



와이파이 비번 : **careercollege**

자료공유 카톡방 : 인터넷 주소창에 입력

<https://open.kakao.com/o/gQrDNCmb>

코드 에디터 : 아무거나

1. HTML Preview 기능 필요
2. 자바스크립트 자동완성 기능 추천
3. 귀찮으면 Brackets 설치
4. 아니면 VS code 이런거



목표 쇼핑몰 (내용 디자인만)

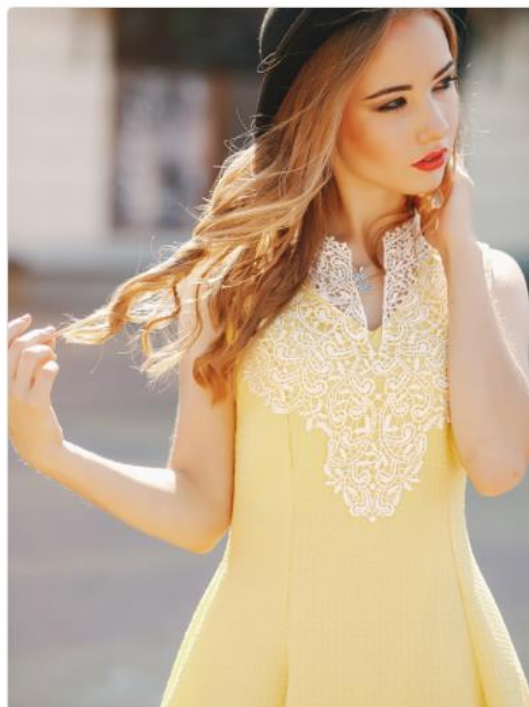
<http://kingstudio.ro/demos/incart/index3.html>



New Nordstrom line

What you've expect

Product line



Blossom dress

Do you like some random yellow stuff? Here's one for you.

from \$299

Details



Springfield shirt

This shirt is from Massachusetts, and we don't mean any springfield bolt-action rifles.

from \$99

Details



Black monastery

The best beer comes from monasteries. Even the best dress comes from them.

from \$199

Details







Details



Springfield Shirt

This shirt is from Massachusetts, and we didn't even mean to promote any Springfield bolt-action rifles.

[Add to Cart](#)[My Cart](#)

Product		Price	Quantity	Total	
	Women's Shirt	\$59	<input type="text" value="1"/>	\$59	
	Women's Jeans	\$69	<input type="text" value="2"/>	\$138	

Apply Coupon

Update Cart

Description

Product Info

Reviews (2)



Simply the Best!

★★★★★

Aliquam pellentesque nisl nec ultricies scelerisque. Proin vel blandit magna. Class aptent taciti sociosqu.

- Johana Doe



Good Product.

★★★★☆

Vestibulum luctus justo justo, ac iaculis dolor luctus nec. In nec molestie mi. Praesent blandit interdum neque.

- Nicole Dowe

Front-end Stacks

HTML

CSS

Javascript (+ES6)

jQuery

Design library (Bootstrap)

HTML/CSS Preprocessor

Javascript Library (Angular/React/Vue)

기타 :

