

Javascript 모달 만들기

# 17번째 세션

---

NEXT X LIKELION 김이린



# 목차

1. 모달이란? (cf. 팝업)

2. 기본 모달 실습

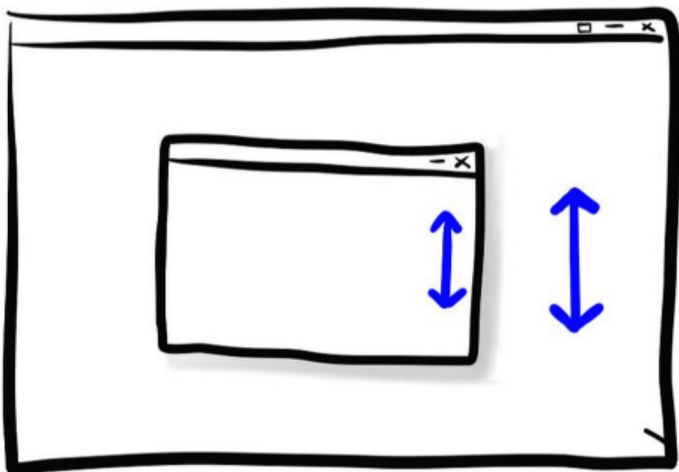
3. ‘하루 동안 열지 않기’ 모달 실습

+ JS Keywords: 모듈, 쿠키, 화살표 함수, 이벤트 핸들러

# 👋 모달? 팝업?

## 팝업(Popup)

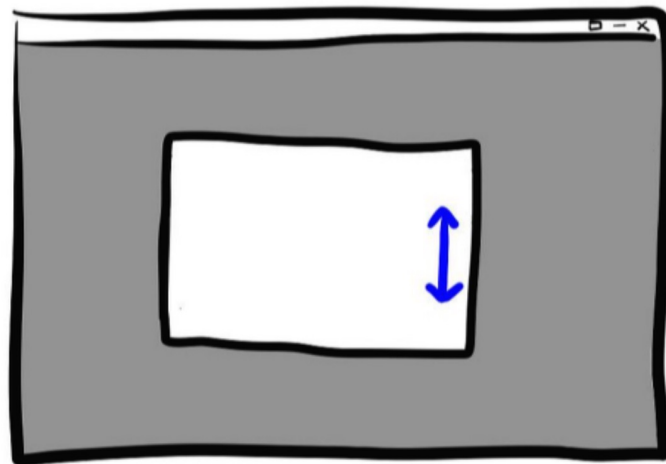
: 현재 화면에 새로운 하나의 창(Browser)을 생성해서 다른 화면을 보여주는 기능



1. 열려 있던 브라우저 페이지 위에, 또 다른 브라우저 페이지를 띄우는 것
2. 원래 브라우저와 독립적인 관계
3. 브라우저의 옵션을 통해 열리지 않도록 강제 가능

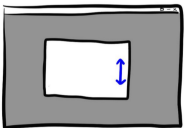
## 모달(Modal)

