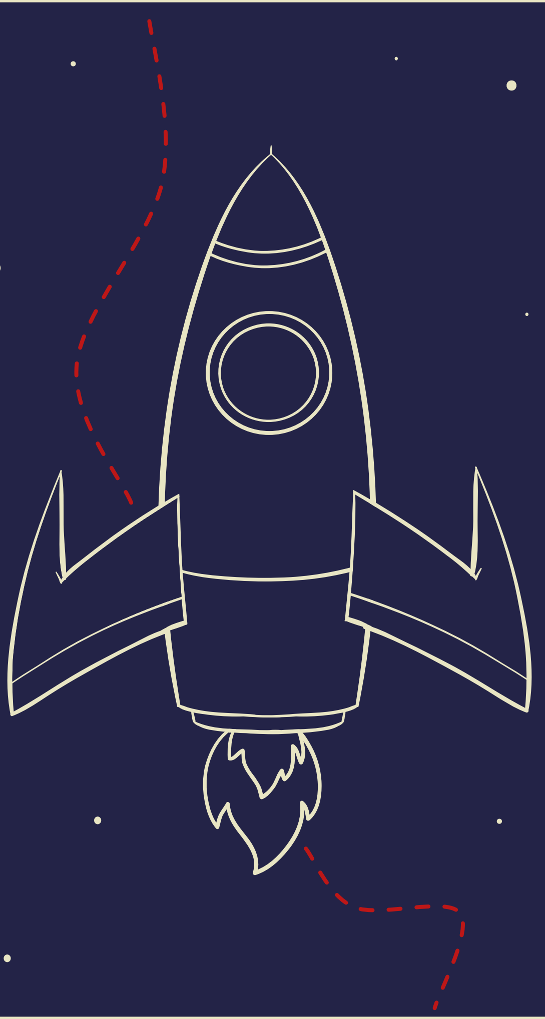
- With React Hooks you can declare state in functional components as well.
- So Functional components are no longer dumb components.
- No Need to use Class Components, Class Components can now be converted or replaced with Functional Hook Components.