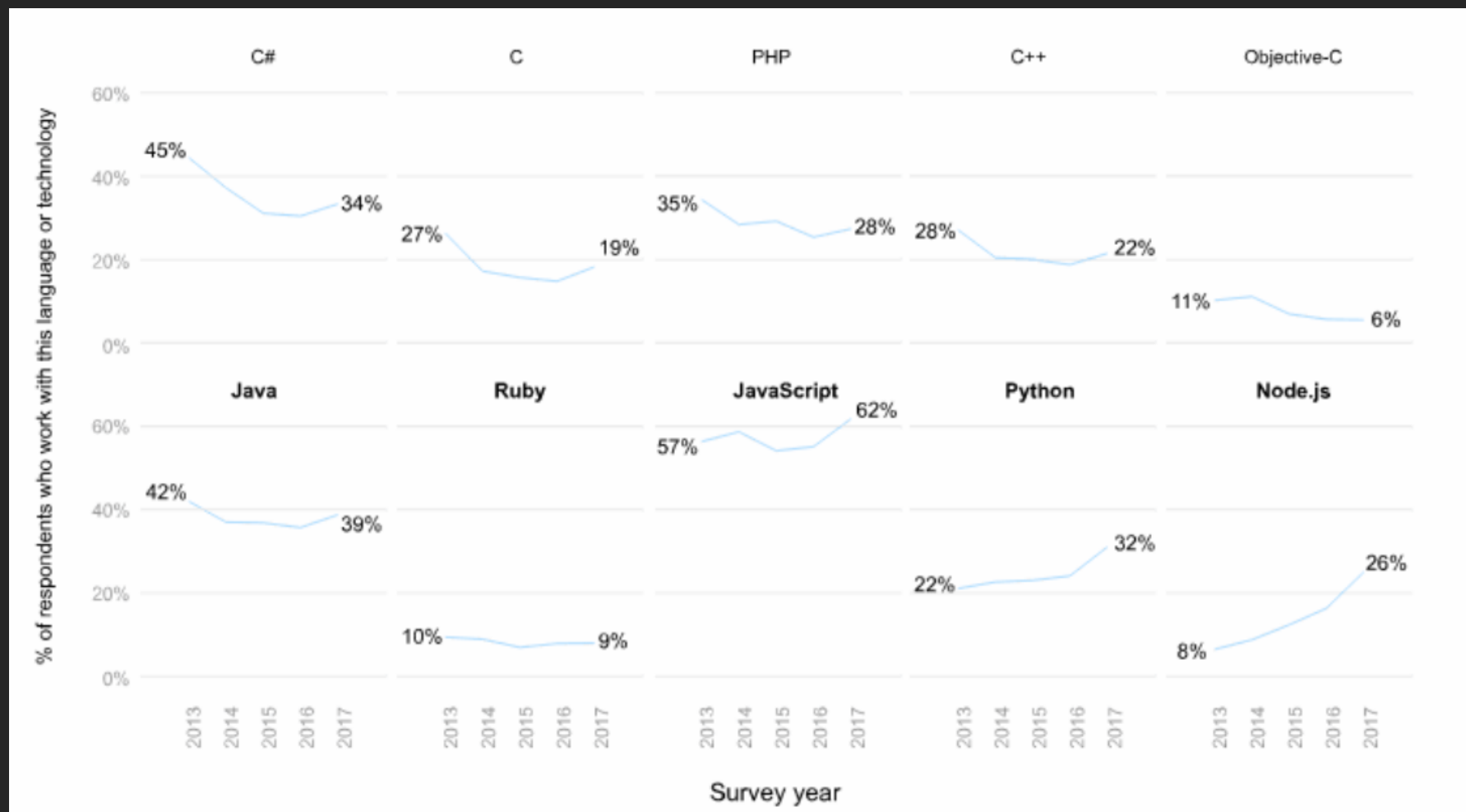
Node.js + 터미널 CLI 이용법

패키지관리툴(npm, yarn) 작업 자동화툴(grunt, gulp) 컴파일도구(babel..)

