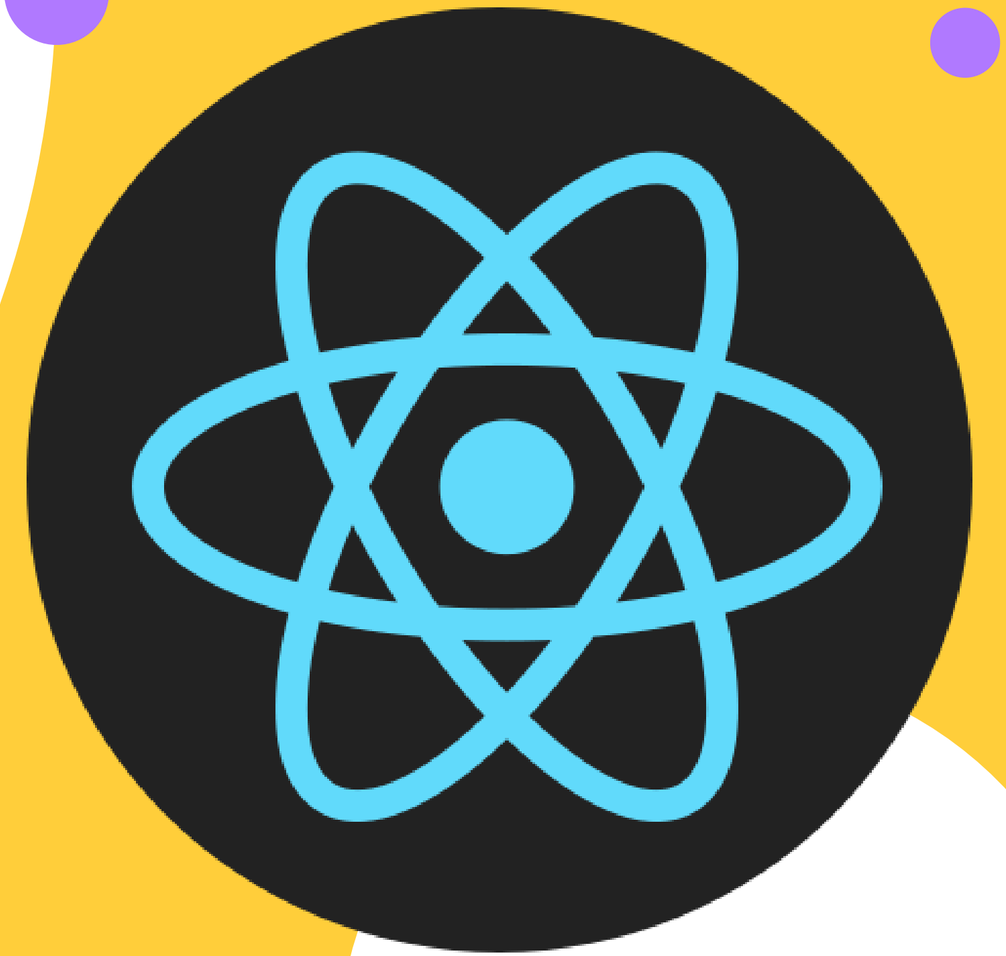
Introduction with React JS

React

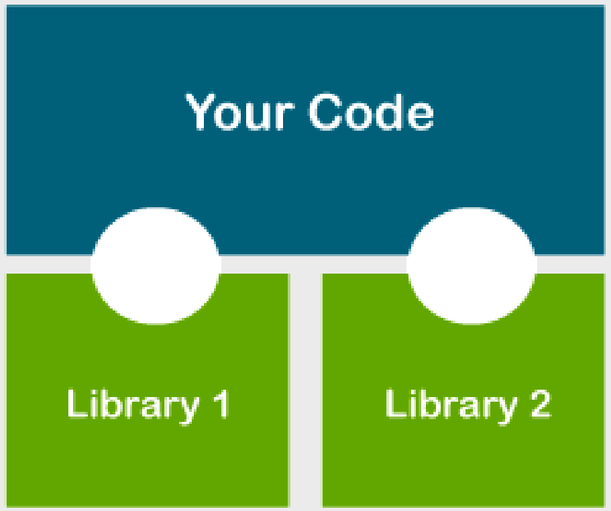


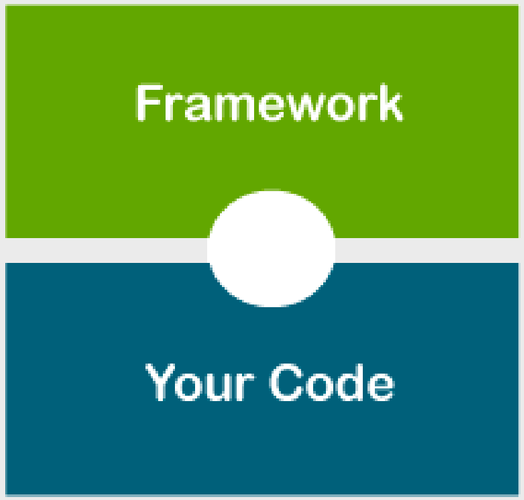
Out Line

- What is React
- Rendering
- JSX
- Component
- Export/Import
- State
- Hook
- Props
- Handling Events
- React router

React is a JavaScript library that is used to create user interfaces (UI). React's most obvious feature is that it writes complex code in separate chunks, each of which can perform separate tasks. freely and can be reused in other UI parts or application screens which was developed by Facebook (META).

Library vs Framework





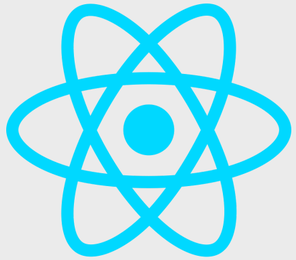
- Library: Call us to get the job done.
- Framework: You don’t call us, we’ll call you.

