**React**

**4. 컴포넌트 만들기(리액트의 전부)**

리액트에서 사용자 정의 태그를 만들 때?

1. 함수(function)을 사용한다.
2. 무조건 대문자로 시작할 것.

function Header() {

return <header>

<h1><a href =”/”></a></h1>

</header>

}

function App() {

return (

<div>

<Header></Header>

<Header></Header>

<Header></Header>

</div>

);

}

리액트에서는 이러한 사용자 정의 태그를 **컴포넌트**라고 한다.

컴포넌트를 사용하면서,

→ 코드의 길이가 줄어들고,

→ 각각의 컴포넌트가 이름을 갖고 있기 때문에 코드의 취지를 알 수 있고,

→ 동시 수정이 가능하다.

**5. Props**

우리가 만든 컴포넌트도 속성(PROP)을 가진다면 얼마나 좋을까?

1. return 값에 들어가는 구문에 {중괄호}를 넣으면, 일반적인 문자열이 아닌 표현식으로 취급되기 때문에 내용이 해석되어 반영된다.

function Nav(props) {

const lis = []

for(let i=0; i < props.topics.length; i++) {

let t=props.topics[i];

lis.push(<li key={t.id}><a href={'/read/'+t.id}>{t.title}</a></li>)

}

return <nav>

<ol>

{lis}

</ol>

</nav>

}

function App() {

const topics = [

{id:1, title:'html', body:'html is ...'},

{id:2, title:'css', body:'css is ...'},

{id:3, title:'javascript', body:'javascript is ...'}

]

return (

<div>

<Nav topics={topics}></Nav>

</div>

);

}

**6. 이벤트**

컴포넌트에도 이벤트 기능이 있어서 컴포넌트에서 어떤 일 이 발생했을 때, 사용자가 추가적인 작업을 처리할 수 있다면 좋겠어!

**7. State**

Prop과 State의 공통점은?!

→ 컴포넌트 함수를 실행해 새로운 리턴값을 만들어 UI를 만든다.

Prop과 State의 차이점은?!

→ 프롭은 컴포넌트를 사용하는 외부자를 위한 데이터. 스테이트는 컴포넌트를 만드는 내부자를 위한 데이터이다.

1. State를 사용하기 위해서는 훅을 작성해 주어야…

→ import {useState} from ‘react’;

useState 설명 URL : <https://dev-yakuza.posstree.com/ko/react/props-state/>