번들링툴(webpack) 등

Python & Javascript

스크립트 언어의 강세

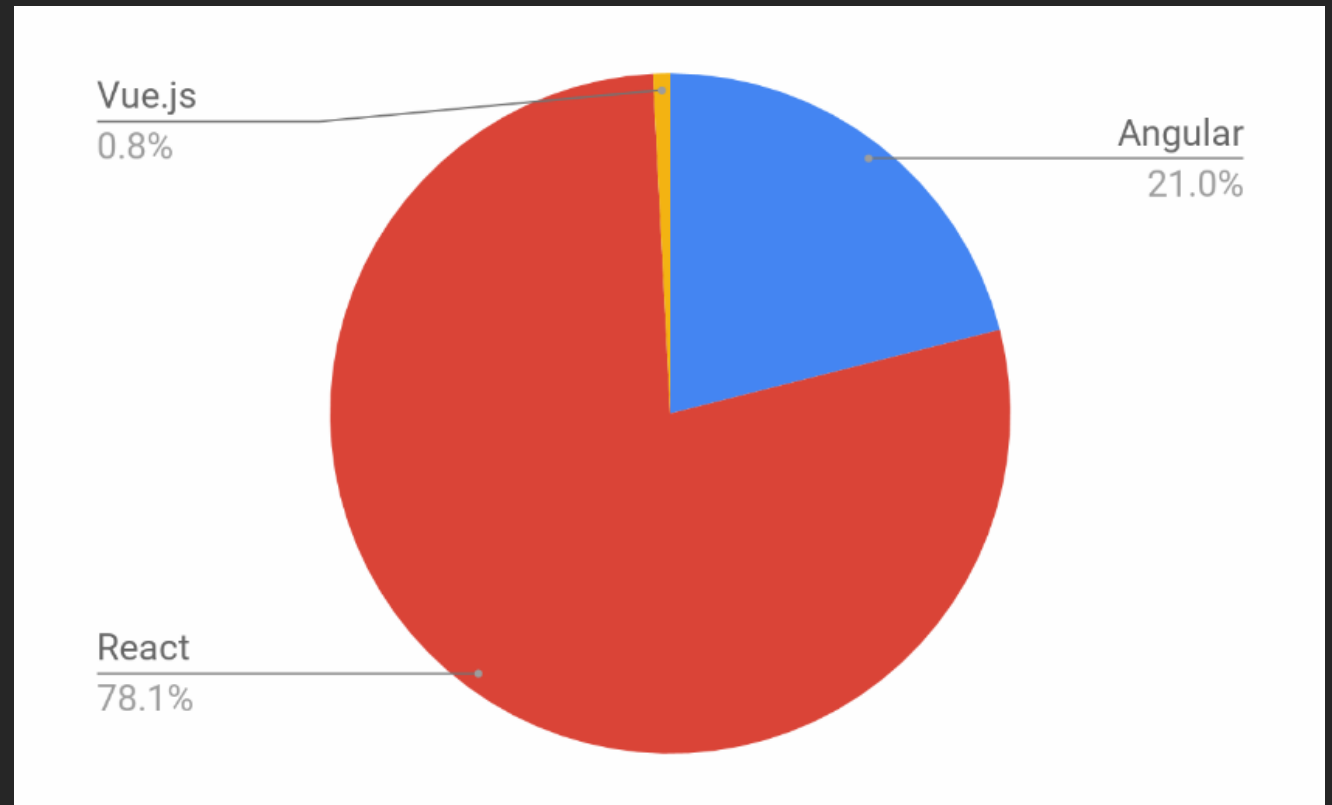


React & Angular & Vue

DOM (HTML) 조작을 쉽게 해주는 최신 라이브러리들이 대세

(jQuery는?)

미국 채용사이트 Job offer 통계



웹의 발전단계

정적 웹 -> 동적 웹 -> 웹을 앱처럼 쓰자

그냥 웹문서

쇼핑몰
SNS
블로그
게시판 등

쇼핑몰
SNS
블로그
게시판 등
+ UX혁명

- > Single Page Application을 이용해 seamless한 사용자경험
- > 구글 AMP (매우빠른 모바일웹)
- > Progressive Web App (웹을 앱처럼 사용하자; flipkart)
- > Hybrid app (HTML/CSS/JS로 iOS 앱을 만들자)

웹을 앱처럼 만드는 현상이 일어난 이유 :

1. Acquisition cost :

'Web 방문'이 'App 설치'보다 마케팅 비용이 10배 싸다.

2. Reach :

Web이 App보다 고객 풀이 2.5배 넓음.

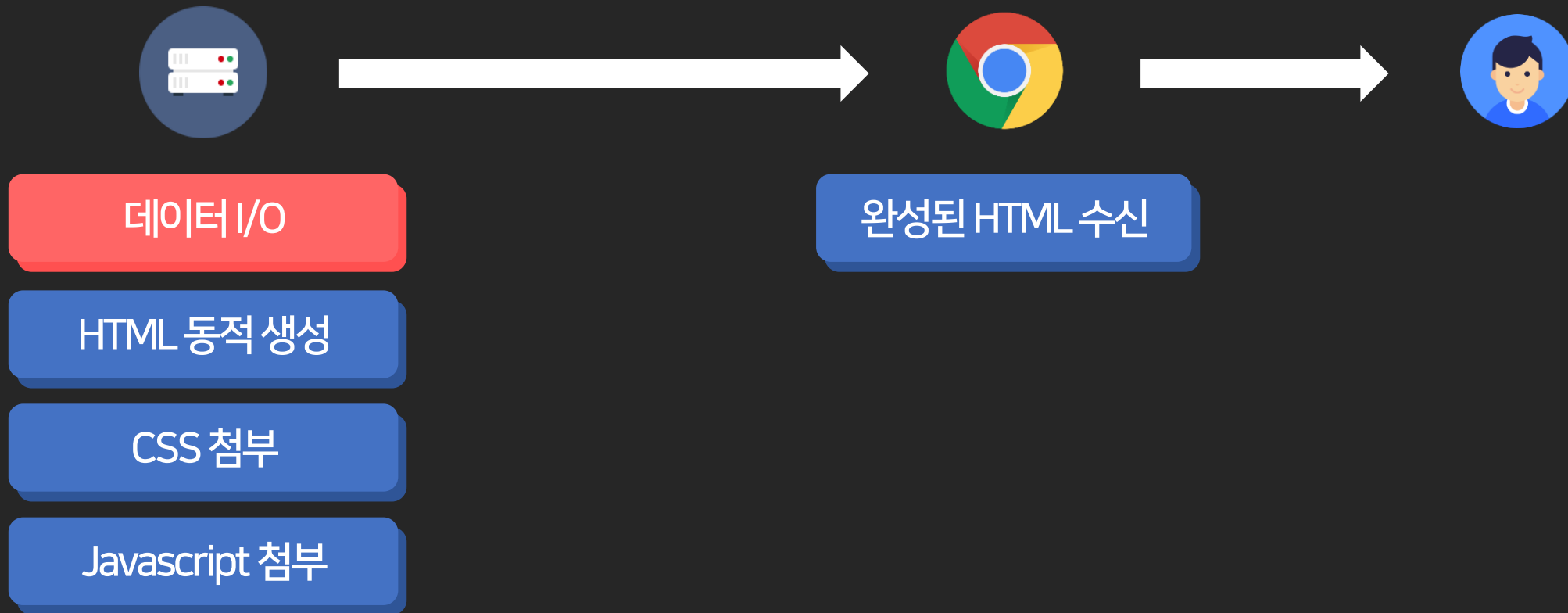
3. Conversion :

PWA로 구축한 사이트는 일반 사이트보다 구매전환율이 더 높음.

