



# React Js Online Training



06

By: Ajeet Kumar





# Images/Assets in React JS

06



# Images/Assets in React JS

## Images/Assets in React JS

- Inside Public Folder
- Inside src Folder

### Inside Public Folder

If you put a file into the public folder, it will not be processed by Webpack. Instead it will be copied into the build folder untouched. To reference assets in the public folder, you need to use a special variable called `PUBLIC_URL`. Only files inside the public folder will be accessible by **%PUBLIC\_URL%** prefix.

```
<link rel="shortcut icon" href="%PUBLIC_URL%/favicon.ico">
```

- None of the files in public folder get bundled or minified.
- Missing files will not be called at compilation time, and will cause 404 errors for your users.
- Result filenames won't include content hashes so you'll need to add query arguments or rename them every time they change.



# Images/Assets in React JS

## Inside src Folder

With Webpack, using static assets like images and fonts works similarly to CSS. You can import a file right in a JavaScript module. This tells Webpack to include that file in the bundle.

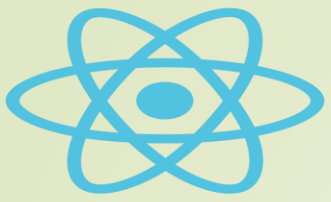
Unlike CSS imports, importing a file gives you a string value. This value is the final path you can reference in your code, e.g. as the src attribute of an image or the href of a link to a PDF.

- Scripts and stylesheets get minified and bundled together to avoid extra network requests.
- Missing files cause compilation errors instead of 404 errors for your users.
- Result filenames include content hashes so you don't need to worry about browsers caching their old versions.



## React Elements in React JS

06



React

# React elements in React JS

## React elements

You can create react element using `React.createElement()` method or JSX

### Using `React.createElement()`

```
React.createElement("h1",null,"Welcome to Sahosoft Solutions");
```

### Using JSX

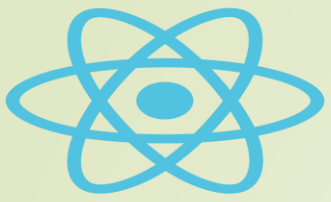
```
<h1>Welcome to Sahosoft Solutions</h1>
```



## createElement() Methods in React JS

06





React

# React elements

## React.createElement()

React.createElement(type, props, Children) – it's used to create react element.

- **Type**: Type of element like : h1, h2, div, span, button etc.
- **props** : Properties objects  
Like : {style :{color: "red"}} or event handlers or name of class etc.
- **Children** : anything , you need to pass in DOM element.

### Example:

```
React.createElement("h1",null,"Welcome to Sahosoft Solutions");
```