Example

Example of Functional Hook Component in React



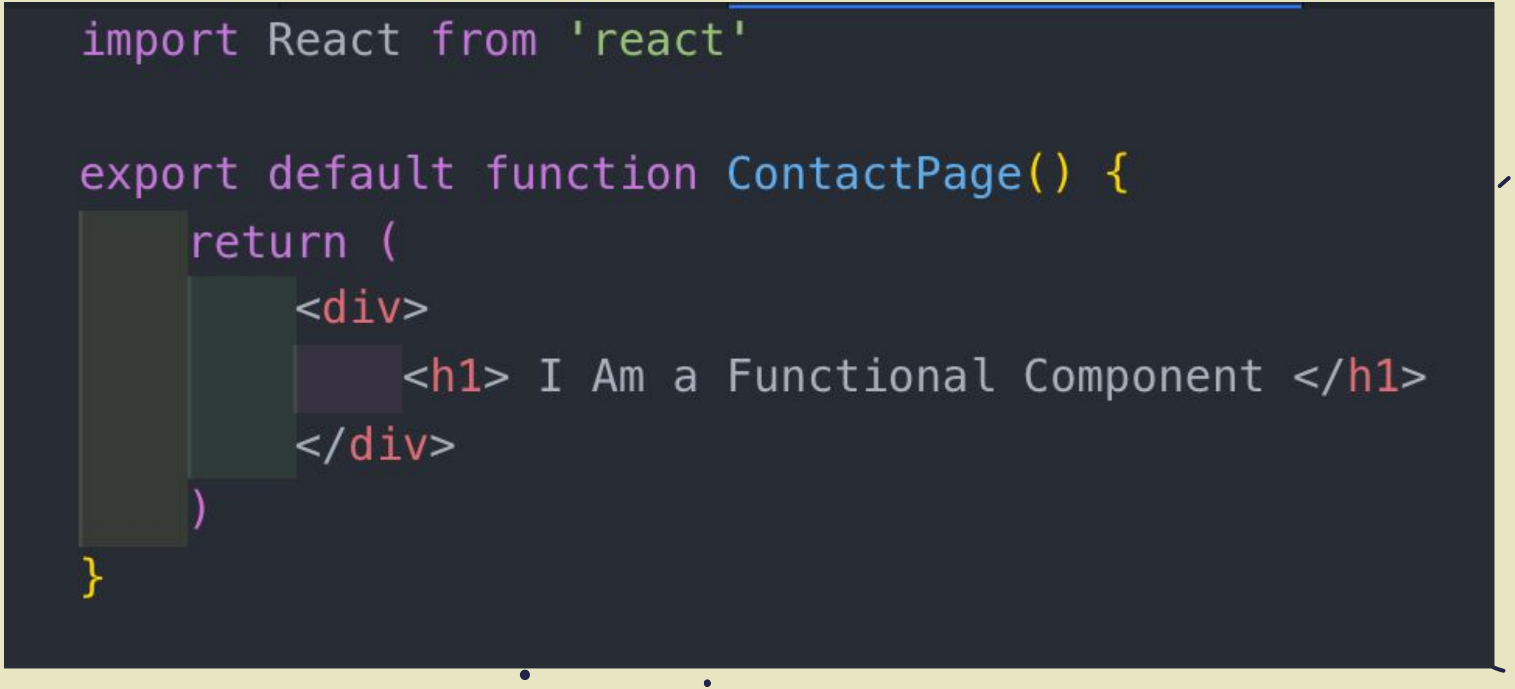


01

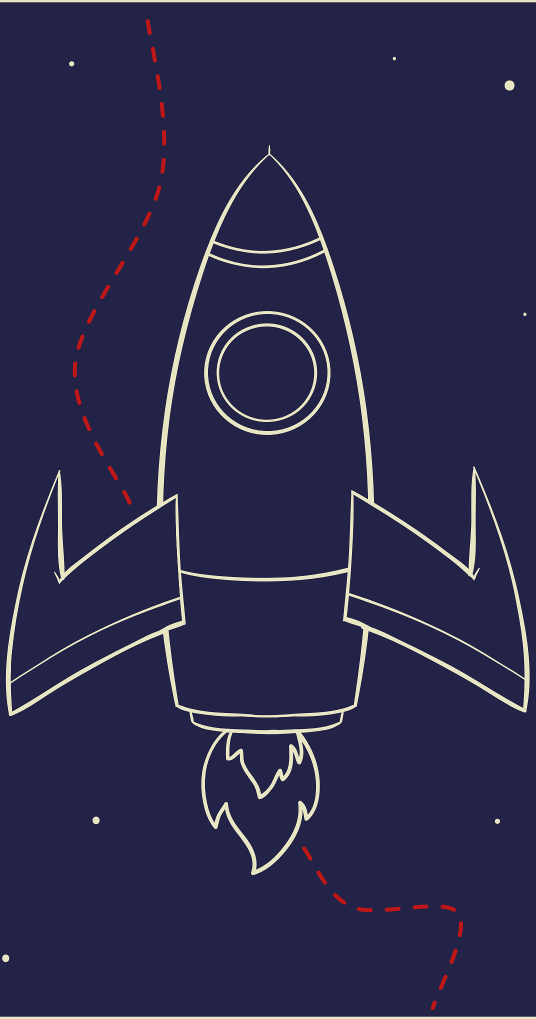
Step 1

Create a Functional
Component in React

Step 1: Create a Functional Component



```
import React from 'react'

export default function ContactPage() {
    return (
        <div>
            <h1> I Am a Functional Component </h1>
        </div>
    )
}
```