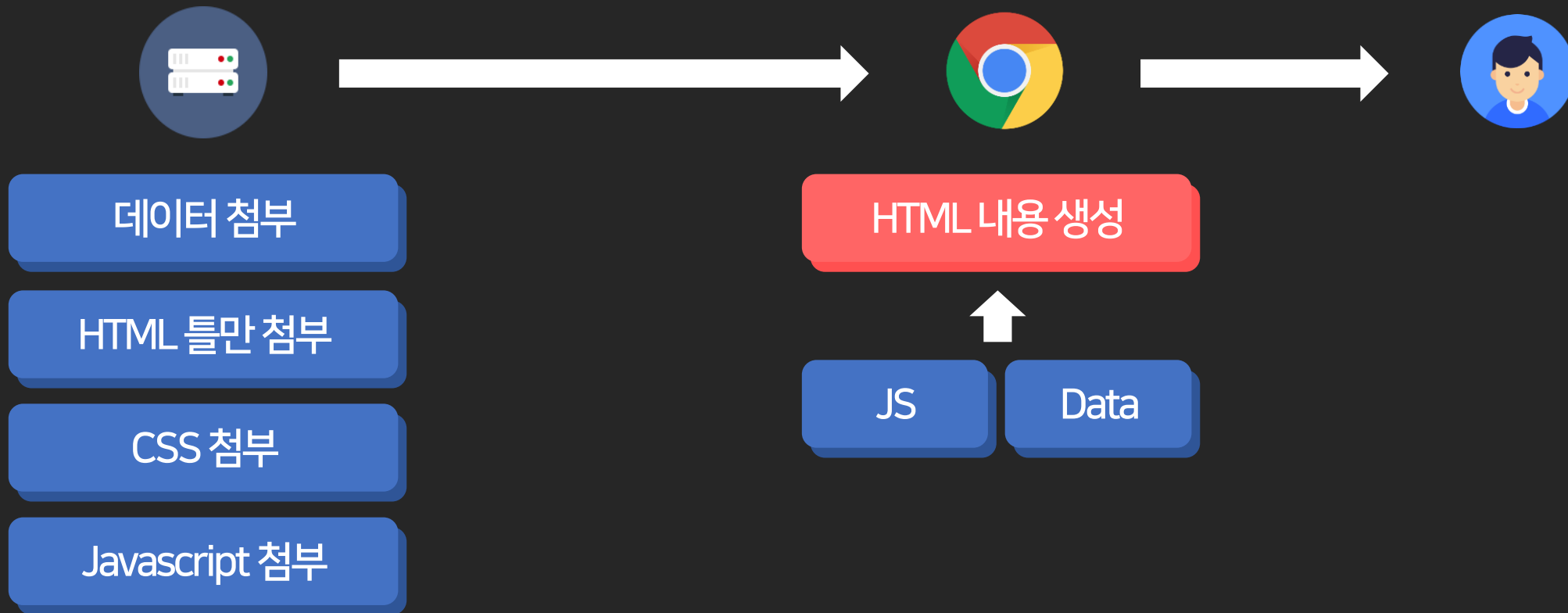
4. App store의 몰락 :

미국 스마트폰 사용자의 64%가 한달에 1개 이하의 앱을 설치.

웹의 구조 <사용자가 상품리스트 페이지를 요청함>



웹의 구조2 <사용자가 상품리스트 페이지를 요청함>





HTML은 무엇

- 마크업언어
- 웹문서의 글,그림,동영상,레이아웃을 만드는 언어

```

```

```
<p>Hello</p>
```

```
<div>
```

```
  <a>Click</a>
```

```
</div>
```

```
<div id="learnpic" class="jumbotron">
  <div class="yellowbox container">
    <p>안녕하세요</p>
    <p>뚝뚝뚝</p>
  </div>
</div>
```



CSS는 무엇

- 마크업언어를 꾸며주는 역할

```
  
<p class="red">Hello</p>  
<div>  
  <a>Click</a>  
</div>
```

```
.red {  
  color: red;  
}
```

JS

는 어디에 쓰나

- HTML 요소 변경
- 마우스/키보드/스크롤 이벤트 다루기
- 데이터 생성과 변경
- 서버와 통신



- 웹 UI요소 개발 (탭, 모달, 팝업 등)
- 데이터 바인딩
- 서버데이터 전송 및 수신
- 외부 API 가져다 쓰기

Think
Like
Code

프로그래밍 하는 법

문제를 푸는데

매우, 상세히, 논리적으로 풀면 됩니다.

아이스크림 먹는 법을 설명하시오

1.

2.

