

Day 5-3. Component Event_0

(Feb 27, 21)

이전까지 한 내용 정리

```
App.js

return(
  <div className="App">
    { /* <Subject title={this.state.subject.title} sub={this.state.subject.sub}></Subject>npx
    <Subject title="React" sub="For UI"></Subject>
    <TOC data={this.state.contents}></TOC>
    <Content title={_title} desc={_desc}></Content>
    */
    <header>
      <h1><a href="/" onClick={function(e){
        console.log('event in', this);
        e.preventDefault();
        return;
        console.log(e);
        this.setState({
          mode: 'welcome'
        })
      }}>{this.state.subject.title}</a></h1>
      {this.state.subject.sub}
    </header>
  </div>
```

a 태그를 클릭했을 때, onClick 이벤트에 설치해놓은 사용자 정의 함수가 호출되면서, App 컴포넌트의 mode값을 welcome으로 바꾸는 코드이다.

이제, Subject 컴포넌트에, Subject를 클릭했을 때, mode의 값을 welcome으로 바꾸는 이벤트를 설치하려 한다.

```
App.js

return(
  <div className="App">
    <Subject
      title={this.state.subject.title}
      sub={this.state.subject.sub}>
      onChangePage={function(){
        this.setState(mode: 'welcome');
      }.bind(this)}
    >
  </Subject>
  {
```

Subject컴포넌트에 onChangePage라는 이벤트를 함수로 추가해주고,

```
Subject.js

class Subject extends Component{

  render() {
    console.log('Subject render');
    return(
      <header>
        <h1><a href="/" onClick={function(e){
          e.preventDefault();
          this.props.onChangePage();
        }.bind(this)}>{this.props.title}</a></h1>
        {this.props.sub}
      </header>
    );
  }
}
```

이벤트가 발생되었을 때, props로 전달되는 onChangePage 호출할 수 있도록, Subject.js의 링크에 onClick을 설치해주고, bind함수를 사용한다.