Framework

NEXT.JS

Express

Library



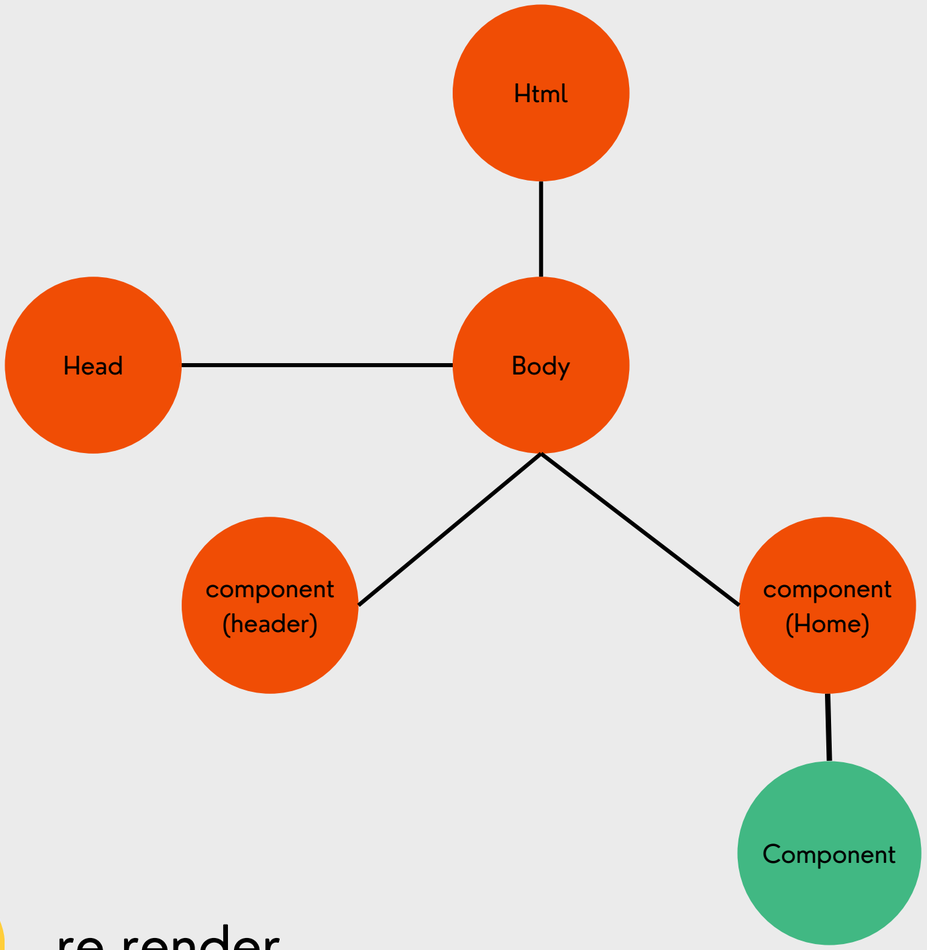
Virtual DOM

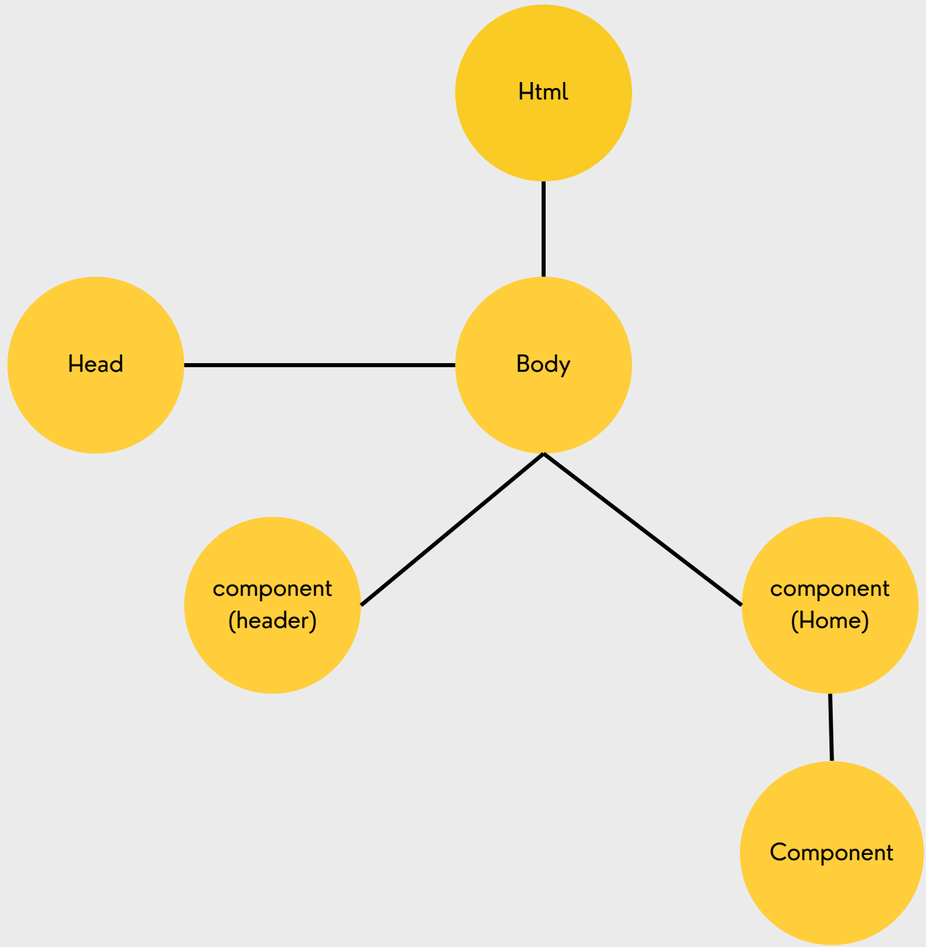
A Virtual Dom works like an HTML Dom in that it copies the real DOM (Real DOM) and stores it in memory. If there are changes in the component Only components that have changed will be updated.