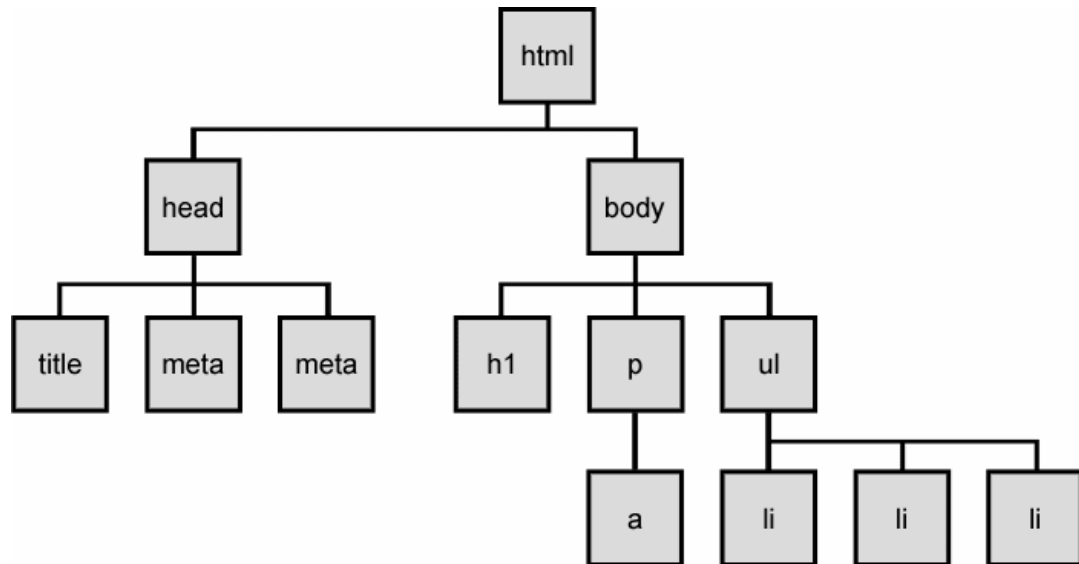
3.

탭(tab) 기능을 만드시오

1. 첫째 버튼을 누르면.. 어.. 탭내용이 보이고, 버튼에 하이라이트
- 2.
- 3.

HTML/CSS하던 습관으로 '감성적'으로 접근하면 망함
무조건 **이성과 논리**로 읽고 씁니다.

자바스크립트는 HTML문서를 이렇게 봅니다.



Document Object Model

= 그냥 **HTML의 집합**

= JS가 HTML을 보는 관점

Javascript 코드 작성 방식

입력

id로 <div>를 선택하고
class 속성을 가져온다



변경

새로운 class를 생성한다



출력

새로운 class를 박아넣는다

id="bluebox" 인 <div> 를 찾아서 새로운 class 를 추가해라

Javascript의 제1목적

= HTML 변경

흔한 Javascript 코드

```
<h1>What Can JavaScript Do?</h1>
```

```
<p id="demo">JavaScript can change every HTML content.</p>
```

```
<button onclick='document.getElementById("demo").innerHTML =  
"Hello JavaScript!'">Click Me!</button>
```

https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_inner_html

예제 여러 개

http://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_inner_html

https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_style

https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_lightbulb

JS

되돌리기 버튼을 만들어보자

https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_inner_html

프로그래밍 기본상식 정리

.어쩌구

한국어로 번역하자면 '~의' (혹은 메소드)

'글자'

"큰따옴표" '작은따옴표' 안에 글을 쓰면 문자열이 됩니다.

100

따옴표가 없는 숫자는 숫자입니다. (숫자에 따옴표가 붙으면 문자)

x = y

= 기호는 등호가 아니라 대입입니다. 오른쪽 값을 왼쪽에 대입.

웹문서 안의 특정 HTML 요소를 자바스크립트로 찾으려면

```
document.getElementById( 'red' )
```

getElementById라는 셀렉터를 이용합니다. (HTML의 ID속성으로 찾아줌)

Syntax

`console.log()`

찾은 요소가 정확한지 확인하려면 :

