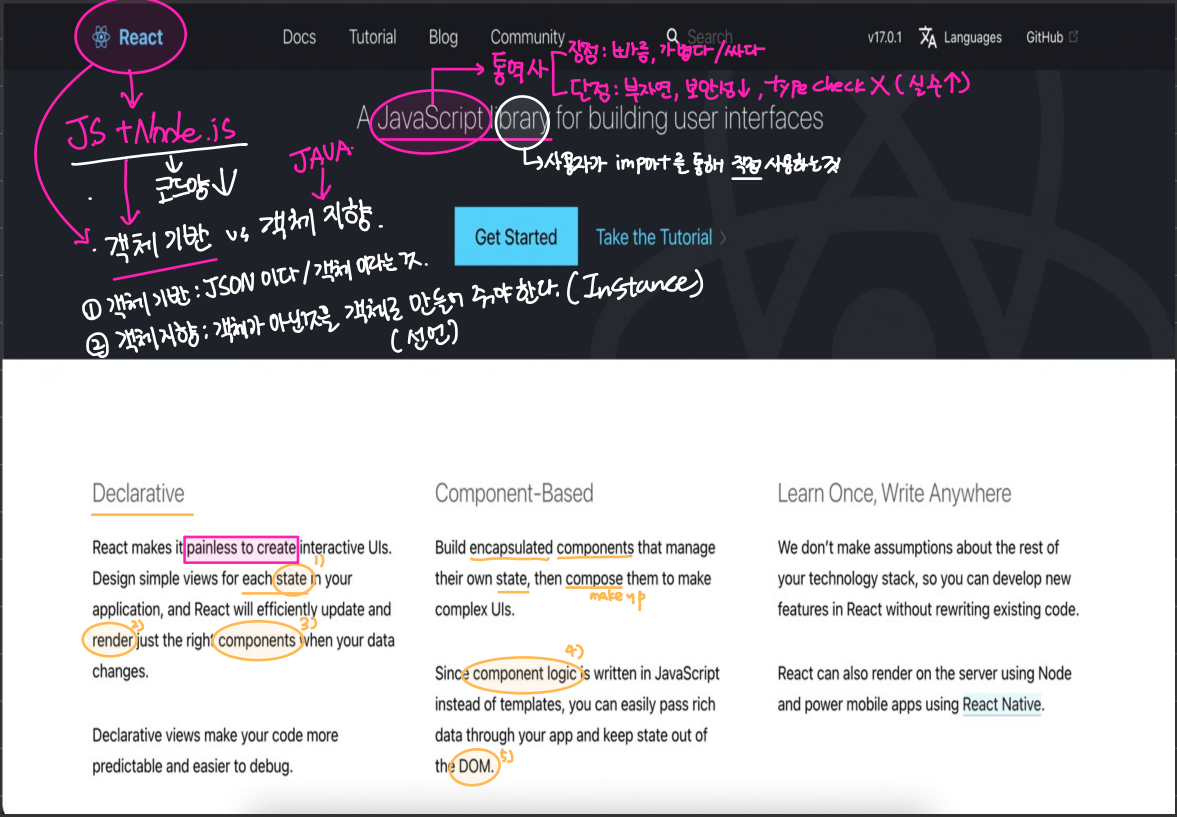


A JavaScript library for building user interfaces

Get Started

Take the Tutorial ›

→ 통역사 [장점 : 빠름, 가볍다 / 싸다
단점 : 부자연, 보안성↓, type check X (실수↑)

↳ 사용자가 import를 통해 직접 사용하는것

JS + Node.js

↓ 코드양↓

JAVA

. 객체 기반 vs 객체 지향.

① 객체 기반 : JSON 이다 / 객체 이라는 것.

② 객체지향 : 객체가 아닌것을 객체로 만들어 주어야 한다. (Instance)
(선언)

Declarative

React makes it painless to create interactive UIs. Design simple views for each state[1] in your application, and React will efficiently update and render[2] just the right components[3] when your data changes.

Declarative views make your code more predictable and easier to debug.

