

# 1. Inception

Hello world using html

```
<h1>Hello World!</h1>
```

Hello world using Javascript

```
<!-- Hello World using javascript -->

<script>
  const heading= document.createElement("h1");
  heading.innerHTML="Hello World in JavaScript";

  //finding root
  const root=document.getElementById("root");

  //put heading inside root
  root.appendChild(heading);
</script>
```

Hello world using React

## Add React CDN

Both React and ReactDOM are available over a CDN.

```
<script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
```

Browser doesn't understand react so we need to add react cdn

CDN- Content delivery network

This is website their react are hosted and we just pulling react into our project

1st file React.development.js - this is core file of react

2nd file React-dom.development.js- this is react library which is useful for DOM operations or this is react DOM which is need to modify DOM

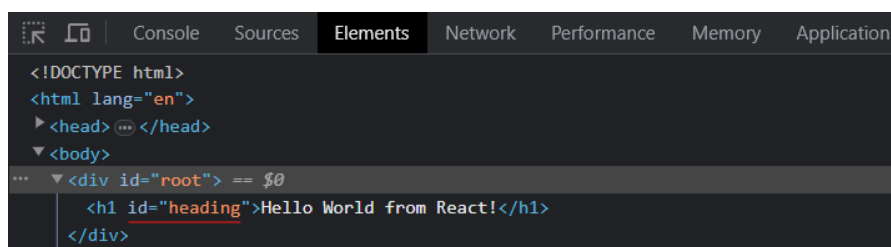
```
<script>
  const heading= React.createElement("h1",{},{}, "hello world")
</script>
```

Here createElement method take 3 arguments

1. What tag we need to create
2. {}
3. What we need put inside it

2.{} is the place where you will give attributes to our tag

```
const heading=React.createElement("h1",{id:"heading"},"Hello World from React!");
const root=ReactDOM.createRoot(document.getElementById("root"));
```



Here heading is React element that is Javascript object

Revision

createElement

createRoot

render

App.js - React code

```
const heading= React.createElement("h1",{id:"heading"},"Hello World from React!");
const root=ReactDOM.createRoot(document.getElementById("root"));
root.render(heading);
```

Index.html -

```
<div id="root"></div>

<script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>

<script src="./App.js"></script>
```

Index.css- css code

Adding Css to H1 tag that id is heading

```
#Heading
{
  color: red;
}
```

```

/*
  <div id="parent">
    <div id="child">
      <h1>I'm H1 tag</h1>
    </div>
  </div>
*/

const parent=React.createElement(
  "div",
  {id:"parent"},
  React.createElement(
    "div",
    {id:"child"},
    React.createElement(
      "h1",
      {}, "I'm H1 tag")))

console.log(parent);
const root=ReactDOM.createRoot(document.getElementById("root"));
root.render(parent);

```

```

/*
  <div id="parent">
    <div id="child">
      <h1>I'm an H1 tag</h1>
    </div>
  </div>
*/

const parent=React.createElement( "div",{id:"parent"},
  React.createElement( "div",{id:"child"},
    React.createElement("h1",{},{ "I'm H1 tag"})));

console.log(parent);
const root=ReactDOM.createRoot(document.getElementById("root"));
root.render(parent);

```

```

▼ <div id="root">
  ▼ <div id="parent">
    ▼ <div id="child" == $0
      <h1>I'm H1 tag</h1>
    </div>
  </div>
</div>

```

ReactElement (object) => HTML (Browser Understand)

**How to create siblings here ?**

```

/*
  <div id="parent">
    <div id="child">
      <h1>I'm an H1 tag</h1>
      <h2>I'm an H2 tag</h2>
    </div>
  </div>
*/

```

So for creating siblings we need to put into array

```

const parent=React.createElement(
  "div",
  {id:"parent"},
  React.createElement( "div",{id:"child"},
    React.createElement("h1",{},{}, "I'm H1 tag"),
    React.createElement("h1",{},{}, "I'm H2 tag"))
);

console.log(parent);
const root=ReactDOM.createRoot(document.getElementById("root"));
root.render(parent);

```

```

/*
<div id="parent">
<div id="child">
  <h1>I'm an H1 tag</h1>
  <h2>I'm an H2 tag</h2>
</div>
<div id="child2">
  <h1>I'm an H1 tag</h1>
  <h2>I'm an H2 tag</h2>
</div>
</div>
if we want to create this type of structure
*/

const parent=React.createElement( "div",{id:"parent"},
[React.createElement( "div",{id:"child"},
  React.createElement("h1",{},{}, "I'm H1 tag"),
  React.createElement("h1",{},{}, "I'm H2 tag")),
  React.createElement( "div",{id:"child2"},
    React.createElement("h1",{},{}, "I'm H1 tag"),
    React.createElement("h1",{},{}, "I'm H2 tag"))]
);

```

But this is messy , it is very complicated to understand so this is why something known as JSX that exist

JSX make life easy when we create tags

```
root.render(parent);
```

Here render putting parent object inside the root

```
<div id="root">  
  <h1>Nikita is here </h1>  
</div>
```

If we have already something in root like  
It will replace by

```
root.render(parent);
```

With this code

React only work inside div id root  
if we have code top of root or bottom of root  
It's code of html work as same

React is Library because react can be applied on small portion of page itself  
React can only work header or footer or sidebar

React only work in the place whatever you make root as

All framework require you to create app using that framework but  
React can work in existing app as well

Create repository on github

In vs code

Git init

Git branch -M main

Create README.md file

Git add .

Git commit -m "episode 1 "

Git remote add origin - copy from github repository and paste - do just once

Git push origin main

