

## # React JS

It is most popular & best way for building user interface with JS (JavaScript)

- It is a front end building framework
- It makes things very easy & neat

React prefer to use a single html file to make & apply things on your website

→ It makes website interactive ~

when you change few things so react map them & only reflect changes in that area that help us to traverse without refreshing

⇒ we can traverse content without refreshing page it is due to the react JS

⇒ React is most popular & most demanded framework in current time

### ④ Code sandbox

To stub app to work with react JS

⑤ We have to add dependencies in sandbox

for using React on screen for that we use react.render function which has 3 parameters

⇒ React-D

ReactDom.render(what to show, where to show,  
callback function())

(this is optional)

ReactDom.render(<h1>Hello world</h1>)  
, document.getElementById("top");

↑  
It finds that element & put that h1 tag  
inside that id tag

By a simple Javascript without having  
added path for html file you can fetch as  
well as manipulate data in html

& you don't have need to write back double  
quotes or else & write tag inside them

) ReactDom = require('react-dom')

ReactDom.render(<h1>Hello</h1>,

document.getElementById("root");

↑  
only this → By this you can write html  
inside js

This all happens with use of module of  
Javascript called JSX

⇒ If you not include react in file you  
Babel is component get error  
which convert any JS in  
such a manner than any browser  
can able to understand it

so with babel you can explore how things get mapped from inside form high level JSX code to plain low level JS code

④ Bible take low no. of lines so code is short & more for same functional

⑤ Importing libraries without doing const variable declaration By using Import from keyword

Import React from "react";  
import ReactDOM from "react-dom";

& you can use code exactly same which you used inside the normal low level JS

⑥ Problem with render method

You can put only one tag at a time inside tags of render

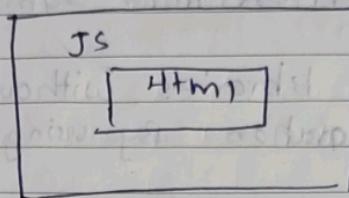
Ex:

ReactDOM.render(<H1>Hello</H1> <H1>World</H1>  
~~(H1)~~ in jlocation)  $\Rightarrow$  this will throw  
{std::bad\_alloc} error

so If you want to add multiple number of code of tags to add them at end put them all in under same parent like div, p, section whatever but should be in same end tag start & end tag

this is calling wrapping tags under `div`  
or `p` or anything else

Basically in JSX



We can write whole HTML in JS so  
and we can also go further we can  
use Javascript in that inserted HTML also

Ex:-

Say your print `Hello world`  
`xyz`

But we want `xyz` is a variable so  
at that pace of time JSX will come to  
play

So as we use `(a + b)` in `(f + 1)`  
`print(f + {a + b} = {a + b})`

where `a` & `b` values are variables so we are also  
able to print them also

So say

## Read function name passing as parameter

Page No.	
Date	

```
import react from "react"
```

```
import ReactDOM from "react-dom"
```

name = "Ghansham"

so ReactDOM.render(<H1>Hello name</H1>, location)  
so code will be

```
ReactDOM.render(<H1>Hello {name}</H1>, location)  
that's it
```

You can put one or more values inside same tag & this will not cause you any trouble.

so In tag

↓ Javascript

```
<H1> Hello {name} </H1>
```

↓ HTML

You also able to perform operations on numbers & all with use of the ~~HTML~~ javascript in html  
let a=10 ; let b=10  
{a+b} ⇒ output 20 will be printed

You also call functions inside the the Javascript expression

⇒ You can use any js expression but you can't use statements like control flow statements.

Expressions vs statements  
just assigning value & such related stuff  
(something which tell computer to do something)

⇒ We also can use template literal inside html as Javascript expression

fname = "ghansham" methods  
lname = "salunkhe" = functions

so { \${fname} \${lname} }

thus we can able to use fname & lname inside Javascript expression & it is called as template literals

^ ⇒ a key above tab.

## # JSX attributes & styling react Elements

In this topic we will add design & CSS stuff inside our Page with JSX

So as we know that we can pass elements inside tags which are classes so here but here is a catch here you are not able to use name as per CSS convention you have to use names per JS convention

⇒ <H1>class = "abc" abc </H1>

In CSS => class ⇒ It will give error

In JS ⇒ className

will <H1> className = "abc" abc </H1>

100 & 100px

Page No.	
Date	

First

⇒ So ~~second~~ one will not provide any conclusion  
JSX supports naming conventions as per JS

In html file & in script the type of  
file should be

type = plain/SSX so it will not give  
any kind of error.

⇒ So html not provide any camel casing so  
If you want to use such elements in  
JSX you have to write them into camelcase  
form  
⇒ without camelcasing they not works.

In html ⇒ contenteditable  
while using in JSX ⇒ contentEditable.

⇒ you can target html directly from CSS or  
manipulate them with JSX & CSS (class)

You have to be careful when you're using  
selfish closing tags they may work well in  
html but in JSX they come with trouble

Ex: ⇒ Img Br style initial end 02

use them like 1> ⇒ closing 02

when:  1> close of  
    <br /> Br tag closure 1> Img tag

## watch `exit` method vs Properties In programming

If you not did it so they you will have  
to handle errors like unmatched closure  
error

⇒ you can use Javascript variable / object  
as argument to attribute of tag

```
Img1 = " /Photos /xyz.png"
```

```
class1 = "Image"
```

```
<Img src={Img1} class={class1}>
```

such like above we can use tags & variables

`alt` in `img` ⇒ text when if image not rendered

## #Inline styling

In `html` you comfortably use `style` like

```
style = "color: Red; background-color: Blue"
```

such but in `JSX` it will go toward  
an error

so here inline styling needs a javascript  
object but above we are sending string  
so that why it will pop error

Object is one which works like `key: value`  
pair

Inline-styling may for  
very helpful many no. of times

Page No.	
Date	

Go for normal variable in JSX we can  
put value in {} But for object  
we have use double curly brackets

`<h1 style={{ color: 'red' }}> Hello world </h1>`  
such like that

JSX only support Camel Case

So you have convert HTML's normal & CSS's  
Kabab Case to Camel Case

HTML  $\Rightarrow$  contenteditable  $\Rightarrow$  normal case

CSS  $\Rightarrow$  Content-editable  $\Rightarrow$  kabab case

JS  $\Rightarrow$  contentEditable  $\Rightarrow$  camel case