08.09.2025 리액트 수업 정리

**1. 컴포넌트 데이터 흐름과 상태 관리**

**1. Props 기본값과 비구조화 할당**

* **Props 기본값 (Default Props)**: 부모 컴포넌트로부터 특정 props가 전달되지 않을 때 발생할 수 있는 오류를 방지하기 위해 사용합니다. 함수 매개변수에서 직접 기본값을 지정할 수 있습니다.
* **비구조화 할당 (Destructuring Assignment)**: props 객체에서 필요한 값만 추출하여 변수로 바로 사용하는 ES6 문법입니다. 이 방법을 사용하면 코드가 간결해지고 가독성이 높아집니다.

**2. Children Props**

* **children props**는 컴포넌트의 여는 태그와 닫는 태그 사이에 있는 모든 내용(JSX 요소, 텍스트 등)을 전달받는 특별한 속성입니다.
* 이 기능은 컴포넌트를 만들 때 내부 내용을 유동적으로 구성할 수 있게 하여 재사용성을 높여줍니다. 특히 레이아웃 컴포넌트나 컨테이너 컴포넌트를 만들 때 유용합니다.

**3. State**

* ‘state’는 컴포넌트 내부에서 동적으로 변하는 데이터를 관리하는 객체입니다.
* state 값이 변경되면 컴포넌트는 자동으로 다시 렌더링되어 UI를 업데이트합니다.
* ‘props’는 부모로부터 전달받는 **읽기 전용** 데이터인 반면, ‘state’는 컴포넌트 자체에서 변경 가능한 데이터라는 점이 가장 큰 차이입니다.

**2. 리액트 이벤트 시스템과 폼 상태 관리**

**1. 리액트 이벤트 핸들링의 특징**

* **이벤트 이름**: HTML의 소문자 이벤트 이름(예: onclick)과 달리, 리액트에서는 **카멜 케이스**(onClick)를 사용합니다.
* **함수 전달**: 이벤트 핸들러에는 함수를 직접 전달해야 합니다. onClick={handleClick}과 같이 함수 참조를 전달하며, onClick={handleClick()}처럼 함수를 즉시 호출하면 안 됩니다. 만약 매개변수가 필요하다면 **화살표 함수**를 사용하여 onClick={() => handleSomething(data)}와 같이 작성합니다.
* **합성 이벤트 (SyntheticEvent)**: 리액트는 브라우저 고유의 이벤트 시스템을 추상화하여 모든 브라우저에서 일관된 이벤트를 처리하는 합성 이벤트 객체를 제공합니다. 이 객체는 크로스 브라우징 문제를 해결합니다.

**2. 폼 제어 (Controlled Component)**

* **Controlled Component**는 리액트 컴포넌트가 <input>, <textarea>, <select>와 같은 폼 요소의 상태를 ‘state’를 통해 직접 관리하는 방식입니다.
* 사용자의 입력 값은 onChange 이벤트 핸들러를 통해 state에 실시간으로 반영되고, 폼 요소의 value 속성은 이 state와 항상 동기화됩니다. 이를 통해 폼의 상태를 항상 리액트 컴포넌트가 제어할 수 있습니다.

**3. 리액트 이벤트 객체**

* 이벤트 핸들러는 자동으로 **합성 이벤트(SyntheticEvent)** 객체를 매개변수로 받습니다. 이 객체는 브라우저의 네이티브 이벤트를 감싸서 표준화된 인터페이스를 제공합니다.

| **속성** | **설명** |
| --- | --- |
| **e.target** | 이벤트가 실제로 발생한 DOM 요소를 가리킵니다. |
| **e.currentTarget** | 이벤트 핸들러가 바인딩된 DOM 요소를 가리킵니다. |
| **e.type** | 이벤트의 유형을 나타내는 문자열입니다. |
| **e.preventDefault()** | 이벤트의 기본 동작을 막습니다. |
| **e.stopPropagation()** | 이벤트가 상위 DOM 요소로 전파되는 것(버블링)을 막습니다. |
| **e.nativeEvent** | 캡슐화된 실제 브라우저의 네이티브 이벤트 객체입니다. |

* 특정 이벤트 유형에 따른 추가 속성

| **이벤트 유형** | **주요 속성** | **설명** |
| --- | --- | --- |
| **마우스 이벤트** | clientX, clientY | 뷰포트(화면) 기준의 마우스 포인터 좌표 |
| pageX, pageY | 문서(페이지 전체) 기준의 마우스 포인터 좌표 |
| button | 누른 마우스 버튼 (0: 왼쪽, 1: 휠, 2: 오른쪽) |
| **키보드 이벤트** | key | 사용자가 누른 키의 문자열 값 (예: 'a', 'Enter') |
| code | 물리적인 키 코드 (예: 'KeyA', 'Enter') |
| altKey, ctrlKey, shiftKey | 보조키(Alt, Ctrl, Shift)의 눌림 상태 (boolean) |
| **터치 이벤트** | touches | 현재 화면에 닿아 있는 모든 손가락 정보 배열 |
| changedTouches | 이벤트에 변화가 생긴 손가락 정보 배열 |
| **클립보드 이벤트** | clipboardData | 클립보드 데이터에 접근할 수 있는 DataTransfer 객체 |