There is no need to update the whole DOM page. cause it to work quickly

**If used (Normal DOM will refresh the whole page to update the changed

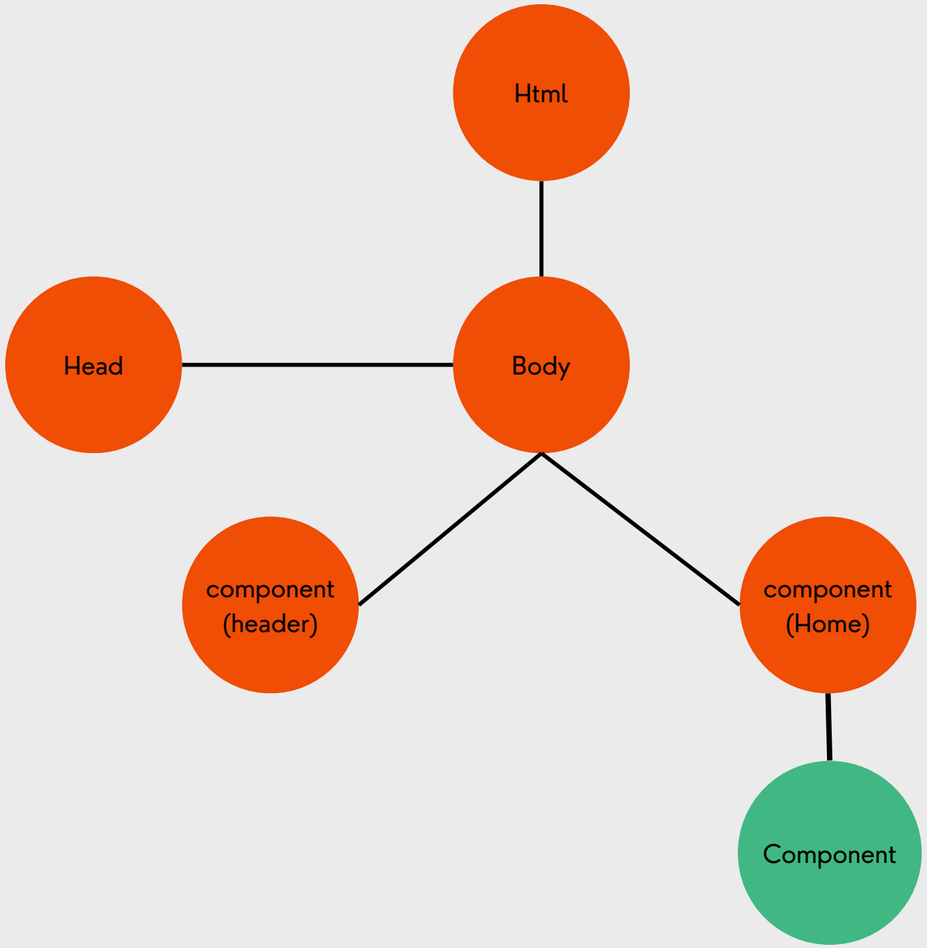
Normal DOM

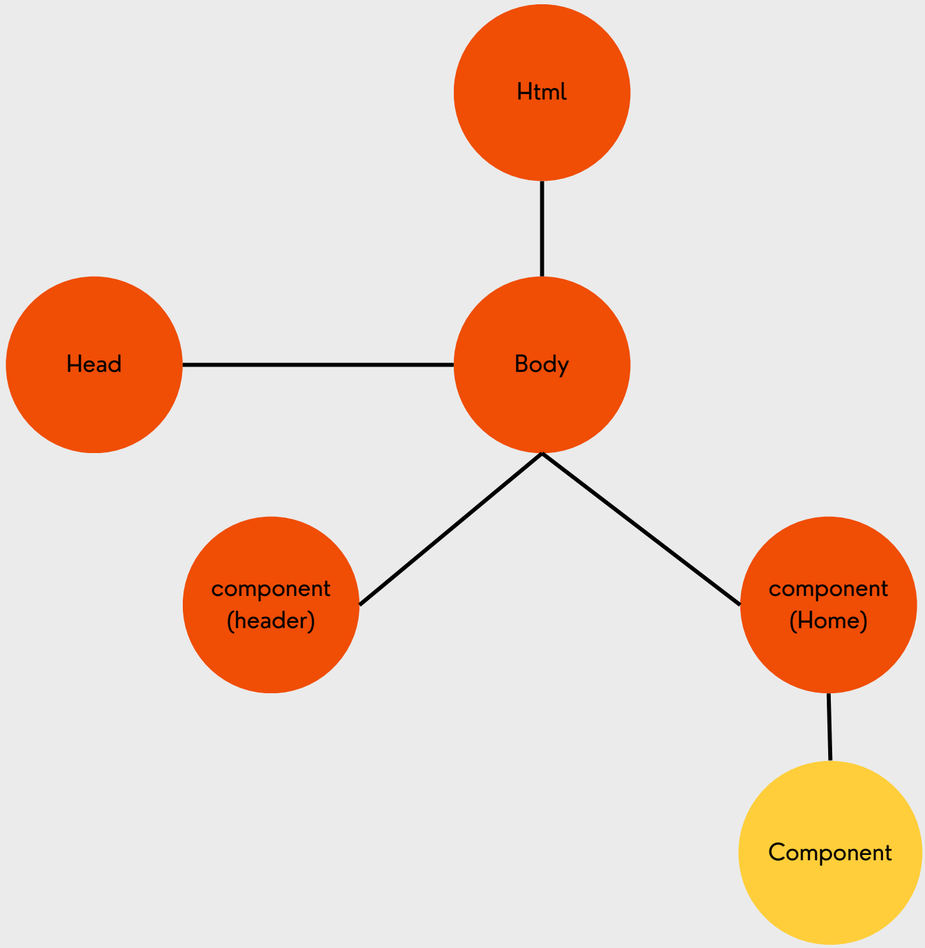




re render

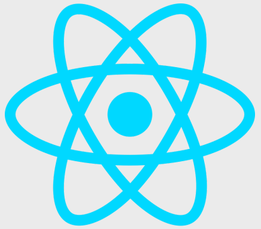
Visual DOM in React





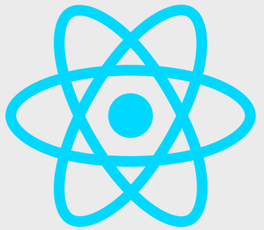
re render

How to Create React project.



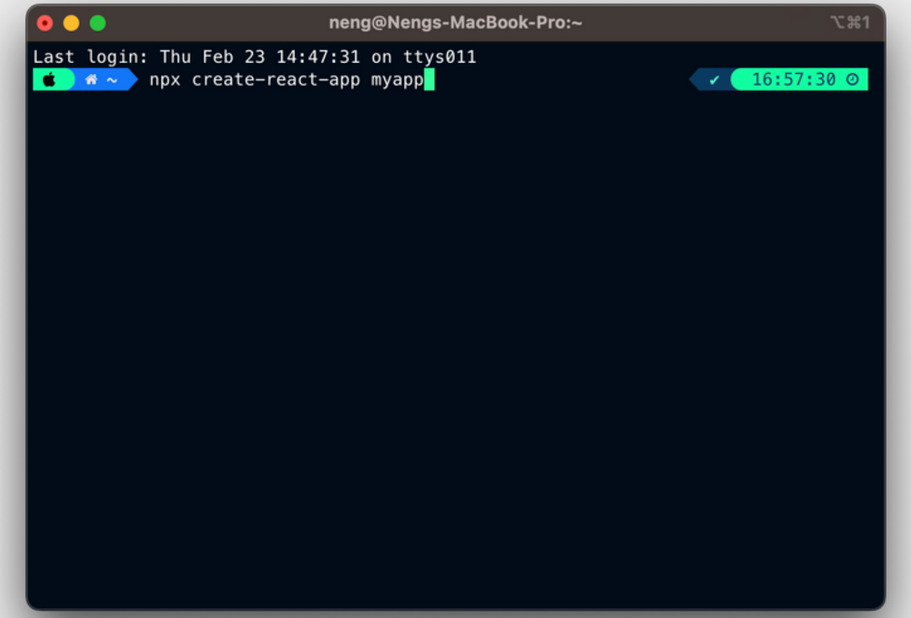
Original way

I will teach you this way



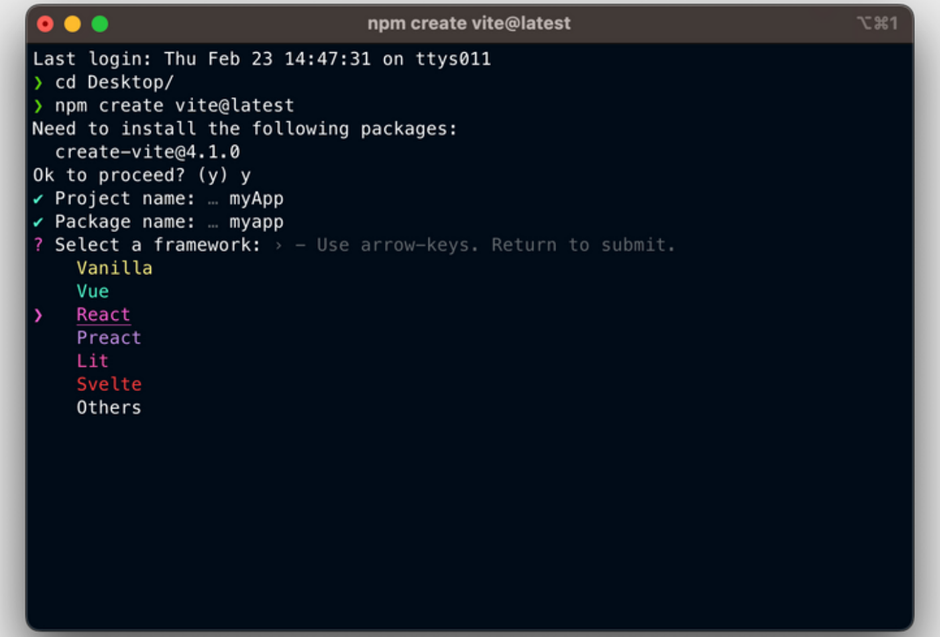
new!