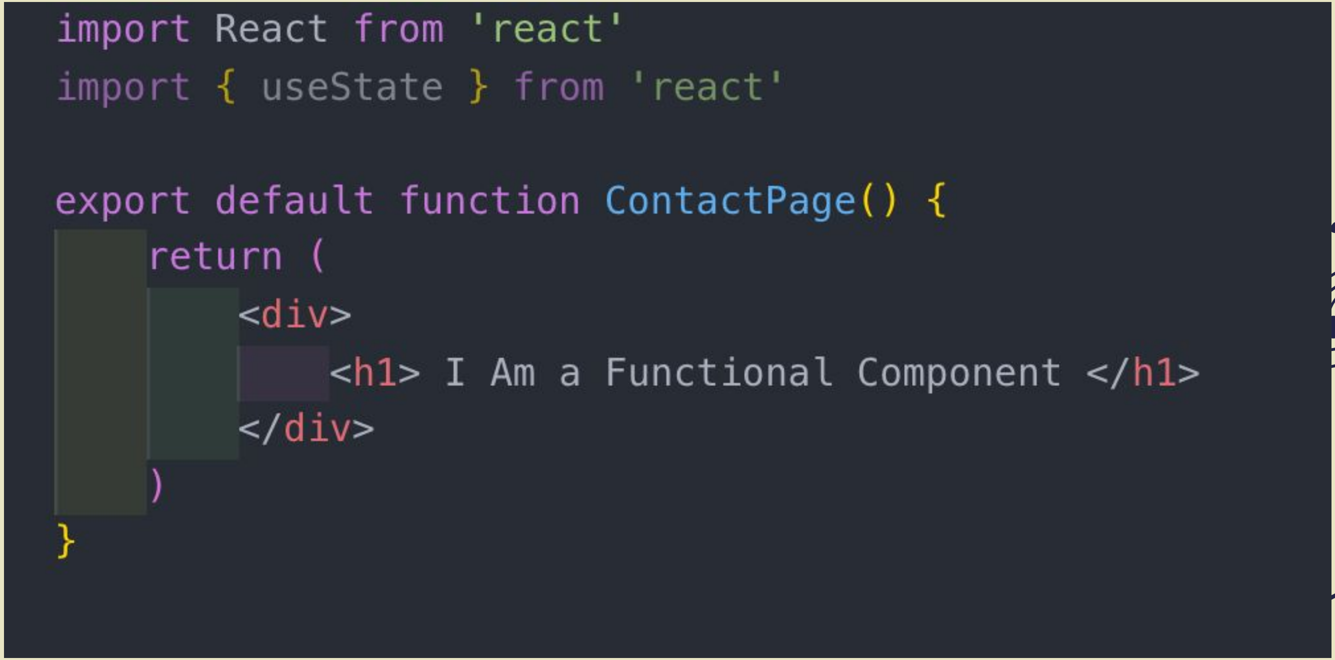


02

Step 2

Import React Hook Methods

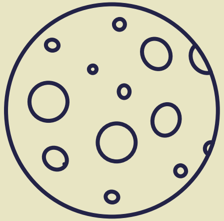
Step 2: Import useState

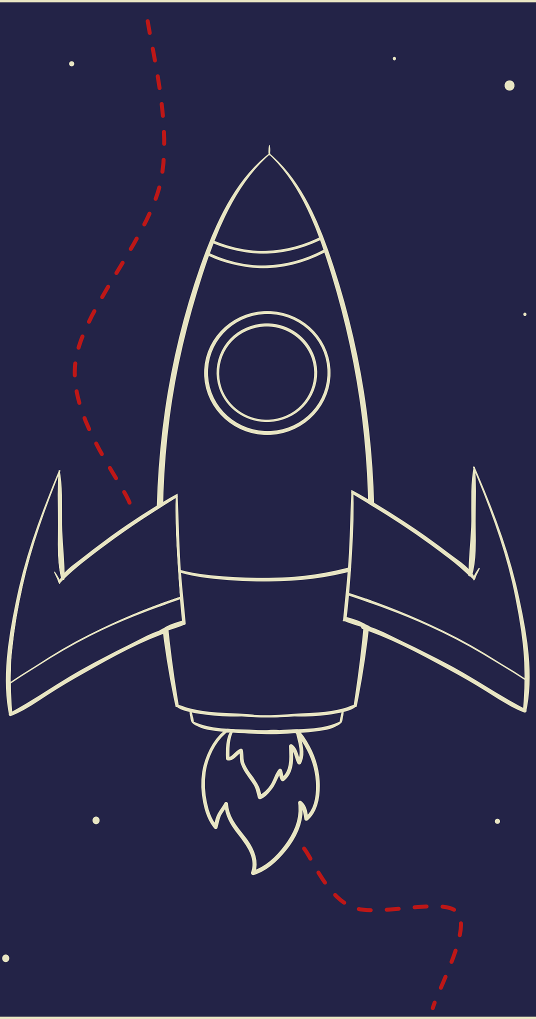


```
import React from 'react'
import { useState } from 'react'

export default function ContactPage() {
    return (
        <div>
            <h1> I Am a Functional Component </h1>
        </div>
    )
}
```

Note:

Methods of React Hooks are available inside React Library only. Hence we will import it from the React Library only



