

Ep 03: Laying the foundation

npm run start is equivalent to npm start.
npm run build (builds code for production)

In package.json, inside scripts → add two new script for development & production.

```
"scripts": {  
  "start": "parcel index.html" ⇒ (npm start)  
  "build": "parcel build index.html" ⇒ (npm run build)  
}
```

Whenever you write npm start it will go the start script inside package.json file and execute the start command
i.e., parcel index.html
Similarly for build command npm run build

In JS we have **DOM** Elements, Similarly in React we have **React Element**
React Elements are equivalent to DOM Elements in Browser

React.createElement ⇒ Object ⇒ HTML Element

ReactElement is
an object

React.createElement
is an object

while render this
react.createElement i.e., object / React Element
is converted into
HTML Element

JSX: JSX is a Javascript Syntax which is easier to Create React Elements

- JSX & React both are different. JSX makes Write our Code Simpler.
- JSX is a Convention which tries to merge JS & HTML together in one file
- ↳ JSX is a HTML like Syntax. JSX is not HTML in JS.
- JSX is Only Syntax, React Element is a object

JSX & HTML both are different. There is a Slight difference betn both of them

In HTML while assigning class name to HTML element we use `class=""` attribute

But In JSX while assigning class name to HTML element we use `className=""` attribute

↳ becoz in JSX class is a reserved keyword

↳ If you want assign attribute to a JSX Element use Camel Case

(1) Single line & Multiple line.

While written Jsx code is Single, there is no need of parenthesis ()

But, While written JSX code in multiple line, then you need to wrap it up in Parenthesis ()

Because Label needs to understand where the JSX Code is started & Where it is ended.

React Component: Everything in React App is a Component.

Two Types of Components

→ class Based Component (old way of writing Component)

- use JS classes to create React Component

functional Component (new way of writing Component)

- uses JS function's to create React Components
- It is Just a normal JS fn
- Always name it with first letter Capital

- Functional Component is a normal JS arrow fn, which returns some JSX elem which returns React Element.

Both are Same

```
const HeadComponent = () => {
  return <h1> Hi </h1>
};
```

```
const HeadComponent = () => <h1> Hi </h1>;
```

```
const HeadComp = () => {
  return <h1> Hi </h1>;
};
```

Both are Same

```
const HeadComp = () => (
  <h1> Hi </h1>
);
```


Component Composition: A Component inside another Component.
- Composing two Component into one another.

- { } in JSX, anywhere you write { } in JSX, Inside these { } you can run any piece of JS Code.
- You can write any JS Code in { }

Very Powerful,

Writing any JS Code in JSX is very Powerful { }

- You put normal React Element in JSX Component using { }

↓
As we React Element
is a JS code, hence
we can use it also

JSX will take care of Injection Attacks.

- It prevents from Cross Side Scripting Attacks. } JSX will not blindly run the code data
it will sanitize your data. Prevent you from Attacks

① <Component></Component>

② <Component />

} Both are Same → { Component() } ③
All are Same