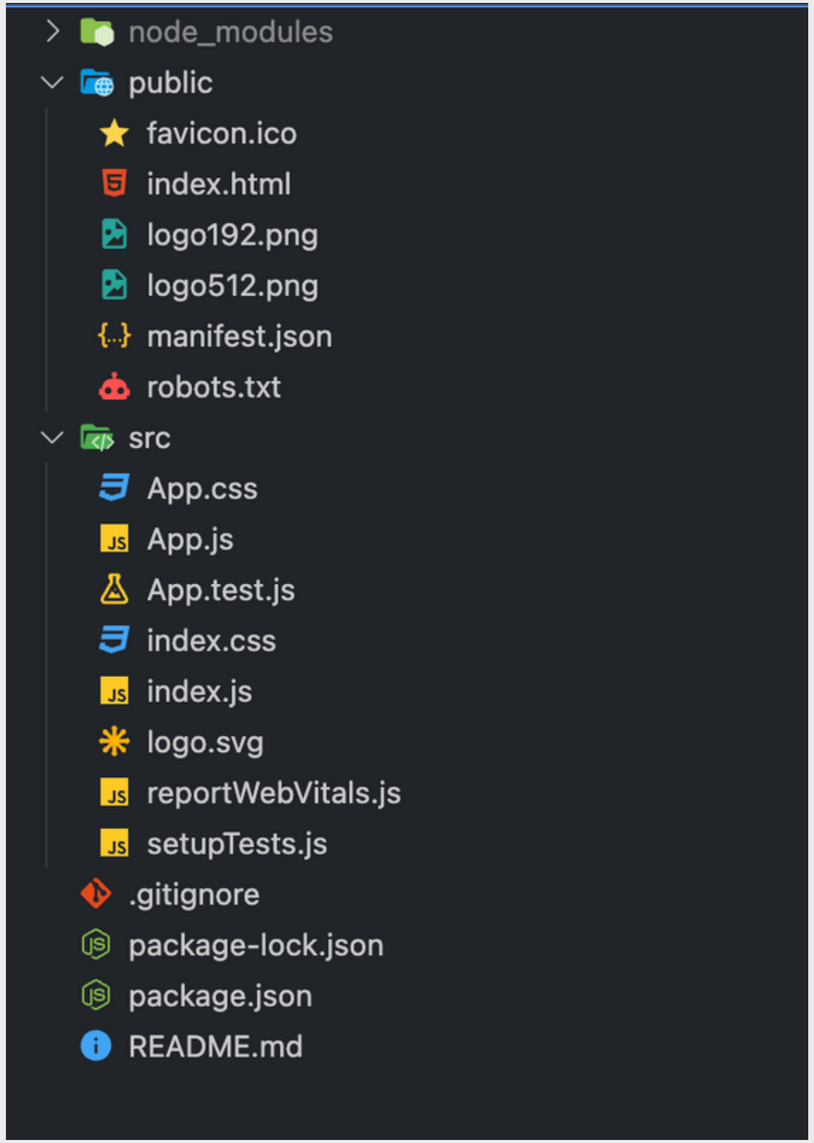
Fast!



environment require !

node js + npm





the directory (folder) & file you must to know

/public : it's a directory to keep html file (index.html)
and the images will keep in this dir

/src : it's a directory to keep javascript or Typescript file
All file that you develop keep in this dir

package.json :this file include information of project and
the dependency that you install and script to run command

Rendering

What is Rendering ?

Rendering is to render the React element we created in the browser using ReactDOM.

```jsx
import { ReactDOM } from 'react';
function Hello(props) {
return <h1>Hello {props.name}</h1>;
}
const element = <Hello name="Poopoo" />; // <h1>Hello Poopoo</h1>
const rootElement = document.getElementById("root");
ReactDOM.render(element, rootElement);
```

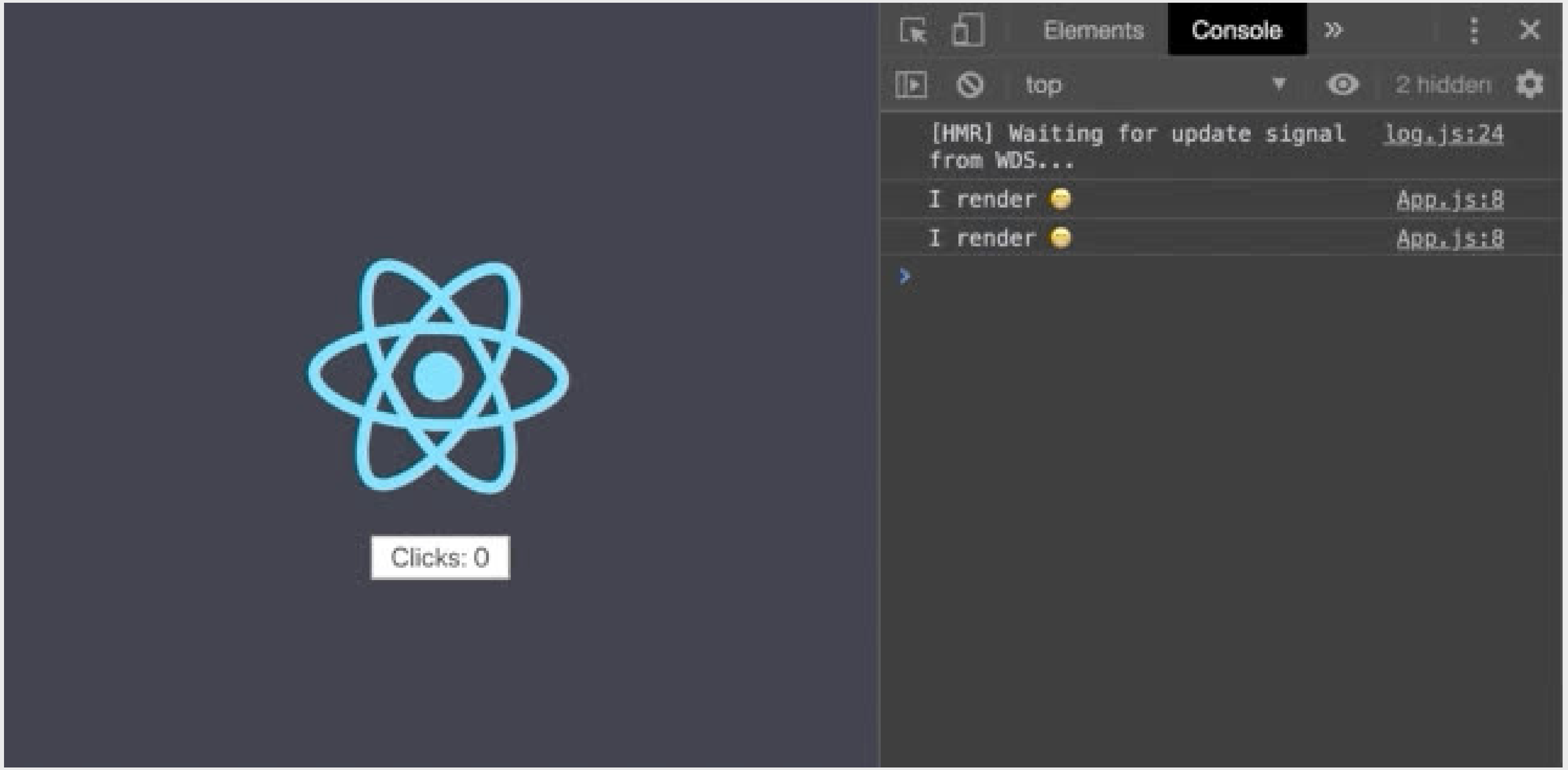
- element is the React element that we will render.
- The rootElement is an HTML element of choice that acts as a container that allows ReactDOM to handle everything inside and render the element.