: 화면 위에 하나의 작은 레이어를 더 얹어 부가적인 일들을 처리하게 만드는 기능



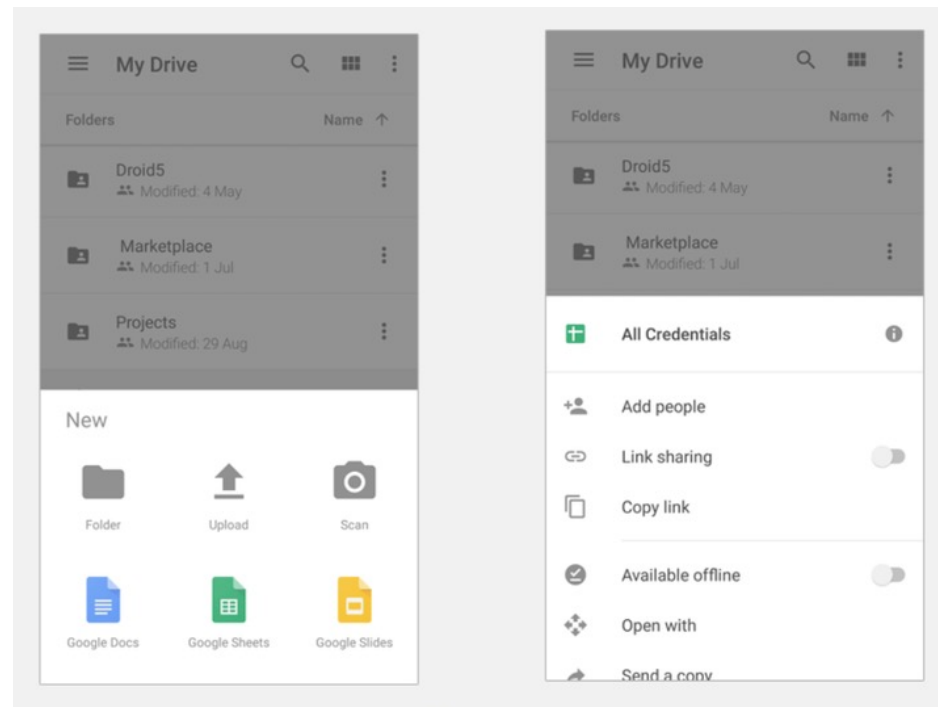
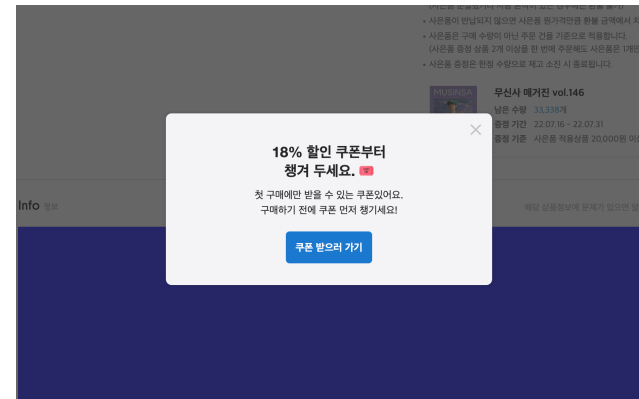
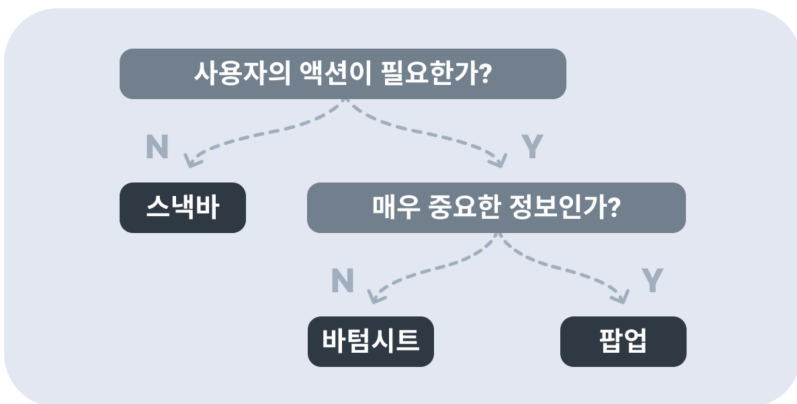
1. 열려 있던 브라우저 위에, 새로운 창이 아닌 레이어를 까는 것
2. 원래 브라우저와 종속적인 관계, 부모 - 자식 형태
3. 브라우저 옵션에 영향 받지 않음

# 모달



모달이 모달이고 팝업도 모달이고....

- 1) 팝업(Popup): Popup, Modal dialogs, Dialog, Alert
- 2) 바텀 시트(Bottom sheet)
- 3) 토스트(Toast) :Snackbar, Toast



✓ 자산을 모두 연결했어요.

# 논모달

## NonLightBox

Dim이 들어가지 않은 Background

A



B



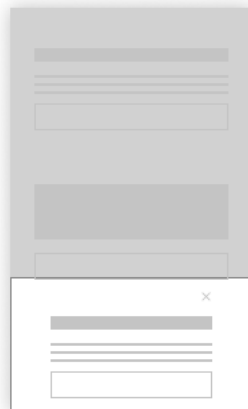
## Modal

사용자가 상위/부모 항목의  
컨텐츠와 상호작용 불가능

C



D



## NonModal

사용자가 상위/부모 항목의  
컨텐츠와 상호작용 가능

## LightBox

Dim이 들어간 Background

일반 모달은 부모 화면을 사용자가 컨트롤할 수 없고,  
모달 내 컨텐츠를 완료하거나 닫아야 다시 부모 화면에 접근 가능  
but 논모달은 부모화면을 사용자가 컨트롤 가능