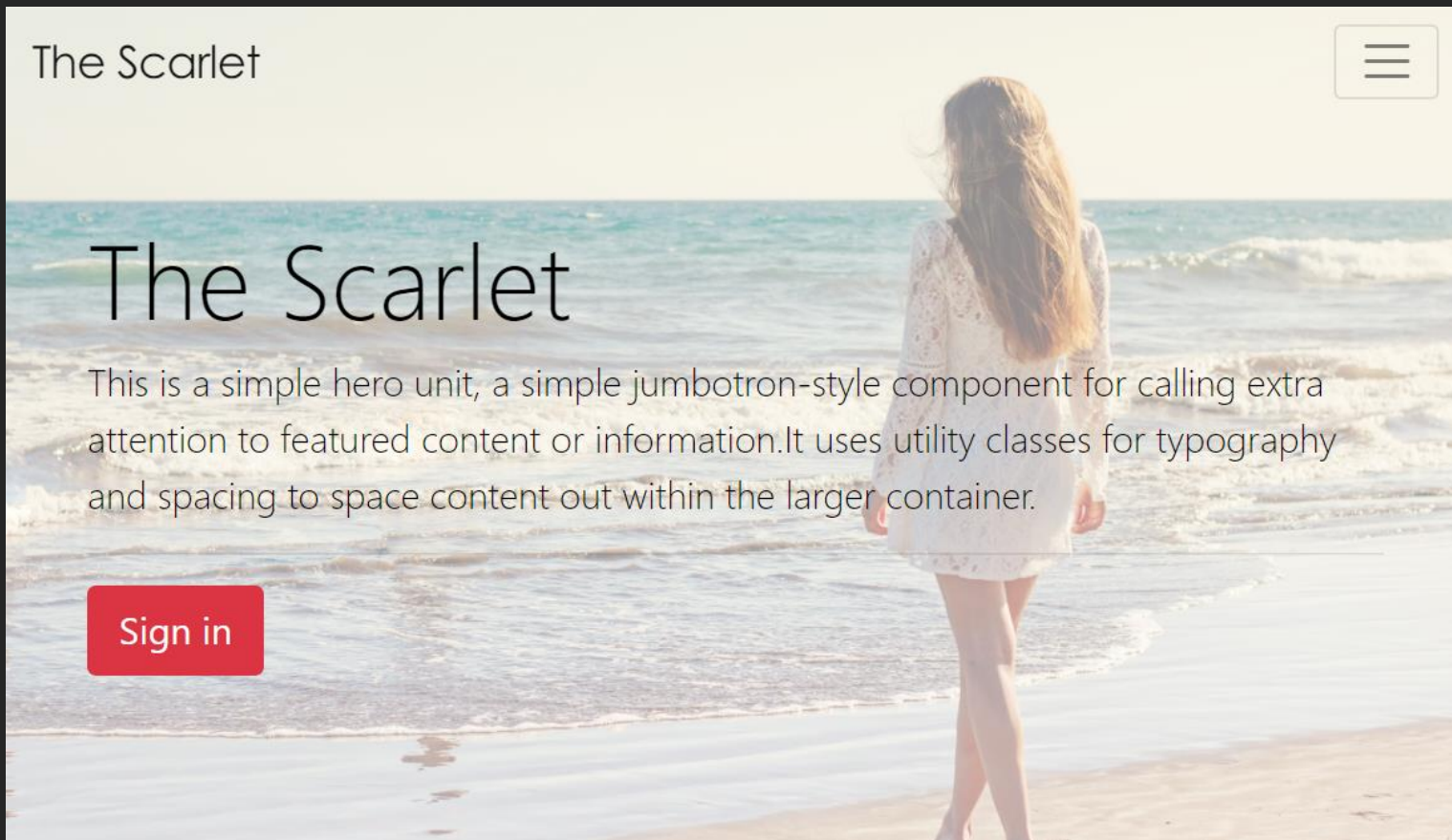
`console.log()`로 자바스크립트 콘솔창에 **로그**를 찍어봅니다.