Difference between React and pure HTML?

The difference between React and pure HTML is that React has its own Virtual DOM .
The virtual DOM to process different components within a React web page converts only the components that have changed, not the entire page. **This will result in faster rendering of the website.**

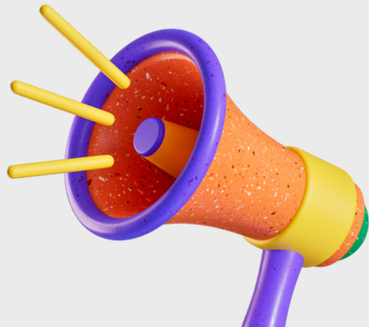
Re-rendering

Happens when a rendered React element has a props or state that has changed from the previous render.
That React element will be rendered again with a new set of props and state values.


Clicks: 0
Elements
Console
top
2 hidden
[HMR] Waiting for update signal from WDS...
log.js:24
I render
App.js:8
I render
App.js:8

What is JSX ?

As an extension of JavaScript, we can create elements by writing HTML tags, and can also use JavaScript's capabilities to perform logic.



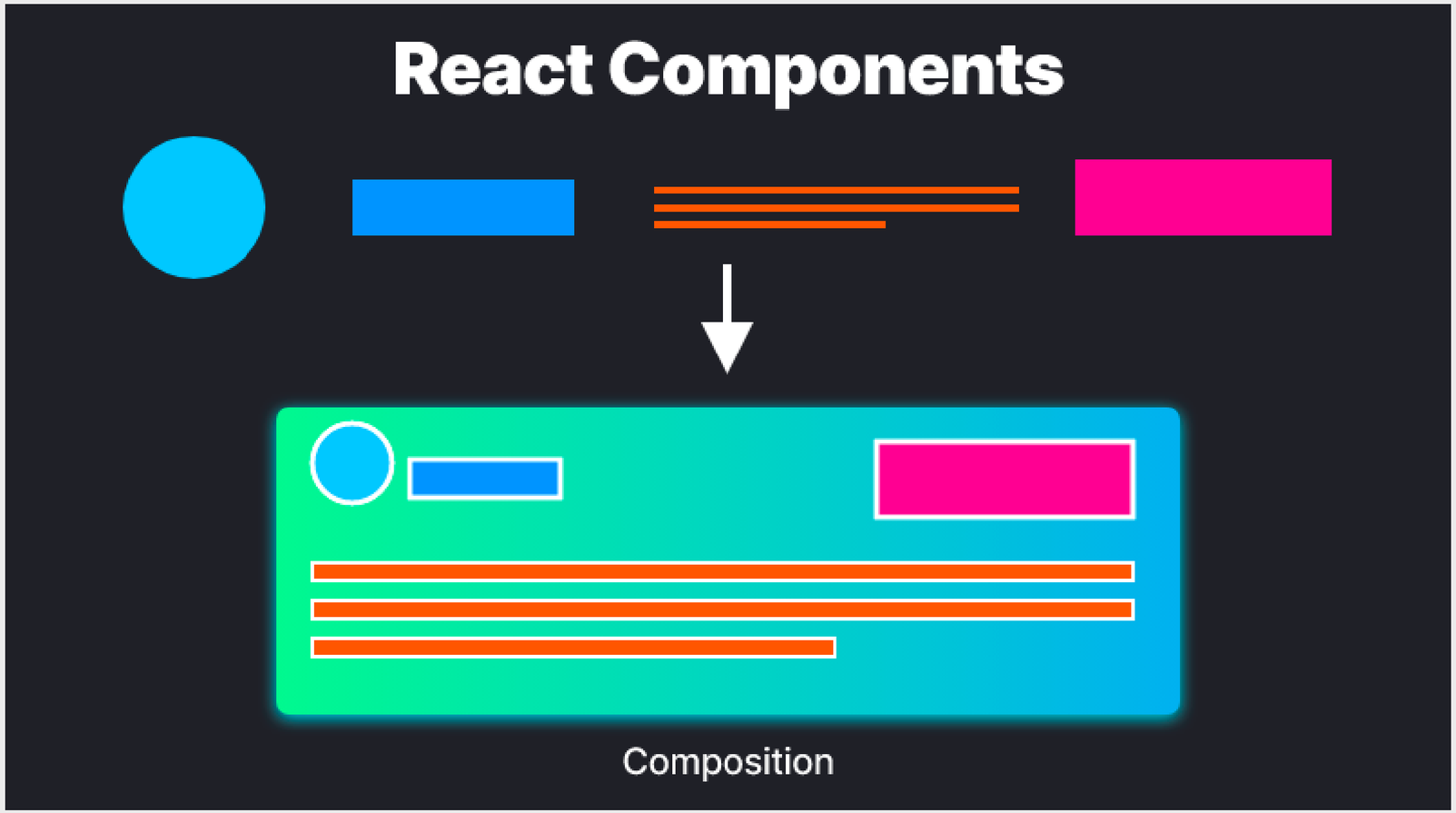
```jsx
const name = "Stthi";
const element = <h1>Hello {name}</h1>;
```

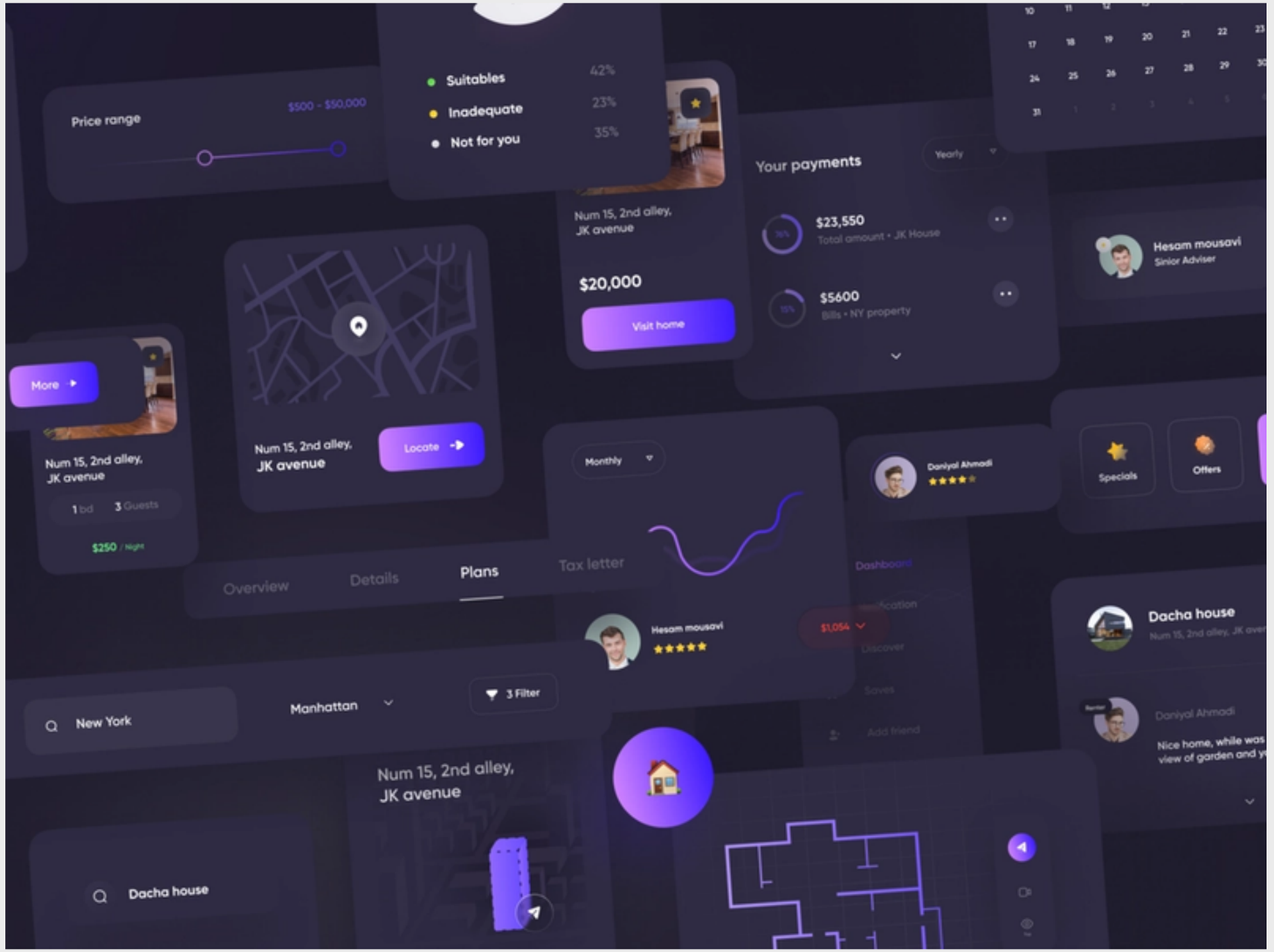
JSX allow use html tags in the JS Syntax

```jsx
const name = "stthi";
const element = React.createElement("h1", null, "Hello ", name);
```

