

# JSX

- It is neither a string nor HTML.
- It is a html-based syntax extension to JavaScript.
- JSX produces React “elements”
- Instead of artificially separating *technologies* by putting markup and logic in separate files, React *separates concerns* with loosely coupled units called “components” that contain both.
- After compilation, JSX expressions become regular JavaScript function calls and evaluate to JavaScript objects.
- use quotes to specify string literals as attributes
- use curly braces to embed a JavaScript expression in an attribute
- Babel compiles JSX down to `React.createElement()` calls and JSX represents Objects. These objects are called “React elements”

## React.createElement vs JSX

- JSX is not a requirement for using React. Using React without JSX is especially convenient when you don’t want to set up compilation in your build environment.

## Behind the Scenes of JSX

- The process through which React updates the DOM is known as Reconciliation. This reconciliation has 2 phases :
  1. Render Phase
  2. Commit Phase
- The Render phase takes your JSX and turns it into a javascript representation. This is nothing but the VirtualDOM.
- The commit phase is actually taking that representation and applying it to the real DOM. React does batch updates
- Reconciliation = Render + Diffing occurs in between + Commit.

## Babel & parcel role in JSX

### Bundler

A **bundler** lets you write modular code and bundle it together into small packages to optimise load time. Webpack, Parcel.js, Browerify, FuseBox, Rollup, Vite.js

### Compiler

A **compiler** lets you write modern JavaScript code that still works in older browsers.

There’s only one choice here, and that is Babel. **Babel** transforms your modern JavaScript code into a form that is compatible with “older” browsers. Fortunately, Parcel comes built-in with a default Babel configuration that uses the **babel-env** and **browserslist**.

<https://levelup.gitconnected.com/creating-a-react-app-from-scratch-with-parcel-a35da9b36086>

## React Components

Conceptually, components are like JavaScript functions. They accept arbitrary inputs (called "props") and return React elements describing what should appear on the screen.



Functional Components

Composing Components

## Role of type attribute in script tag? What options can I use there?

### Crossorigin

Normal script elements pass minimal information to the window.onerror for scripts which do not pass the standard CORS checks. To allow error logging for sites which use a separate domain for static media, use this attribute.

### Defer

This Boolean attribute is set to indicate to a browser that the script is meant to be executed after the document has been parsed, but before firing DOMContentLoaded.

### Type

This attribute indicates the type of script represented. The value of this attribute will be one of the following:

#### Module

This value causes the code to be treated as a JavaScript module. The processing of the script contents is deferred. The charset and defer attributes have no effect. For information on using module, see our JavaScript modules guide. Unlike classic scripts, module scripts require the use of the CORS protocol for cross-origin fetching.

## Importmap

This value indicates that the body of the element contains an import map. The import map is a JSON object that developers can use to control how the browser resolves module specifiers when importing JavaScript modules.

Attribute is not set (default), an empty string, or a JavaScript MIME type

## Integrity

This attribute contains inline metadata that a user agent can use to verify that a fetched resource has been delivered free of unexpected manipulation

## Async

For classic scripts, if the async attribute is present, then the classic script will be fetched in parallel to parsing and evaluated as soon as it is available. For module scripts, if the async attribute is present then the scripts and all their dependencies will be executed in the defer queue, therefore they will get fetched in parallel to parsing and evaluated as soon as they are available.