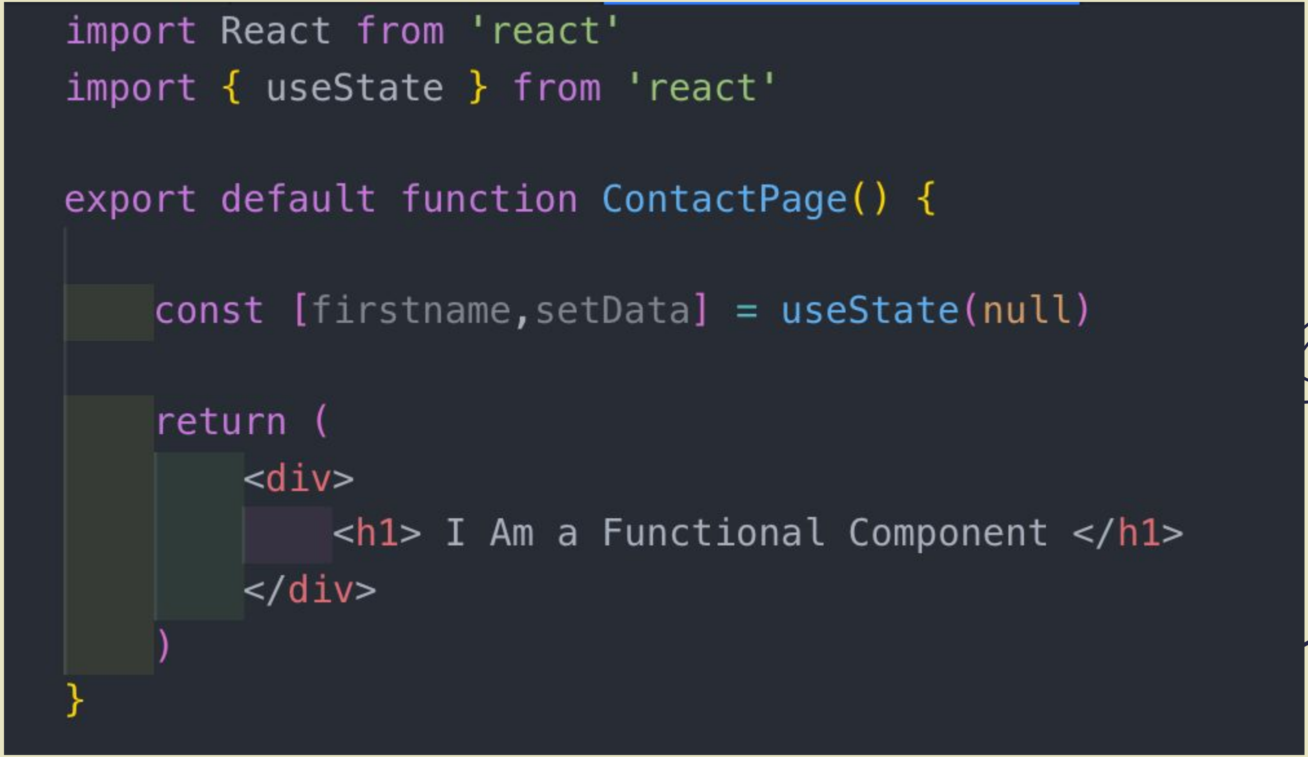


03

Step 3

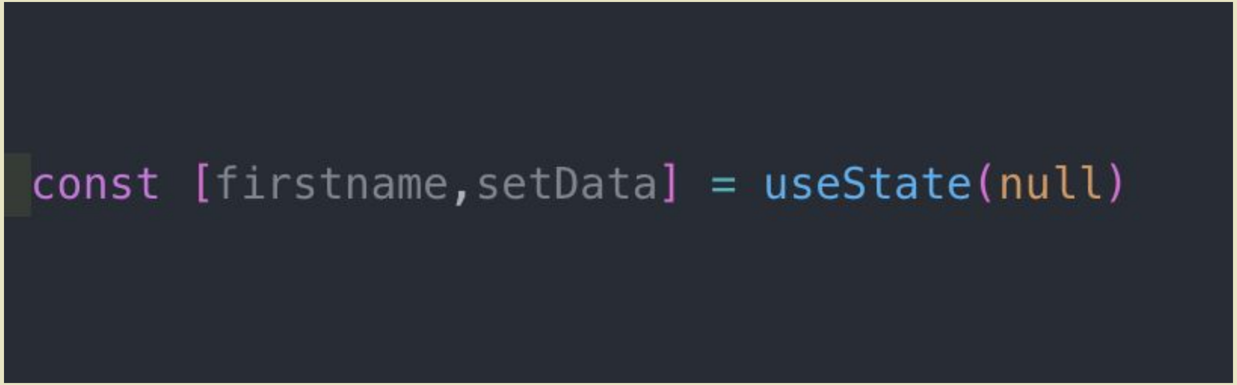
Create State Variable
inside the component
using useState Hook

Step 3: Create State Variable Firstname



```
import React from 'react'
import { useState } from 'react'

export default function ContactPage() {

    const [firstname,setData] = useState(null)

    return (
        <div>
            <h1> I Am a Functional Component </h1>
        </div>
    )
}
```

Code Explained

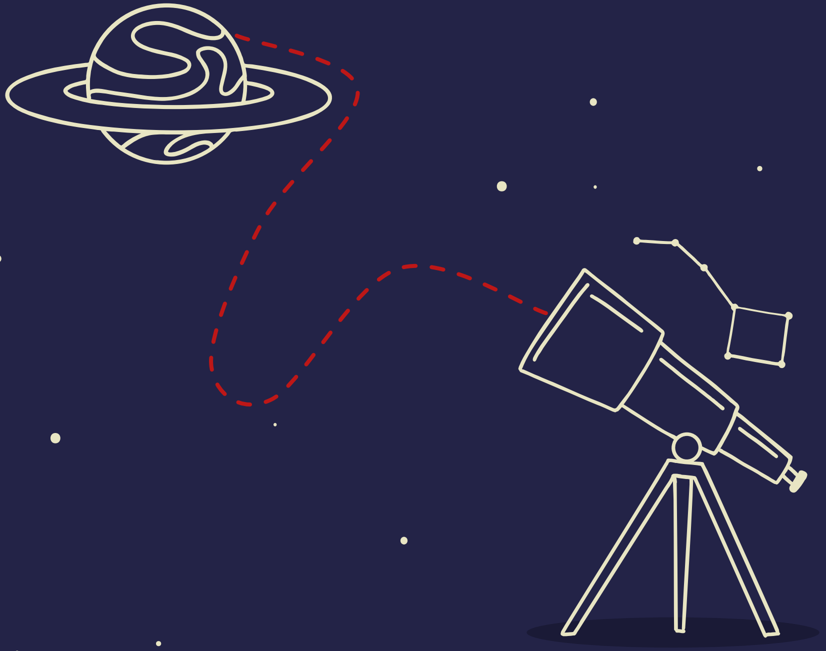


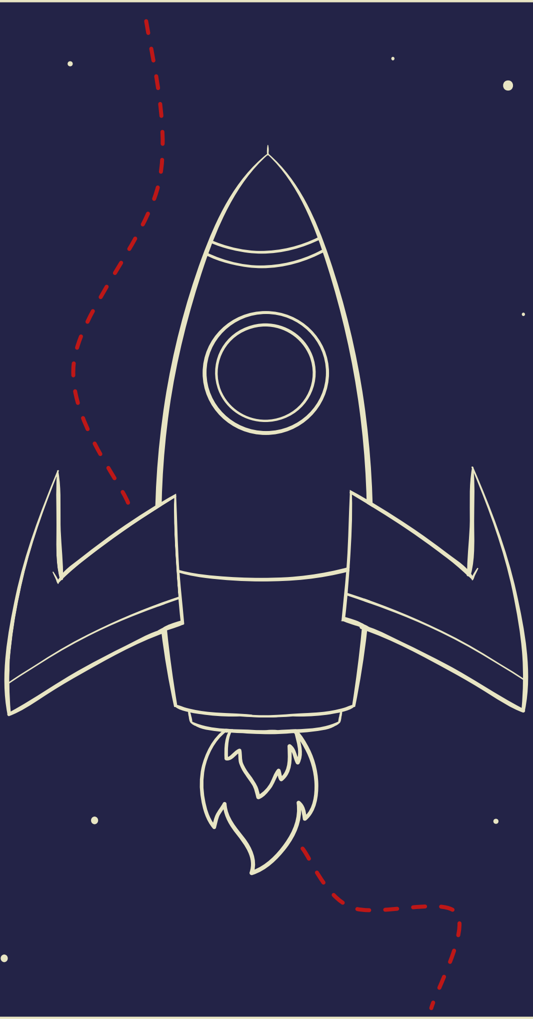
```
const [firstname,setData] = useState(null)
```