# 모달 사용

**최근 UX를 고려해 전통적인 팝업창은 잘 사용하지 않는 경향**

비슷한 용도로 모달을 많이 쓰지만,

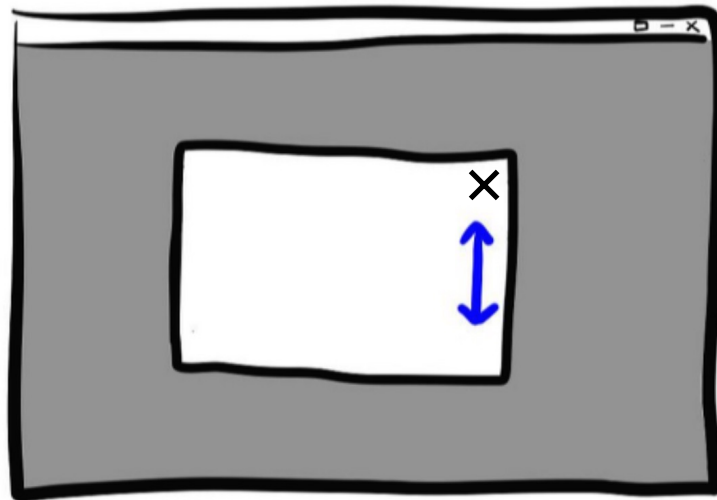
- ‘팝업’과 혼용하는 등 기업마다 워딩이 달라서 용어에 혼동이 많고
- 꼭 써야만 좋은 것이 아니기 때문에

**→ UI, UX를 신중하게 고려해서, 스스로 원하는 형태에 따라 레퍼런스를 체크하고 검색해서 구현하자!**

\* 기업들의 UI Guideline을 기획, 디자인시 참고해도 좋아요~

<https://developer.apple.com/design/human-interface-guidelines/patterns/modality/>

# 실습 - 기본 모달 구현



## 요구사항

- 특정 버튼을 누르면 모달이 켜진다. 모달을 끄지 않는 한 기존 페이지는 어둡게 처리되며 컨트롤할 수 없다.
- 모달의 닫기(x) 버튼을 누르면 모달이 꺼진다.
- 모달 바깥 어두운 영역 (backdrop: 배경)을 클릭하면 모달이 꺼진다.

# 실습 - 기본 모달 구현

## ✍️ 요구사항 분석 (개발 순서 계획)

- ☐ index.html에 모달 요소 만들기
- ☐ 모달 요소에 CSS 적용하기
- ☐ 모달을 여는 함수 openModal
- ☐ 모달을 닫는 함수 closeModal
- ☐ 카드 영역에 클릭 이벤트가 발생하면 openModal 호출
- ☐ 모달 닫기 버튼 클릭, 모달 바깥 어두운 영역 (backdrop: 배경) 클릭 이벤트가 발생하면 closeModal 호출

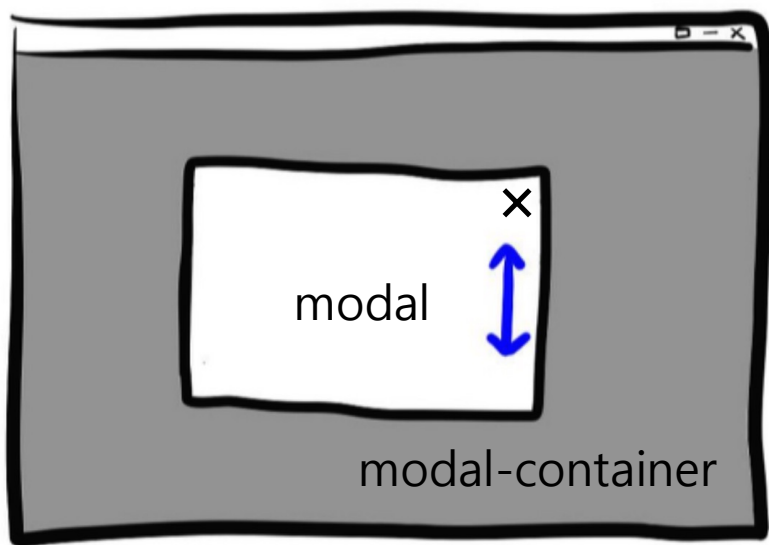


# 실습 - 기본 모달 구현

index.html

✍️ 요구사항 분석 (개발 순서 계획)

