**필독 Keycloak Javascript Adapter 적용 가이드**

[JS Adapter 기본 사용 방법](https://www.keycloak.org/docs/latest/securing_apps/#_javascript_adapter)

[JS Adapter Reference](https://www.keycloak.org/docs/latest/securing_apps/#javascript-adapter-reference)

참고용 예시

1. 연동 시, 필요 정보
   1. Keycloak Server 주소
   2. Realm 이름
   3. Client Id
2. 준비
   1. npm 사용하지 않는 경우
      * App 실행 시, keycloak.js 파일 load

|  |
| --- |
| <head>  <script src="{keycloak server}/auth/js/keycloak.js"></script> </head> |

1. npm 사용하는 경우
   * package.json에 keycloak-js install

|  |
| --- |
| npm install keycloak-js@11.0.2 |

* App 실행 시, 최초로 실행되는 script 파일에서 keycloak-js import

|  |
| --- |
| import Keycloak from 'keycloak-js'; |

1. 설정 및 초기화

* App 실행 시, keycloak instance 생성( url, realm, clientId 설정) 및 초기화 함수 실행

|  |
| --- |
| <head>  <script src="{keycloak server}/auth/js/keycloak.js"></script>  <script>  const keycloak = new Keycloak({  url: 'https://keycloak-server/auth', *// keycloak server 주소*  realm: 'myrealm', *// keycloak admin에서 생성한 realm 이름*  clientId: 'myapp' *// keycloak admin에서 해당 app과 연결한 clientId*  });  keycloak.init()  .then(function(authenticated) {  *// 정상적으로 초기화된 경우*  alert(authenticated ? 'authenticated' : 'not authenticated');  }).catch(function() {  *// 초기화 실패(해당 로직 이전에 코드 에러나는 경우)*  alert('failed to initialize');  });  </script> </head> |

1. 연동
   1. Login

|  |
| --- |
| keycloak.init()  .then(authenticated => {  if (!authenticated) { *// 로그인 안된 경우*  keycloak.login({  redirectUri: 'client\_redirect\_uri',  locale: 'en' *// login page locale 설정. 'en' or 'ko' 설정.*  }); *// keycloak login page로 redirect*  return;  }  *// 로그인된 경우 실행할 로직 작성*  }).catch(function() {  alert('failed to initialize'); *// 초기화 실패(연동 잘못된 경우)*  }); |

1. Logout (정책에 따라 필요한 곳에 연동)

|  |
| --- |
| keycloak.onAuthLogout = function() {  keycloak.logout(); }; keycloak.onTokenExpired = function() {  keycloak.logout(); }; |

1. Token 사용
   * keycloak API 사용 시, token 사용(= accessToken)

|  |
| --- |
| keycloak.token keycloak.tokenParsed |

* oidc 연결된 Server API 사용 시, idToken 사용

|  |
| --- |
| keycloak.idToken keycloak.idTokenParsed |

* token refresh

|  |
| --- |
| keycloak  .updateToken(minValidity)  .then(refreshed => {  if (refreshed) { *// expired time < minValidity*  console.log('Token was successfully refreshed');  } else { *// expired time > minValidity*  console.log('Token is still valid');  }  })  .catch(() => { *// refresh token 없음*  console.log('Failed to refresh the token, or the session has expired');  }); |

1. Account Console 연동

|  |
| --- |
| window.open(keycloak.createAccountUrl()); *// return 받은 Account Console URL로 tab open* keycloak.accountManagement(); *// Account Console로 바로 redirect* |

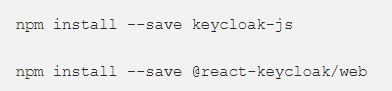
**리액트 App에서 keycloak 사용할 경우**

([@react-keycloak/web](https://www.npmjs.com/package/@react-keycloak/web) 패키지 적용한 예시)

필독 : [react-keycloak/web README.md](https://github.com/react-keycloak/react-keycloak/blob/master/packages/web/README.md)

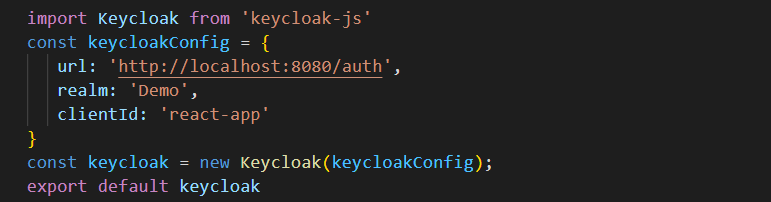
리액트 프로젝트 내 패키지 설치

@react-keycloak/web 설치 전 keycloak-js 가 설치되어 있어야 합니다.

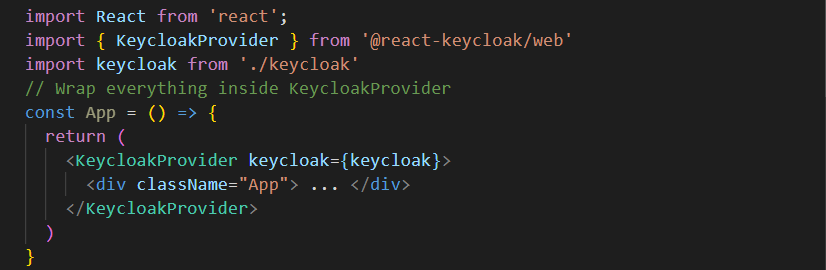


**Keycloak instance 설정**

프로젝트의 src 폴더에 다음 내용으로 keycloak.js 파일을 생성하여 필요에 따라 Keycloak 인스턴스를 설정합니다. 다른 설정값이 필요할 경우 config 객체 안에 추가.



**KeycloakProvider 설정**

KeycloakProvider로 App을 래핑하고 위에서 생성한 keycloak 인스턴스를 prop으로 전달합니다.  


그 외 설정 값

초기 옵션 : initOptions props로 설정 가능

이벤트 핸들러 함수 : onEvent props로 설정 가능

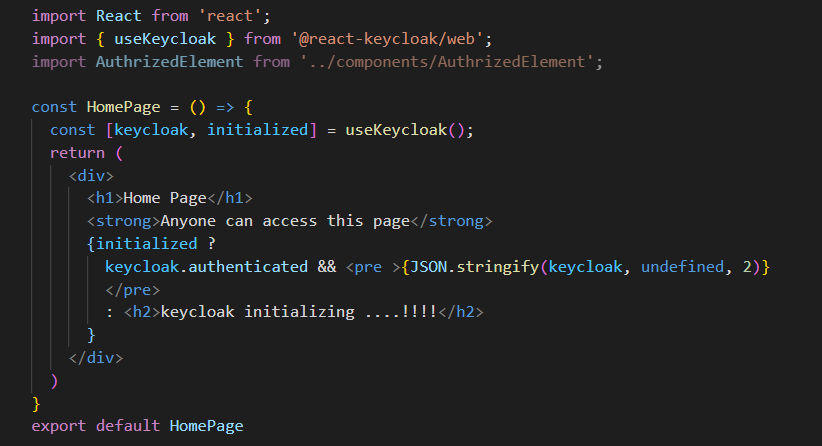
<https://www.npmjs.com/package/@react-keycloak/web#getting-started> 참고

**컴포넌트에서 keycloak접근 방법**

useKeycloak  Hook (React >=16.8 required).

함수형 컴포넌트 내부에서 useKeycloak 사용.

예시 )



withKeycloak HOC (Higher-Order Components).

클래스형 컴포넌트에서 withKeycloak 이용.

예시)