React will tranform html tag to the htmlElement with React.createElement

Component


React Components
Composition


Price range
$500 - $50,000
Suitables
42%
Inadequate
23%
Not for you
35%
Num 15, 2nd alley,
JK avenue
$20,000
Visit home
Your payments
Yearly
$23,550
Total amount • JK House
$5600
Bills • NY property
Hesam mousavi
Sinior Adviser
More
Num 15, 2nd alley,
JK avenue
1 bd
3 Guests
$250 / Night
Num 15, 2nd alley,
JK avenue
Locate
Monthly
Overview
Details
Plans
Tax letter
Hesam mousavi
Daniyal Ahmadi
Specials
Offers
Dashboard
Discover
Saves
Add friend
$1,054
New York
Manhattan
3 Filter
Num 15, 2nd alley,
JK avenue
Dacha house
Dacha house
Num 15, 2nd alley, JK aven
Daniyal Ahmadi
Nice home, while was
view of garden and y

What is Component ?

Components allow us to break down the user interface into smaller, self-organizing parts. and we can reuse it
There are two types of components in React:

1.Class component

2.Function component

Class components

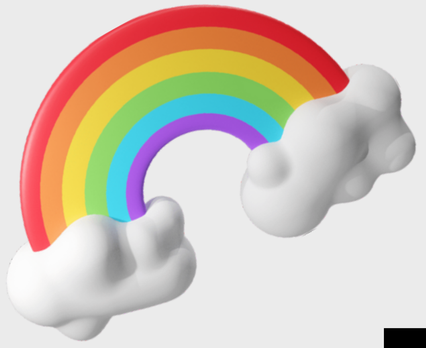
This is the original way of coding react, which later has react hooks, so function components can be written as class components as well.

```
1 class Hello extends React.Component{
2   render(){
3     return(
4       <h1>Hello React</h1>
5     );
6   }
7 }
8 ReactDOM.render(<Hello/>,document.getElementById('root'));
```

Function components

It is a component written with a function that must return a value as a React element that will be rendered by React.

```
1 function Hello() {
2 return <h1>Hello React</h1>;
3 }
4 export default Hello ;
```



Export / Import

Export / Import

is a statement used to share components between JavaScript files.
We call JavaScript files with export/import **Module.

There are 2 types of commonly used forms:

1. **Named export/import**
2. **Default export/import**

Named export/import

- import must use the same name as export
- import must be inside { }

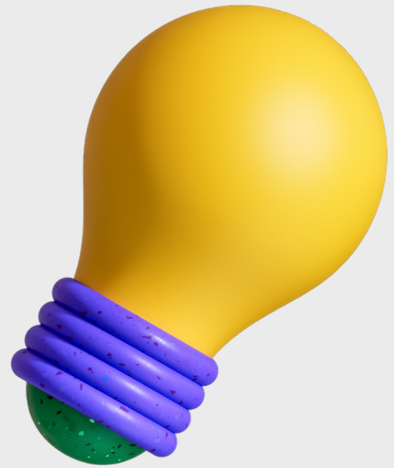
```
1 // fileA.js
2 export const name = "stthi";
3 export const age = 15;
4 // fileB.js
5 import { name, age } from "./fileA.js";
```

Default export/import

- You can export default only once for each module, importing it with any name.

```
// fileA.js
const name = "stthi";
export default name;
// fileB.js
import nameOrSomething from "./fileA.js";
```

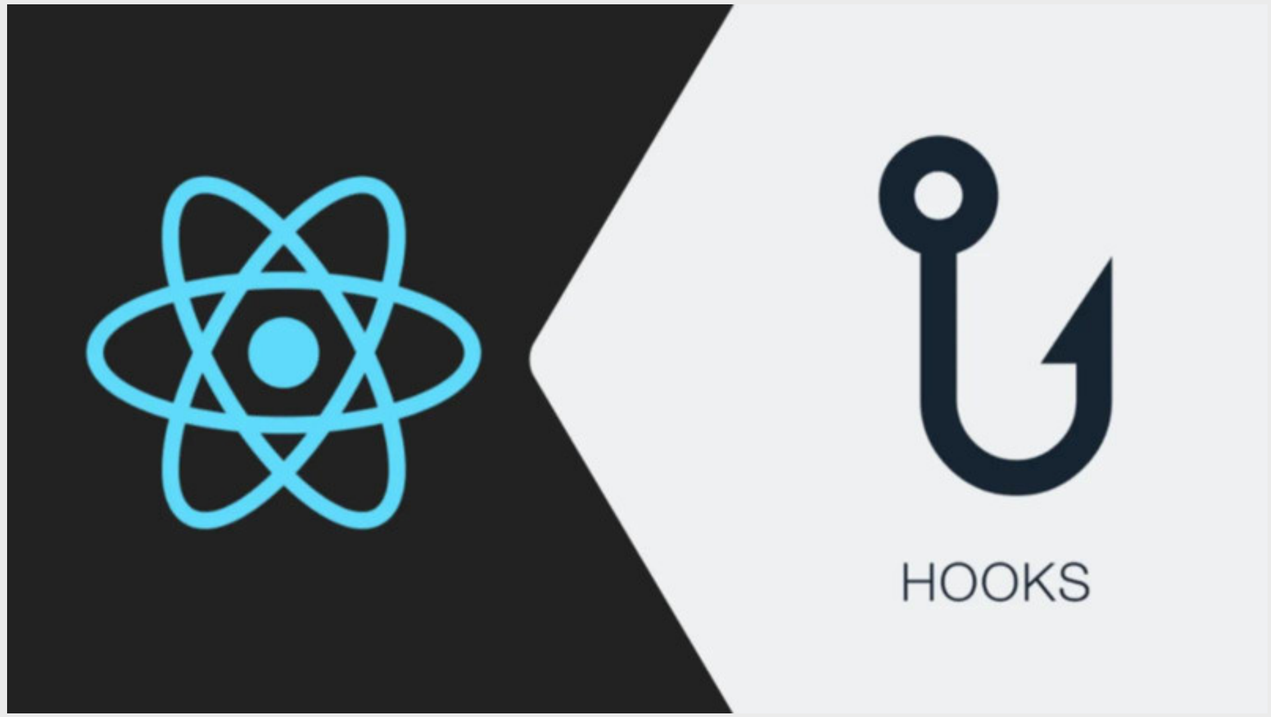
State



What is State ?