☑️ index.html에 모달 요소 만들기



```
1  <!-- 모달을 강조할 배경 레이어 -->
2  <div class="modal-container">
3    <!-- 배경 위에 얹어지는 실제 모달 컨테이너 -->
4    <div class="modal">
5      <!-- 모달 상단 헤더 -->
6      <div class="modal-header">
7        <!-- 상단 제목 -->
8        <p class="modal-title">모달</p>
9        <!-- 닫기 버튼 -->
10       <button class="modal-close">
11         
12       </button>
13     </div>
14     <!-- 모달 내용 -->
15     <article class="modal-body">모달 내용 히히하하호호</article>
16   </div>
17 </div>
```

# 실습 - 기본 모달 구현

✍️ 요구사항 분석 (개발 순서 계획)

☑️ index.html에 모달 요소 만들기

☑️ 모달 요소에 CSS 적용하기

css/ui/modal.css

```
1 .modal-container {  
2   display: flex;  
3   position: fixed;  
4   top: 0;  
5   right: 0;  
6   bottom: 0;  
7   left: 0;  
8   background: rgba(0, 0, 0, 0.5);  
9   backdrop-filter: blur(4px);  
10  transition: opacity 0.25s ease;  
11  /* 투명도 */  
12  
13 }  
14  
15 .modal-container.open {  
16  
17 }
```

# 실습 - 기본 모달 구현

- ☒ index.html에 모달 요소 만들기
- ☒ 모달 요소에 CSS 적용하기
- ☐ 모달을 여는 함수 openModal
- ☐ 모달을 닫는 함수 closeModal

## HINT

- modal 요소에 'open' 이라는 class를 넣고 빼면서 동작한다!  
<https://developer.mozilla.org/ko/docs/Web/API/Element/classList>
- 모달은 잘 뜨는데 흐린 배경이 스크롤되면 안되니까 overflow 속성 사용

# 실습 - 기본 모달 구현

☑ index.html에 모달 요소 만들기

☑ 모달 요소에 CSS 적용하기

☑ 모달을 여는 함수 openModal

☑ 모달을 닫는 함수 closeModal

```
1 const $ = (selector) => document.querySelector(selector);
2 const body = $("body");
3 const modal = $(".modal-container");
4
5 const openModal = () => {
6
7
8 };
9
10 const closeModal = () => {
11
12
13 };
14
```

# 실습 - 기본 모달 구현

☒ index.html에 모달 요소 만들기

☒ 모달 요소에 CSS 적용하기

☒ 모달을 여는 함수 openModal

☒ 모달을 닫는 함수 closeModal

☐ 카드 영역에 클릭 이벤트가 발생하면 openModal 호출

☐ 모달 닫기 버튼 클릭, 모달 바깥 어두운 영역 (backdrop: 배경) 클릭 이벤트가 발생하면 closeModal 호출

## 1 HINT

- 특정 클래스 요소에 대해 (querySelector) 지정한 유형의 이벤트(click)를 대상이 수신할 때마다 호출할 함수를 설정

# 실습 - 기본 모달 구현

- ☑ index.html에 모달 요소 만들기
- ☑ 모달 요소에 CSS 적용하기
- ☑ 모달을 여는 함수 openModal
- ☑ 모달을 닫는 함수 closeModal
- ☑ 카드 영역에 클릭 이벤트가 발생하면 openModal 호출
- ☑ 모달 닫기 버튼 클릭, 모달 바깥 어두운 영역 (backdrop: 배경) 클릭 이벤트가 발생하면 closeModal 호출

# 모듈

js/modal.js

자네 항상 모달 함수를 index.js에서 새로 만들어 쓸 테야?