- firstname is the name of your state variable
- setData() is the hook method you have passed as second argument, it will be used to change or update the value of the variable.
- useState(null) means that the current value of firstname is null.

Print your State

Example of Functional Hook Component in React



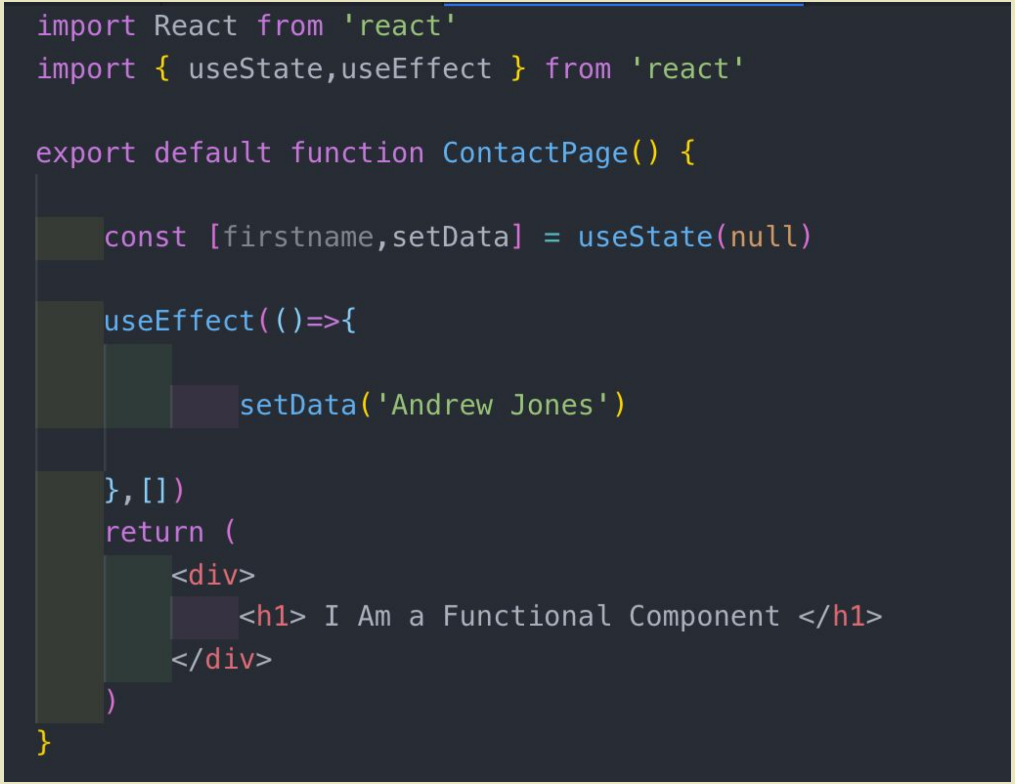


04

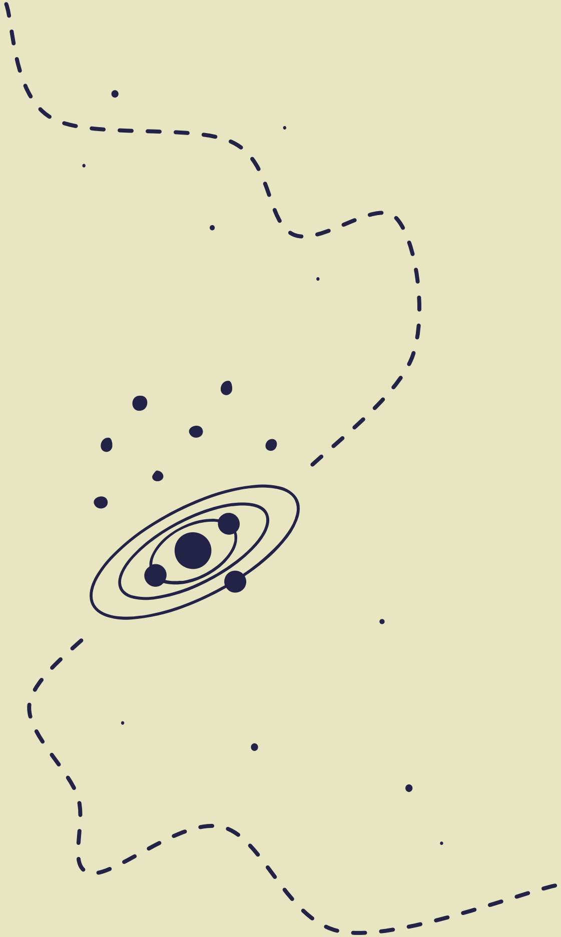
Step 4

Adding Effects using Hooks

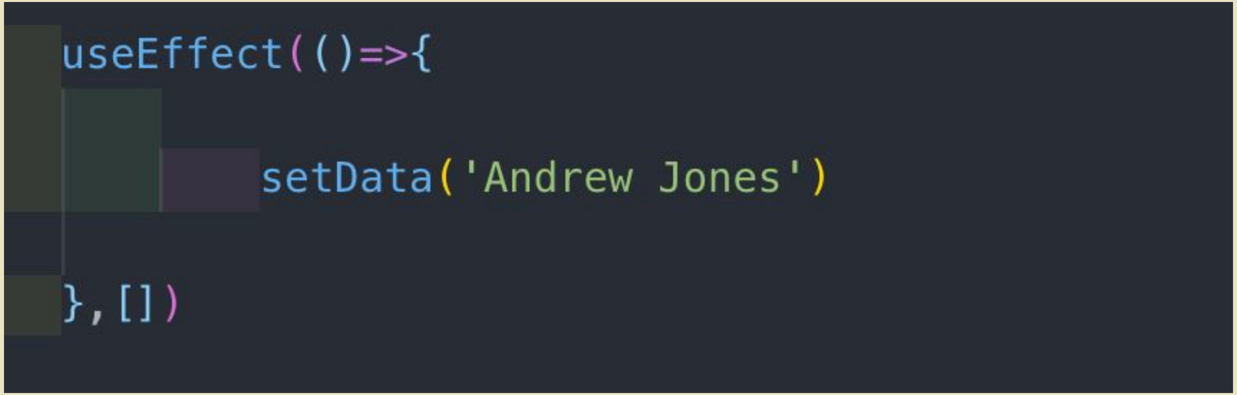
Step 4: Create your First Effect



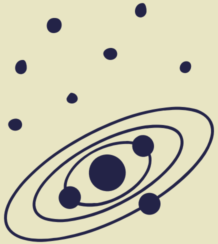
```
import React from 'react'
import { useState,useEffect } from 'react'

export default function ContactPage() {

    const [firstname,setData] = useState(null)