State is a variable that stores data inside a component, similar to props, but using props, the data cannot be changed, but the state can. During app execution, it uses State, which is the original format written internally.

Hooks



What is Hook ?

Hook, for calls to add state or other React capabilities to function components. There are two hooks that we often see:

1.useState

2.useEffect

useState

Used to declare state variables to function component.
In 1 component, you can declare as many times as you like.
*The values of state variables are stored with React.

Terms of Use

- It must only be called within a function component.
- must be called at the top of the function component
- The call order must be the same for every render (not in a condition or loop).

The value that useState accepts.

- [Optional] Default state variable (defaults to undefined if omitted).

The value that useState will return.

An array with two members is

1. The current value of the state variable.
2. function to change the state variable

useState will run when ?

When the component first renders : useState declares a state variable and returns it as an array with default members that we pass along with a function to change the state variable.
render next time

Example

```
1  import React, { useState } from 'react';
2  function Counter() {
3  const [count, setCount] = useState(0);
4  return (
5  <div>
6  You have {count} cookies
7  <br />
8  <button onClick={() => setCount(count + 1)}>
9  Steal a cookie
10 </button>
11 </div>
12 )
13 }
14 const [count, setCount] = useState(0);
```

In our example, we declare a state variable called count with a default value of 0 and use array destructuring. to grab the value from the array that useState returns for us

Example

```
return (
    <div>
    You have {count} cookies
    <br />
    <button onClick={() => setCount(count + 1)}>
    Steal a cookie
    </button>
    </div>
    )
```

Then we display the count and have a button that, when clicked, increments the count by 1 via calling setCount .

**Changing the state variable will cause the component to re-render to apply the latest state.

useEffect

It is used to deal with side effects of function components such as displaying data, updating the title of a website page, etc.

Terms of Use

- It must only be called within a function component.
- must be called at the top of the function component
- The call order must be the same for every render (not in a condition or loop).

The value that useEffect can take.

1. [required] effect (function to run after render)

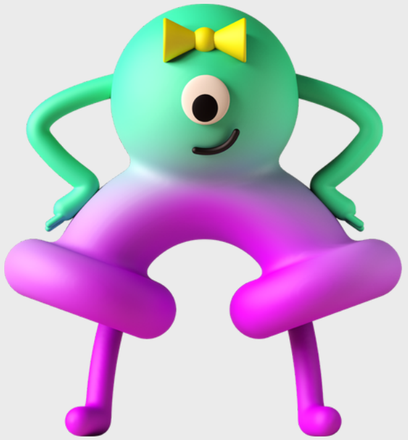
2. [Optional] dependency array

The effect will only be executed after rendering is complete.

useEffect will run when ?

When the component first renders : Effect 1 time
Subsequent renders : Based on array dependencies.
1. If an array is included with its members : The effect is obtained when there is any member value that must be remembered from the last render first.
2. If it's an empty array: the rest of the effects work as well.

Props



What is Props ?

Props is a variable that can be passed into Components through attribute assignment, resulting in each component being able to receive value from the outside to work.

```
function Hello(props) {
return <h1>Hello {props.name}</h1>;
}
const element = <Hello name= "stthi" />; // <h1>Hello sthhi</h1>
```

**The function component takes props as one argument (parameter names are commonly used as props, but other names can be used).

Props.children

Just like HTML, the components we create can have opening and closing tags. props.children (including JavaScript expressions)

```
function Hello(props) {
    return <h1>Hello {props.children}</h1>;
    }
    const element = <Hello>Doodoo</Hello>; // <h1>Hello Doodoo</h1>
```

Handling Events

Handling events in React

Unlike in HTML, The name of the event attribute is case-sensitive. and in the form of camelCase e.g. onClick , onChange , onMouseOver

```
1
2  function showMessage() {
3      alert("A Message");
4      }
5      <button onClick={showMessage}>
6      Show A Message
7      </button>
```

Validation Form

```jsx
function MyForm() {
  const [name,setName] = useState("");

  //handleForm
  function handleChange(event){
    setName(
      event.target.value
    )
    console.log(name);
  }

  //handleEvent

  function handleSubmit(event){
    event.preventDefault();
  }

  return (
    <div className="App">
      <form onSubmit={handleSubmit}>
        <label>Name</label>
        <input type="text" onChange={(event)=>handleChange(event)} /><br></br>
        <input type="submit"></input>
      </form>
    </div>
  );
}

export default MyForm;
```

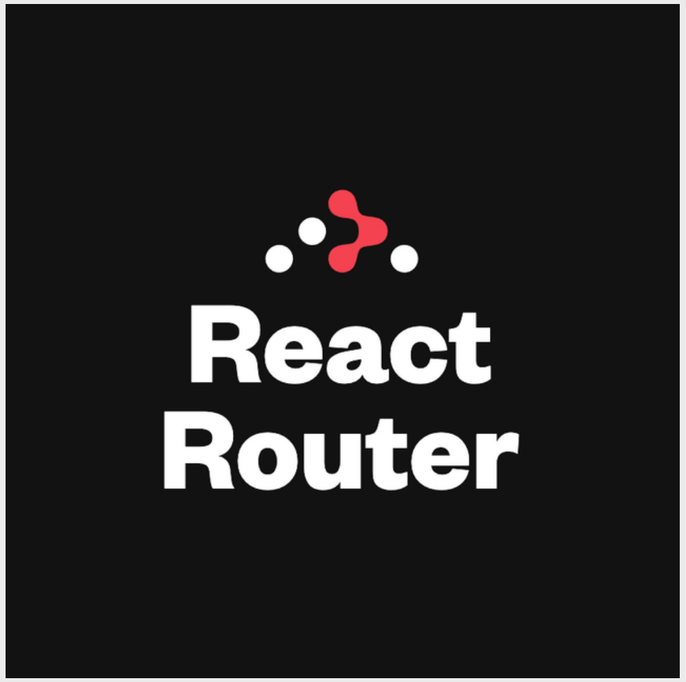
OnChange : event Attribute to handle input

event.preventDefault() : is a function to prevent re-render webpage when you click the submit form

React Router

Create React App doesn't include page routing.

React Router is the most popular solution.