<https://ko.javascript.info/import-export>

script 하나, 파일 하나 = 모듈 하나

- export 지시자를 변수나 함수 앞에 붙여 외부 모듈에서 해당 변수나 함수에 접근 (모듈 내보내기)
- import 지시자를 사용해 외부 모듈의 기능을 가져오기 (모듈 가져오기)

사용 예시)

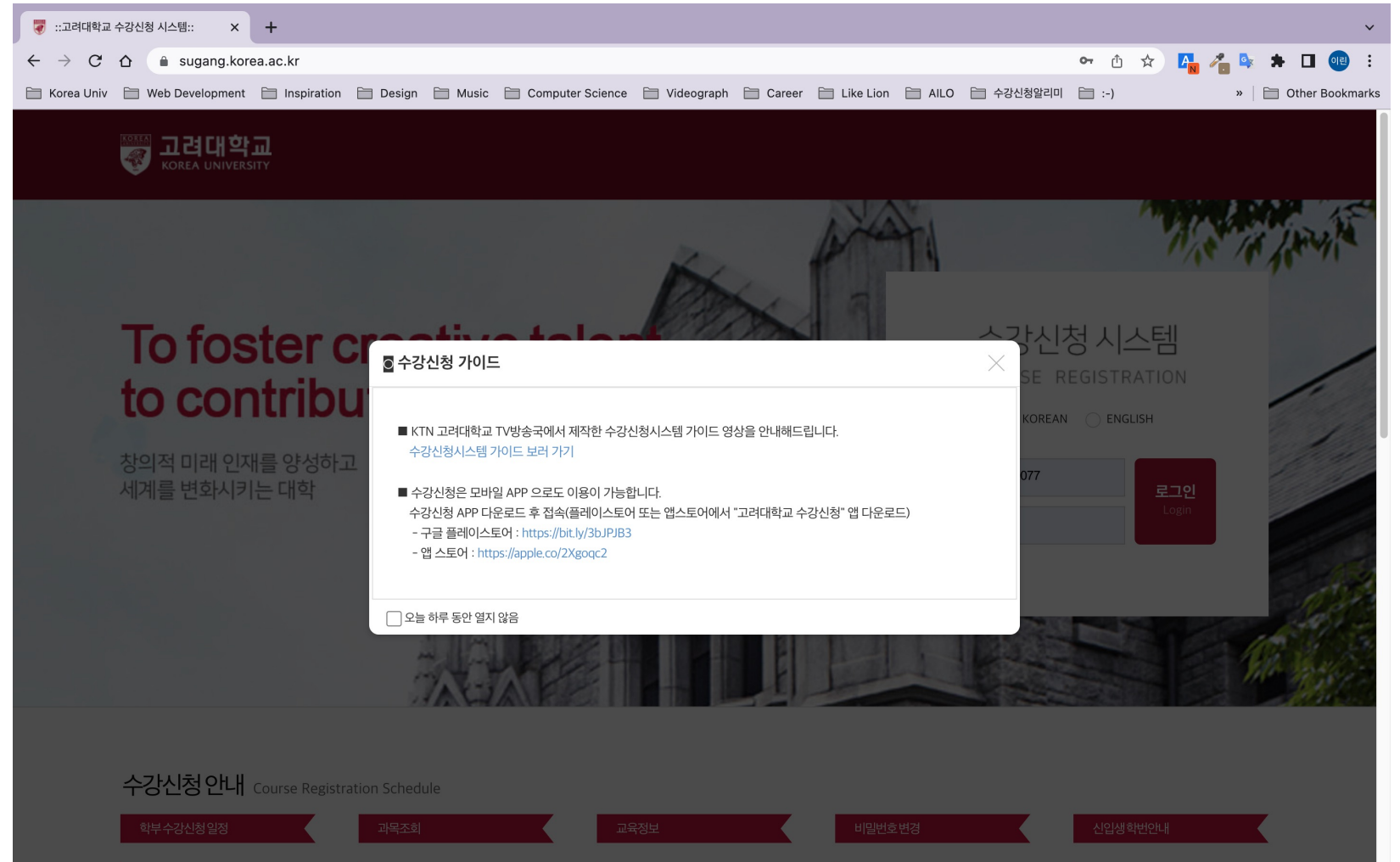
```
1 // 📁 sayHi.js
2 export function sayHi(user) {
3   alert(`Hello, ${user}!`);
4 }
```

```
1 // 📁 main.js
2 import {sayHi} from './sayHi.js';
3
4 alert(sayHi); // 함수
5 sayHi('John'); // Hello, John!
```



# 실습 - ‘오늘 그만 보기’ 모달

<https://sugang.korea.ac.kr/>







# 실습 - ‘오늘 그만 보기’ 모달

## 요구사항

- 페이지 최초 렌더링시 모달이 켜진다.
- 닫기 버튼 또는 모달 바깥 어두운 영역을 클릭하면 모달이 꺼진다.
- 모달이 꺼져도 페이지 재접속 및 새로고침시 모달이 다시 켜진다.
- ‘오늘 그만 보기’ 버튼을 누르면 모달이 꺼지고 재접속 및 새로고침시에도 모달이 켜지지 않는다.