프로젝트 준비 : Bootstrap 4

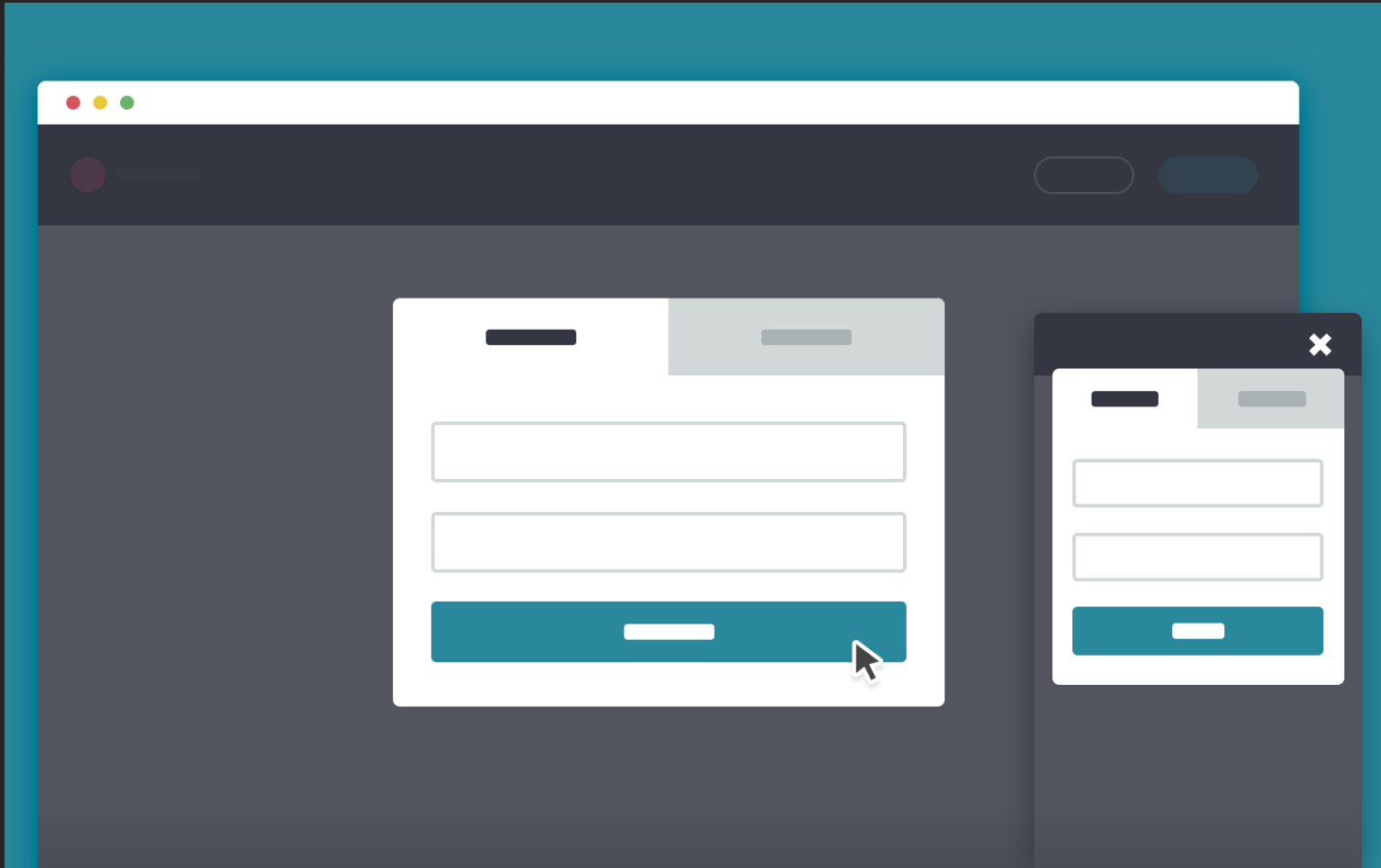
<https://getbootstrap.com>

JS

Project0 : 쇼핑몰 대문 + navbar 빠르게 만들기



JS Project1 : Modal 만들기 (로그인용)



Modal의 작동방식을 자세하게 설명해봅시다.

0. 모달창을 미리 만들어놓고 숨겨놓은 후에

1. 로그인 버튼을 누르면, 모달창이 뜸

1. 닫기버튼을 누르면, 모달창이 사라짐

2.

3.

...

이벤트 처리를 위한 <onclick=" "> 속성

```
<button onclick='document.getElementById("demo").innerHTML =  
"Hello JavaScript!'">Click Me!</button>
```

하지만 HTML상에 코드를 바로 집어넣는건..

마크업 언어가 복잡해집니다.

```
<button onclick='document.getElementById("demo").innerHTML =  
"Hello JavaScript!'">Click Me!</button>
```

함수의 역할 1 : 기능 묶기

코드 덩어리(기능)을 하나로 깔끔하게 묶고 싶으면 함수라는걸 사용합니다.

```
function 함수이름 () {  
    //묶을 코드  
}
```

```
function 피카츄() {  
    console.log('백만볼트')  
}  
  
function eat() {  
    return '음쓰'  
}
```

Modal cheating sheet

요소 잡는 법 : `document.getElementById('어쩌구')`

보이게 하는 법: `어쩌구.style.display = "block";`

함수로 묶는 법: `function showModal(){`
 `// 실행할 코드`
 `}`

다른 내용의 Modal을 여러 개 만들려면?

이용동의약관

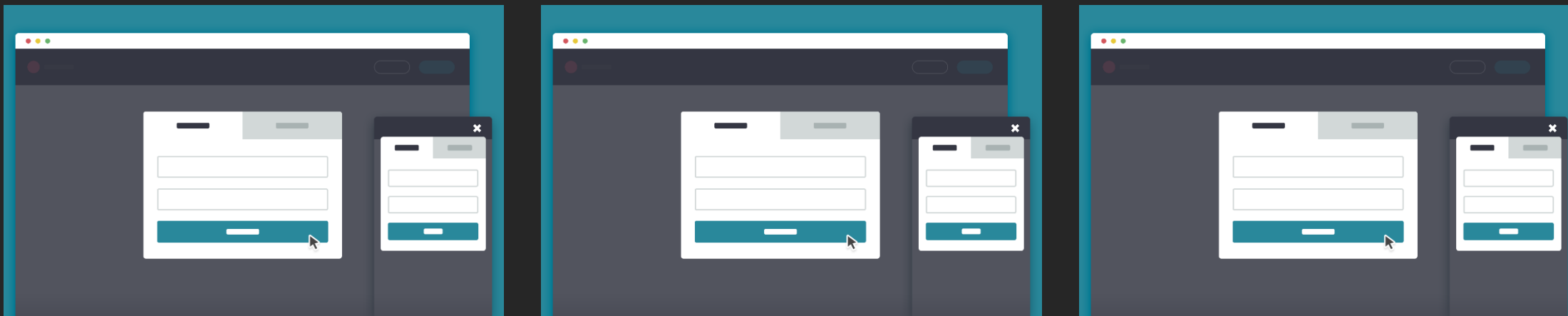
개인정보보호 약관

기타 각종 계약서

(초보) : 그냥 모달창 HTML을 세개 하드코딩하면 되겠죠?

