

쿠키(Cookie)





➤ 쿠키(Cookie) 란 ?

- 쿠키는 웹사이트가 사용자의 브라우저에 저장하는 작은 데이터 조각이다.
- 로그인 정보, 사이트 설정, 장바구니 내용처럼 사용자(User)가 다시 방문했을 때 기억 해야 할 정보를 저장해 둔다.

➤ 쿠키(Cookie) 기본 개념

| 항목 | 설명 |
|----------|--------------------------------|
| 저장 위치 | 사용자의 브라우저(Local에 저장) |
| 크기 제한 | 약 4KB (작은 용량) |
| 저장 형태 | key=value 쌍 (예: token=abc123) |
| 자동 전송 여부 | 브라우저가 자동으로 서버에 전송 |
| 유지 기간 | 만료 시간(expires / max-age) 설정 가능 |
| 주요 사용 예 | 자동 로그인, 방문 기록, 언어 설정 등 |



➤ 쿠키(Cookie)의 특징(장단점)

✓ 장점

- 자동으로 서버와 동기화되어 인증용으로 편리함
- 여러 페이지 간 데이터 공유 가능

✓ 단점

- 용량이 작음 (4KB 제한)
- **보안에 취약**할 수 있음 (노출될 가능성)
- 너무 많이 쓰면 **트래픽 증가** (매 요청마다 전송됨)

➤ 쿠키(Cookie) 주요 속성

| 속성 | 설명 |
|----------|---------------------------------------|
| expires | 쿠키 만료 시간 (날짜로 지정) |
| max-age | 쿠키 유지 시간 (초 단위) => js-Cookie에서는 사용 못함 |
| path | 쿠키가 유효한 경로 지정 |
| domain | 유효한 도메인 지정 |
| secure | HTTPS 환경에서만 전송 |
| HttpOnly | 자바스크립트에서 접근 불가 (보안용) |



➤ 쿠키(Cookie)의 동작 과정

1. 사용자가 웹사이트에 접속 → 서버가 응답할 때 Set-Cookie 헤더를 함께 보낸다.

```
Set-Cookie: userToken=abc123; Path=/; HttpOnly; Max-Age=3600
```

2. 브라우저는 이 정보를 저장하고, 다음에 같은 사이트에 요청을 보낼 때 자동으로 쿠키를 포함해서 보낸다.

```
Cookie: userToken=abc123
```

3. 서버는 이 쿠키를 보고 사용자를 식별하거나, 이전 상태를 복원할 수 있다.



➤ 쿠키(Cookie)의 시간 설정 방법

- 브라우저가 시간을 밀리초(ms) 단위로 계산하기 때문에 시간 → 밀리초(ms)로 바꿔서 계산해야 한다.
- 1초 = 1000밀리초(ms)
- 1분 = 60초
- 1시간 = 60분

| 시간 | 밀리초 계산법 |
|-----|----------------------|
| 1초 | 1000 |
| 1분 | $60 * 1000$ |
| 1시간 | $60 * 60 * 1000$ |
| 2시간 | $2 * 60 * 60 * 1000$ |



➤ React에서 쿠키(Cookie) 관리 라이브러리 설치

```
npm install js-cookie
```

➤ 쿠키(Cookie) 사용 예시

```
import Cookies from 'js-cookie';
```

```
// 쿠키 저장
```

```
Cookies.set('userToken', 'abc123', { expires: 7 }); // 7일간 유지
```

```
// 쿠키 읽기
```

```
const token = Cookies.get('userToken');  
console.log(token);
```

```
// 쿠키 삭제
```

```
Cookies.remove('userToken');
```



➤ 여기서 잠깐

```
import Cookies from 'js-cookie';
import {useState} from 'react'

export default function CookieTest(){
    // 쿠키를 읽어와서 cookieValue의 상태변수에 저장한다.
    const [cookieValue, setCookieValue] =
    useState(Cookies.get('userName'));

    // 쿠키를 저장하는 함수를 생성
    const CookieSave = () => {
        Cookies.set('userName','홍길동',{expires:1})//1루 동안 유지
        setCookieValue('홍길동')
        alert('쿠키 저장됨')
    }

    // 쿠키 확인하는 함수
    const CookieChk = () => {
        const value = Cookies.get('userName')
        if(value){
            alert(`쿠키 ${value}`)
        }else{
            alert('쿠키가 없습니다.')
        }
    }

    // 쿠키 삭제하는 함수 생성
    const CookieDel = () => {
        Cookies.remove('userName')
        setCookieValue('')
        alert('쿠키가 삭제 되었습니다.')
    }
}
```