# 실습 - ‘오늘 그만 보기’ 모달



## 요구사항 분석 (개발 순서 계획)

- ☐ 처음 페이지가 렌더링될 때 모달을 여는 함수를 실행한다.
- ☐ ‘오늘 그만 보기’에 대한 쿠키를 읽어와 조건을 만족하면 최초 렌더링시 모달을 여는 함수가 실행되지 않는다.
- ☐ ‘오늘 그만 보기’ 버튼을 누르면 1일 기한의 쿠키를 만든다.

## 복습 - 쿠키와 웹 스토리지

	Cookie	Session Storage (Web Storage)	Local Storage (Web Storage)
개당 사용 가능 용량	최대 4KB	평균 5MB	평균 5MB
유지 기간	설정 가능	한 세션 (탭 닫을 시 내용 제거)	사용자가 삭제하기 전까지
통신 여부	서버와 계속 연결	설정 가능	설정 가능
사용처	기간 설정 필요한 부분 (팝업 창)	보안을 필요로 하는 경우 (은행 로그인 등)	상시 저장이 필요한 경우 (로그인 유지 등)

# 복습 - 템플릿 리터럴

## Template String

이중 따옴표(“ ”)나 작은 따옴표 ( ‘ ’ ) 대신 백틱 ( ` ` )을 사용하여 변수를 표현하는 문법

``${변수명}``



```
const a = 5;  
const b = 10;  
console.log(`a + b = ${a+b}입니다`);
```



# 쿠키 사용법

<https://ko.javascript.info/cookie>



```
1 // document.cookie에 값을 할당하면, 브라우저는 이 값을 받아 해당 쿠키를 갱신(다른 쿠키의 값은 변경 X)  
2 // 아래와 같이 코드를 작성하면 이름이 user인 쿠키를 찾아 그 값을 Next으로 갱신  
3 document.cookie = "user=Next";
```



# 쿠키 사용법

<https://ko.javascript.info/cookie>



```
1 // path, domain, expires, secure 등 옵션 사용 가능  
2 // 옵션은 key=value 뒤에 나열하고 ;로 구분  
3 document.cookie = "user=Next; path=/; expires=Tue, 19 Jan 2038 03:14:07 GMT"
```

# 사용자 정의 함수 - setCookie

js/cookie.js

`name`, `value`, `expireDays` 를 인자로 받아 어떤 쿠키를 만료기한을 얼마나 설정할지 정한다.

```
1 // 만료기한을 정해 쿠키 생성하기
2 const setCookie = (name, value, expireDays) => {
3   // 현재 날짜와 시간이 저장된 Date 객체 생성하기
4
5   // 오늘 날짜에 expireDays만큼 더해서 만료시간을 구하기
6
7   // 쿠키 생성하기 (템플릿 리터럴 활용)
8   document.cookie = ``;
9 };
```

# 사용자 정의 함수 - getCookie

js/cookie.js

```
1 // 특정 이름의 쿠키를 가져오기
2 const getCookie = (name) => {
3   // 로컬에 저장된 모든 쿠키 읽어오기
4   let cookie = document.cookie;
5   // 로컬에 저장된 쿠키가 존재하면 쿠키를 배열에 저장해서 배열을 순회하며 내가 원하는 이름의 쿠키를 리턴하기
6   if (document.cookie) {
7     // 배열을 순회하면서 쿠키의 name에 대한 value값을 리턴 (반복문)
8   }
9   return;
10 };
```





# 실습 - '오늘 그만 보기' 모달



```
1  this.init = () => {  
2    //스케줄 불러오는 함수를 호출해 첫 페이지 렌더링시 화면에 보여준다.  
3    renderSchedule();  
4    // 모달을 닫는 쿠키가 있는지 확인한다.  
5  
6    // 모달을 닫는 쿠키가 없다면 무조건 모달을 열어둔다.  
7  
8  };
```



## 실습 - ‘오늘 그만 보기’ 모달



### 요구사항 분석 (개발 순서 계획)

- ☒ 모달을 여는 함수를 처음 페이지가 렌더링될 때 실행한다.
- ☒ ‘오늘 그만 보기’에 대한 쿠키를 읽어와 조건을 만족하면 최초 렌더링시 모달을 여는 함수가 실행되지 않는다.
- ☐ ‘오늘 그만 보기’ 버튼을 누르면 1일 기한의 쿠키를 만든다.