    useEffect(()=>{

            setData('Andrew Jones')

    },[])
    return (
        <div>
            <h1> I Am a Functional Component </h1>
        </div>
    )
}
```



Code Explained

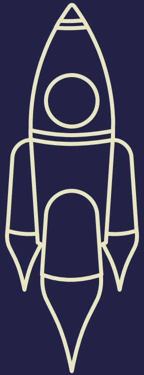


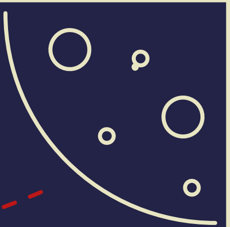
```
useEffect(()=>{

        setData('Andrew Jones')

},[])
```

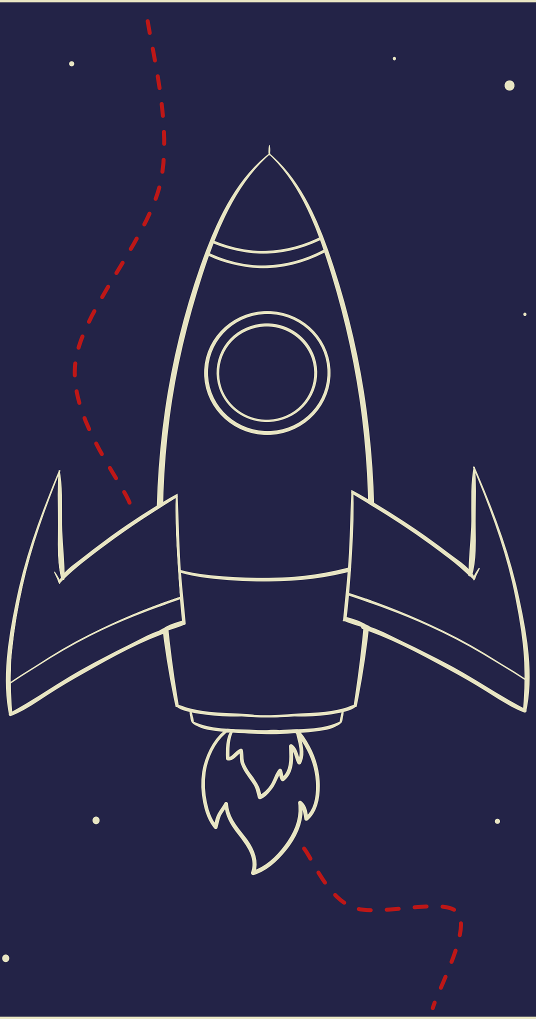


- useEffect() is a method of React Hooks to create effects in your component.
- It takes a callback function as its first argument.
- It acts just like componentDidMount and componentDidUpdate of React Class Components.
- It takes conditions as its second argument which will execute the effect if the condition is satisfied.