Component-Based

Build encapsulated components that manage their own state, then compose them to make complex UIs.

make up

Since component logic[4] is written in JavaScript instead of templates, you can easily pass rich data through your app and keep state out of the DOM.[5]

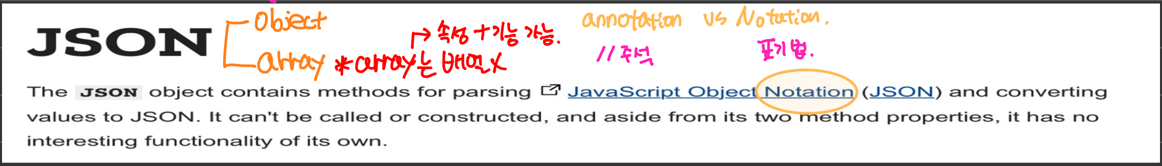
Learn Once, Write Anywhere

We don't make assumptions about the rest of your technology stack, so you can develop new features in React without rewriting existing code.

React can also render on the server using Node and power mobile apps using React Native.

Object Based Progamming

- 객체 지향과 차별화 두려고 노력 → getter/Setter, 인스턴스 선언 하는 것을 '극혐'
- JSON 이다 { }



JSON

[Object
array *array는 배열X

→ 속성 + 기능 가능.

annotation vs Notation.

// 주석

표기법.

The `JSON` object contains methods for parsing JavaScript Object Notation (JSON) and converting values to JSON. It can't be called or constructed, and aside from its two method properties, it has no interesting functionality of its own.

JS : 함수형 프로그래밍으로 진행.

* key + value 의 집합.

value or key

key = value 이면 { } → ( ) 로 변경가능

DOM

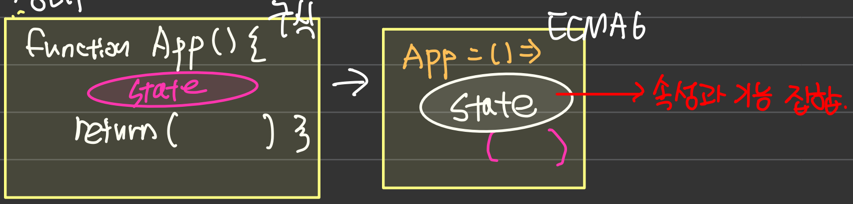
⇒ 김춘수 '꽃' (타자화)

ID를 줘야 객체(DOM)이된다.

① State

- 상태

Arrow function = Lamda



⇒ OOP의 Property 와 같은 개념

함수형P는 State 라고 부름.

② Render

SPA 와 관련

SPA : Single Page Application

페이지 1개 에서 [추가/이동 X] 작동하는 것.

⇒ 이 일을 하는 것이 Rendering ! (뿌려주다)

③ Component

- 쉽게 생각 : Class = Component
- JS : const/let 의 type 은? Value의 타입에 따라 달라짐.

⇒ Component 타입 : JSON.

Componet는 JSON 을 사용하는 것이 아니라 JSON 그 자체.

④ Component Logic

- Component ≠ Component Logic

① 함수형 프로그래밍

```
function App () {

  return {  }

}
```

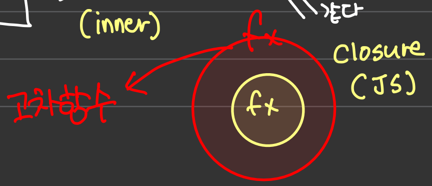
→ ① JSON 경우 : Component (주로 화면)

② 함수 경우 : Logic (서비스)

(inner) \\ 같다

→ Client 측은 이 부분만 볼 수 있다

⇒ 은닉 한다.



⑤ DOM

- Document Object Model

:웹페이지에 대한 프로그래밍 인터페이스

HTML 태그들을 자바스크립트가 이용할 수 있는 객체로 만들고 웹브라우저가 인식하는 방식

*김춘수 `꽃' : 너의 ID를 불러 줬을때 DOM 이된다.

<P> → <P id=" ">

*<Fragment>

↓ ECMA6

<> </>

*<App> = <table> ? NO!

Component Tag

React 관련사항 JSON TYPE = `${boarder:"1px"}`

* <table Style={{boarder:"1px"}}>

key +value

⇒ JAVA의 Map과 같다. But Map은 "김자바" : 1 이다

Why? JAVA에서 key 값은 여러타입이 올수있다 int, String... 미리정한게 와야함.

⇒ JS에서는 JAVA의 Map 특성이 Defalt 이기때문에 " " 안붙여도 됨.

Lamda / Arrow function

자바 J.S

→ ⇒

int App (int a) → a++ : JAVA

Param return

Const App (a) ⇒ a~~ : J.S

이름 Param