## 실습 - '오늘 그만 보기' 모달



```
1  // 오늘 보지 않기 버튼을 누르면 만료기간이 1일인 쿠키를 생성하고 모달을 닫는다.  
2  $(".modal-stop-button").addEventListener("click", () => {  
3      // 하루 기한의 쿠키를 생성한다. (쿠키 생성시 이름은 modal-closed, 값은 true로 설정한다.)  
4  
5      // 모달을 닫는다.  
6  
7  });
```



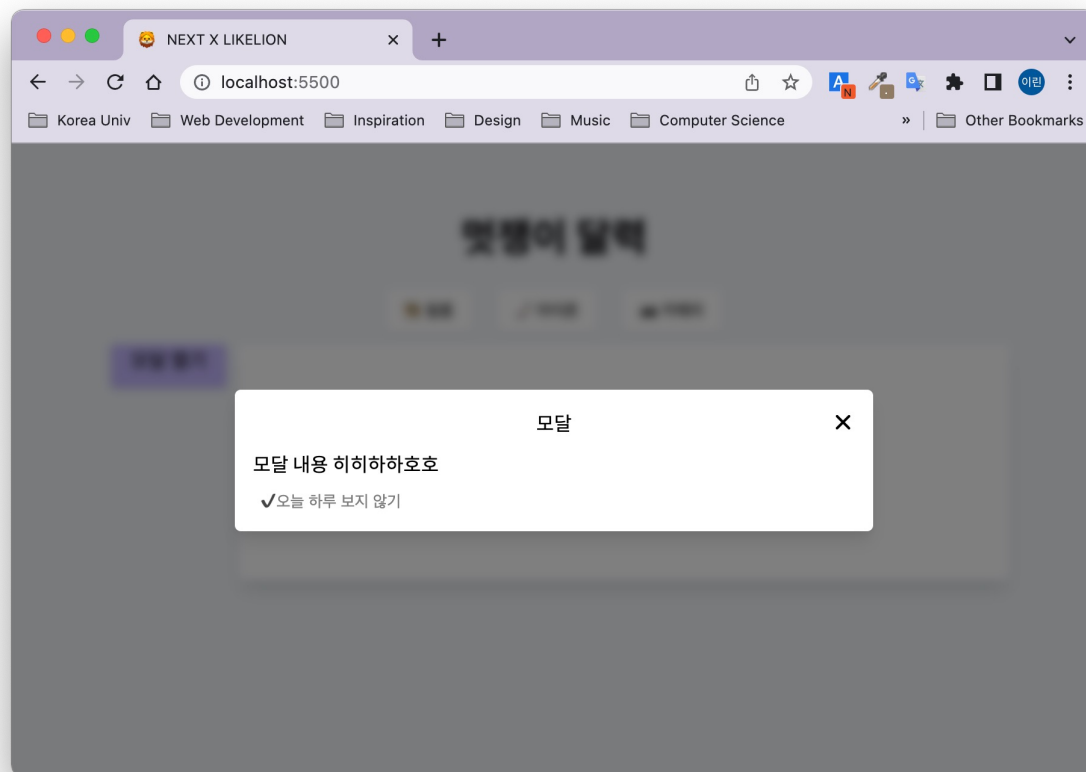
# 과제

~8월 8일 19:00까지

옵션 1) 특정 목적의 모달이 열리게 만들기(ex. 제목 클릭시 정보 안내) + ‘다시 보지 않기’ 버튼 기능 추가하기

옵션 2) 기존에 진행하던 프로젝트가 있다면 해당 프로젝트에서 모달 만들기

옵션 3) 페이지에 모달과 비슷한 원리로 작동하는 UI인 ‘햄버거 메뉴’ 만들기





---

수고하셨습니다! 8월 힘차게 시작해보아요~

NEXT X LIKELION   김이린