Print your Data





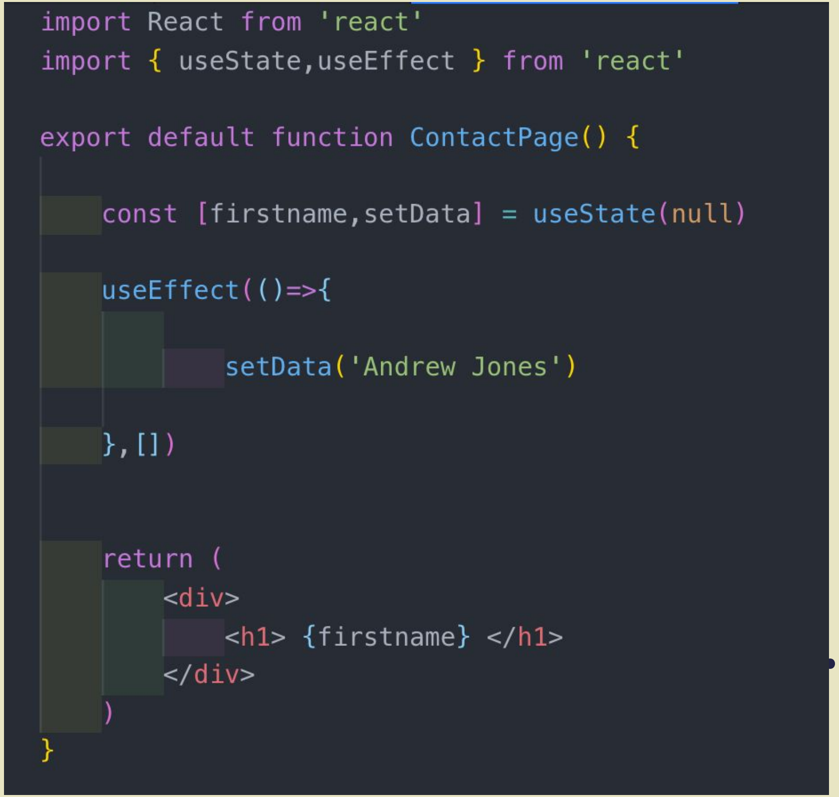


05

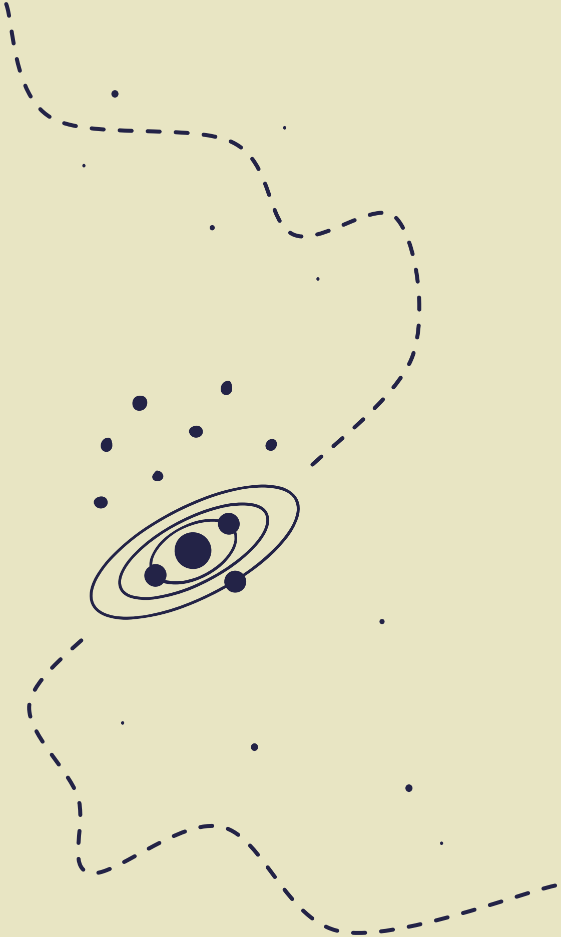
Step 5

Print your Component State Variables

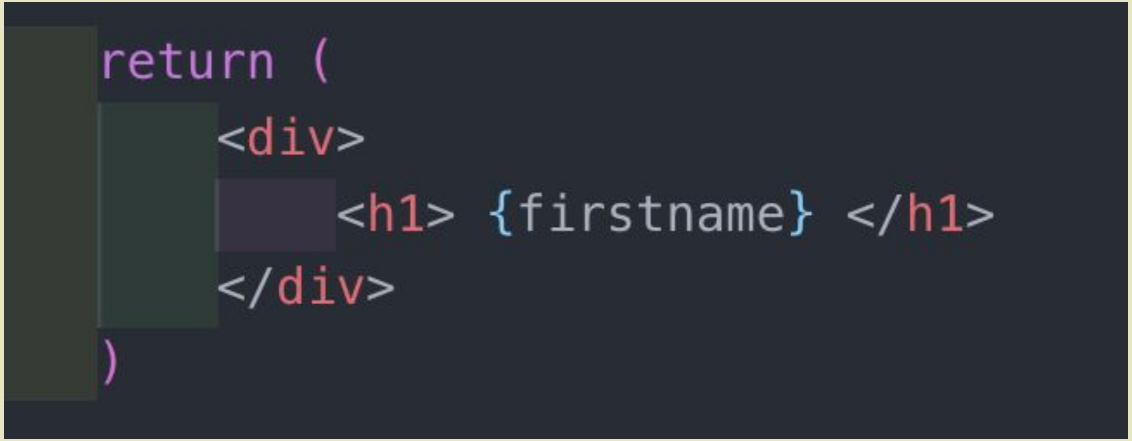
Step 5: Printing your Data



```
import React from 'react'
import { useState,useEffect } from 'react'

export default function ContactPage() {

    const [firstname,setData] = useState(null)

    useEffect(()=>{
            setData('Andrew Jones')

    },[])


    return (
        <div>
            <h1> {firstname} </h1>
        </div>
    )
}
```