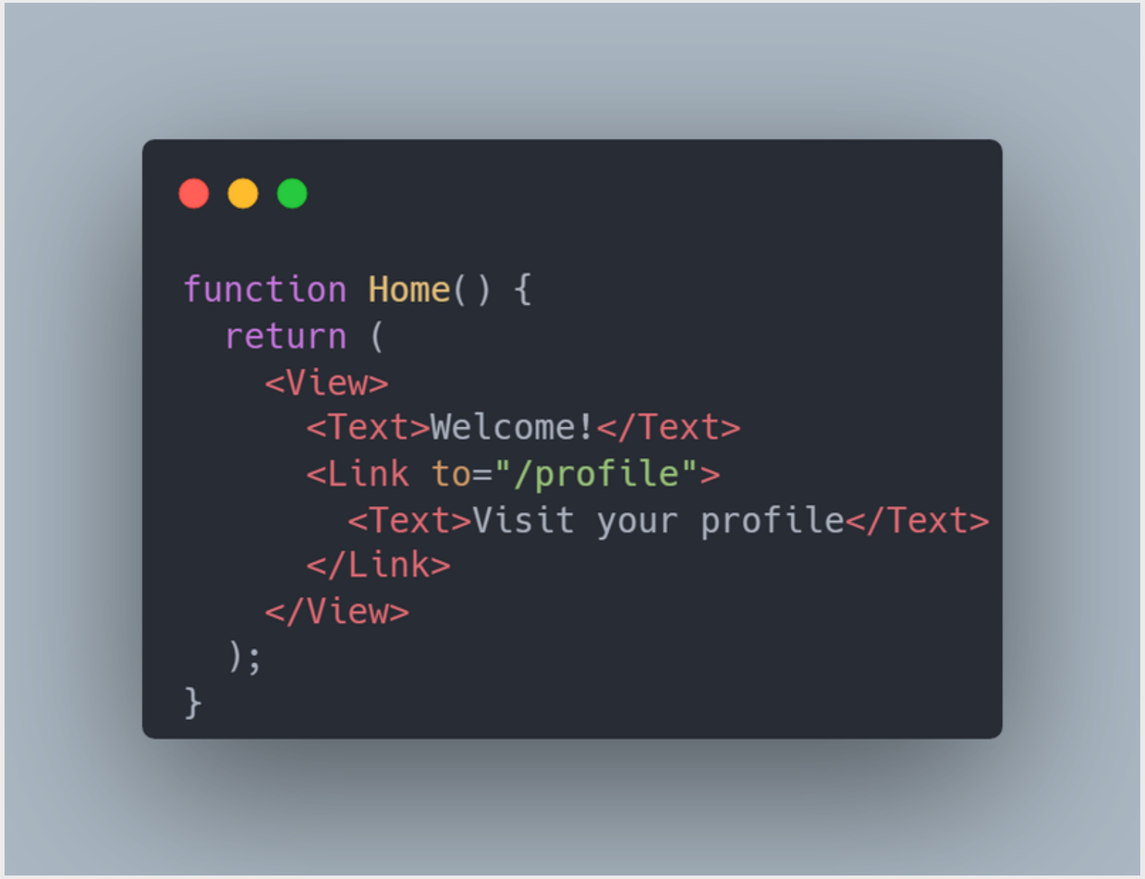


```
export default function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Layout />}>
          <Route index element={<Home />} />
          <Route path="blogs" element={<Blogs />} />
          <Route path="contact" element={<Contact />} />
          <Route path="*" element={<NoPage />} />
        </Route>
      </Routes>
    </BrowserRouter>
  );
}
```

In The main javascript file

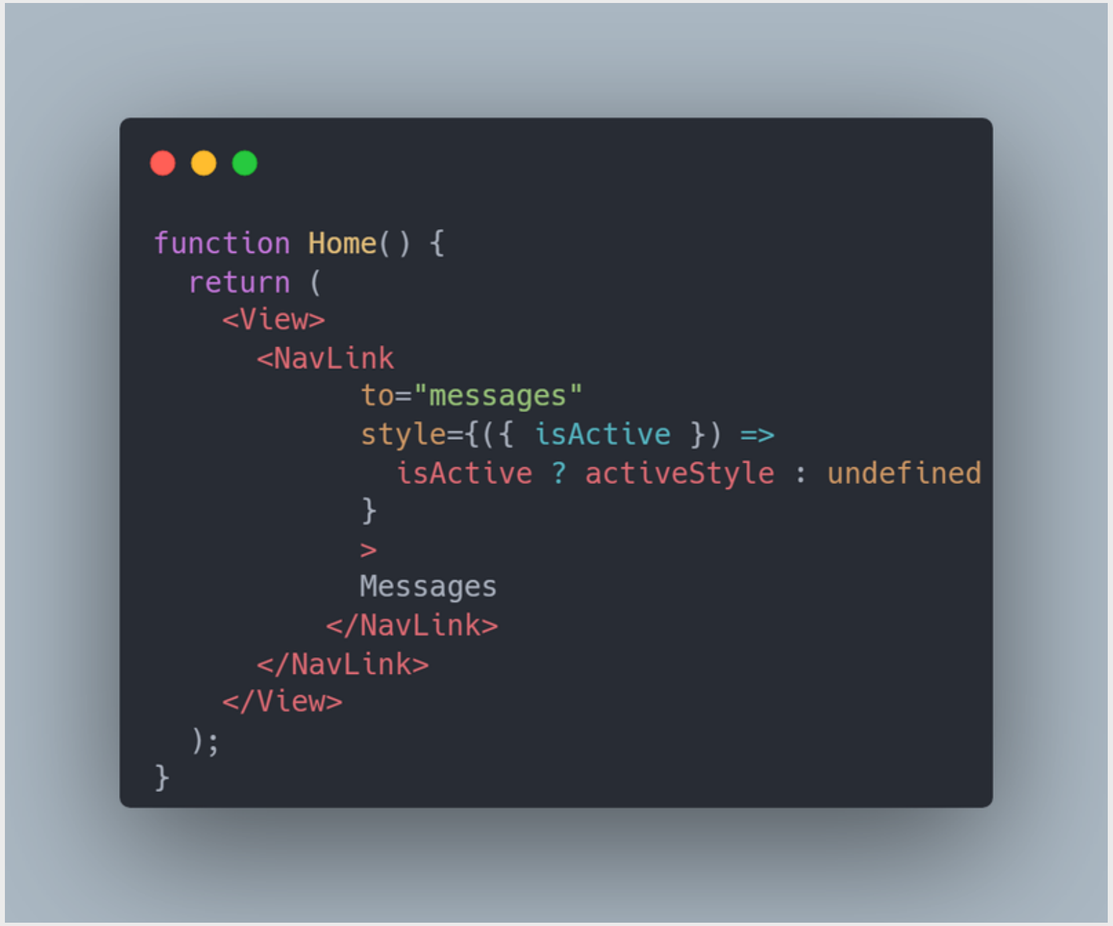
BrowserRouter : stores the current location in the browser's address bar using clean URLs and navigates using the browser's built-in history stack.

<Router> is the low-level interface that is shared by all router components (like <BrowserRouter> and <StaticRouter>). In terms of React, <Router> is a context provider that supplies routing information to the rest of the app.



```
function Home() {
  return (
    <View>
      <Text>Welcome!</Text>
      <Link to="/profile">
        <Text>Visit your profile</Text>
      </Link>
    </View>
  );
}
```

A <Link> is an element that lets the user navigate to another view by tapping it, similar to how <a> elements work in a web app.



```
function Home() {
  return (
    <View>
      <NavLink
            to="messages"
            style={({ isActive }) =>
              isActive ? activeStyle : undefined
            }
            >
            Messages
          </NavLink>
      </NavLink>
    </View>
  );
}
```

A <NavLink> is a special kind of <Link> that knows whether or not it is "active". This is useful when building a navigation menu such as a breadcrumb or a set of tabs where you'd like to show which of them is currently selected. It also provides useful context for assistive technology like screen readers.