➤ 여기서 잠깐

```
return(  
  <>  
  <div>  
    <h2>쿠키 예제</h2>  
    <p>현재 쿠키 값 : {cookieValue}</p>  
    <button type='button'  
      style={{backgroundColor:'dodgerblue',color:'#fff'}}  
      onClick={CookieSave}>쿠키 저장</button>  
    <button type='button'  
      style={{backgroundColor:'dodgerblue',color:'#fff'}}  
      onClick={CookieChk}>쿠키 확인</button>  
    <button type='button'  
      style={{backgroundColor:'dodgerblue',color:'#fff'}}  
      onClick={CookieDel}>쿠키 삭제</button>  
  </div>  
)
```

React Cookie 기본 예제



여기서 잠깐 - 출력 결과

쿠키 예제

현재 쿠키 값 :

쿠키 저장 쿠키 확인 쿠키 삭제

쿠키 예제

현재 쿠키 값 : 흥길동

쿠키 저장 쿠키 확인 쿠키 삭제

localhost:5173의 메시지

쿠키가 삭제 되었습니다.

확인

쿠키 예제

현재 쿠키 값 :

쿠키 저장

쿠키 확인

쿠키 삭제

Application에서 -> cookie 확인

Application

Application

- Manifest
- Service workers
- Storage

Storage

- Local storage
- Session storage
- Extension storage
- IndexedDB
- Cookies

localhost:5173

| Name | Value | Do... | Pa... | Expi... | Size | Htt... | Sec... | Sa... |
|----------|-----------------------------|----------|-------|---------|------|--------|--------|-------|
| userName | %ED%99%8D%EA%B8%B8%EB%8F%99 | local... | / | 202... | 35 | | | |

로그인 성공 시
이름 유지





문제] 다음의 조건에 만족하도록 React를 작성 하시오.

조건

- ① 로그인 화면에서 아이디 입력 후 아이디를 쿠키에 1분 동안 저장하도록 구현하시오.
- ② 로그인 화면에서 아이디 입력, 비밀번호 입력 후 아이디 저장을 체크한 후 화면을 새로고침을 해도 1분 동안은 로그인 화면의 아이디 입력창에 쿠키에 저장된 아이디가 유지되도록 기능을 구현하시오.
- ③ 버튼 클릭 시 텍스트 표시 여부 변경
- ④ src 폴더안에 -> cookie 폴더 생성 -> LoginCookie.jsx 파일을 생성하여 작성하시오.
- ⑤ src 폴더안에 -> cookie 폴더 생성 -> LoginNaver.css 파일을 생성하여 작성하시오.
- ⑥ src 폴더안에 -> App.jsx에 작성한 LoginCookie.jsx 를 import하여 실행 하시오.

출력 결과

NAVER

ID전화번호

아이디 또는 전화번호

비밀번호

아이디 저장

IP보안

로그인



NAVER

ID전화번호

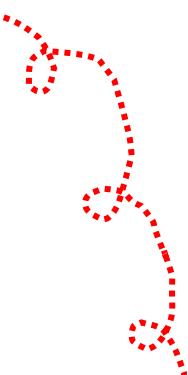
아이디 또는 전화번호 (highlighted with a red box)

비밀번호

아이디 저장

IP보안

로그인



애플리케이션

- 매니페스트
- Service workers
- 저장용량

저장용량

- 로컬 스토리지
- 세션 저장소
- 확장 프로그램 저장용량
- IndexedDB
- 쿠키

필터

| 이름 | 값 |
|--------|-----|
| userId | kkk |

http://localhost:5173



새로 고침 후의 출력 결과

NAVER

ID전화번호

kkk

비밀번호

아이디 저장 IP보안

로그인

1분 후 새로 고침 후의 출력 결과

NAVER

ID전화번호

아이디 또는 전화번호

비밀번호

아이디 저장 IP보안

로그인