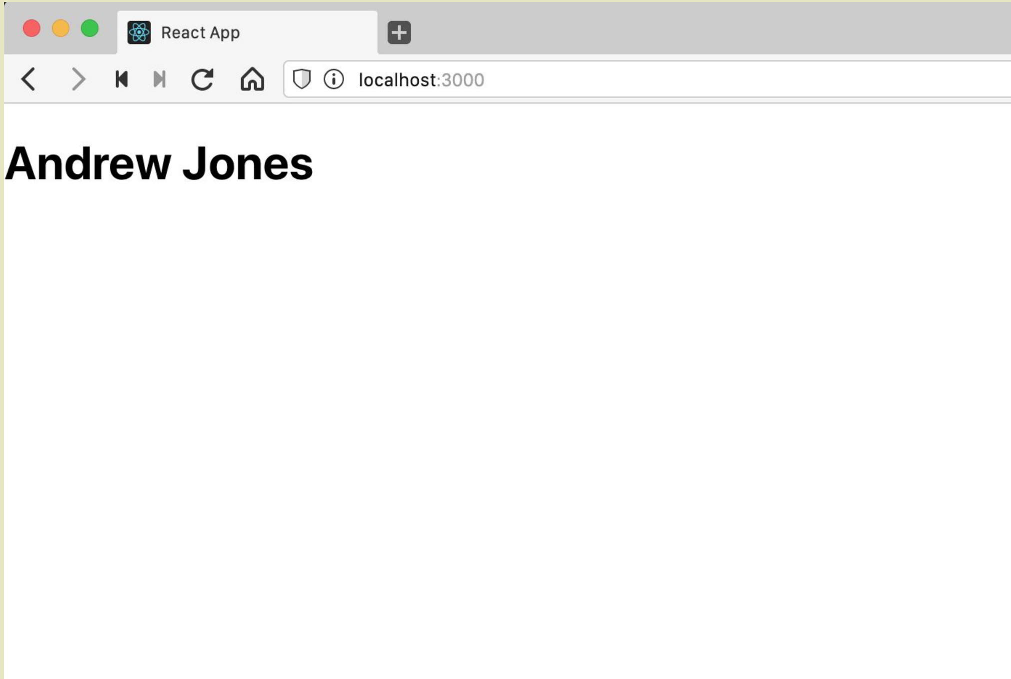
Code Explained



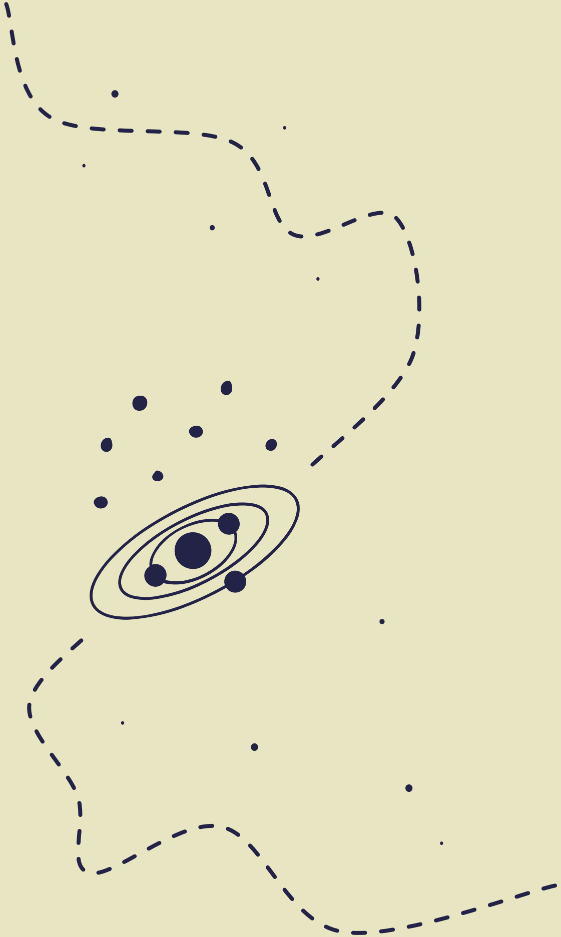
```
return (
    <div>
        <h1> {firstname} </h1>
    </div>
)
```

- return() method of your component returns HTML Markup.
- In order to print dynamic data wrap the state variables inside {} braces.
- The Values of the variables will now Print.

Output:

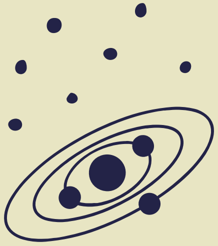


Andrew Jones



Additional Reading: Types of Components in React

- Class Components
- Functional Components
- Dumb Components
- Smart Components
- Presentational Components
- Container Components



Queries:

Discord - DM @mithileshtarkar

Email - mithilesh.tarkar@gmail.com

Thank You :)