(중수) : 함수의 인자를 이용해 다른 각각 기능을 가진 함수를 만들면 되겠죠?

다른 내용의 Modal을 여러 개 만들려면?



: 함수의 인자를 활용한 함수 기능 확장!

함수기능 1. 모달창 띄우기

함수기능 2. HTML 내용 샤라락 바꾸기

함수의 역할 2 : (I/O) 박스 만들 때

인자를 넣어 Input에 따라 Output이 달라지는 박스를 만들 수 있다.

```
function plus() {  
    console.log(10 + 20);  
}
```

```
plus();
```

```
function plus(a) {  
    console.log(a + 2);  
}
```

```
plus( 3 );  
plus( 12 );
```

.. 수학의 함수와 동일합니다. $f(x) = 2x$

함수의 역할 2 : 인자 (argument)를 이용한 기능확장 (I/O)

Input은 **argument**

Output은 **return**

```
function plus( a , b ) {  
    return a + b  
}
```

```
plus( 3, 5 );
```

Javascript 변수

= 데이터를 저장할 수 있는 공간!

```
var 차 = 'BMW' ;
```

```
var 차가격 = 30000 ;
```

```
var 차모델 = '520d' ;
```



변수명엔 - (dash)는 사용금지, 보통 소문자로 시작함, Camel case

Javascript 변수의 Hoisting

아직 변수를 만들지도 않았는데 변수가 있네?

```
console.log(name);    //변수 정의도 하기전에  
var name = 'John';    //변수를 사용해버리면? = 잘 작동됨.
```

변수는 1.선언 2. 값할당 두가지가 있는데,
자바스크립트는 변수의 선언을
전부 최상단으로 강제로 끌어와서 해석함.

ES6 : 변수의 종류와 Scope(범위)

var	Function-scoped	재선언 O	재할당 O
let	{Block-scoped}	재선언 X	재할당 O
const	{Block-scoped}	재선언 X	재할당 X

그리고 보통의 프로그래밍 언어라면
이 변수는 문자다! 숫자다! 라는걸 미리 직접 정해줘야하지만
자바스크립트 언어는 그런게 없습니다. 지가 알아서 정해줌.

여기까지 옛날 자바스크립트였습니다.

요즘 자바스크립트는

.addEventListener() (IE \geq 9)

.attachEvent() (IE $<$ 9)

Javascript `.addEventListener()`

```
element.addEventListener('click', function(){  
    //element 클릭 시 실행할 코드  
});
```

선택한 요소에 Element에 이벤트를 달 수 있음.

첫번째 인자로 click, mouseover 등의
이벤트 상황을 지정해주면 됩니다.

두번째 인자로 콜백함수() { }를 넣고 그 안에 실행할 코드를 넣습니다.

문법 : callback 함수

```
element.addEventListener('click', function () {  
    //element 클릭 시 실행할 코드  
});
```



(함수에 이름이 없음)

함수 안에 인자로 함수가 들어가는 형태를
교양있게 콜백함수라고 합니다.

흔히 자바스크립트에서 이벤트 발생 -> 기능동작! 이렇게
뭔가 순서대로 실행하고 싶을 때 많이 사용하는 패턴입니다. (거의 외우다시피 해야..)

문법 : callback 함수

```
element.addEventListener('click', showModal );
```

당연히 다른데서 정의한 함수도 집어넣을 수 있습니다.
훨씬 더 깔끔하죠?

어떻게 더 좋은 것인가

```
element.addEventListener('click', function(){  
    //모달창 보여주는 코드  
});
```

x개

```
<div onclick="showModal()" > </div>  
  
function showModal(){  
    //모달창 보여주는 코드  
}
```

1개

.addEventListener()와 콜백 함수를 배웠으니
modal button 코드를 refactoring 해보자

Modal

디버깅 팁

- `<script>` 위치 바꿔보기 (우선시 되는 자바스크립트는 `</body>` 바로 전에 넣는게 좋다)
- 콘솔창에 에러가 뜨면 감사해야 합니다. 에러가 무엇인지 구글에 자주 검색해보자
- `console.log()` 혹은 `alert()`로 중간중간 테스트해보는 습관
- 변수의 선언위치는 어디인지 체크 (하지만 함수나 변수 선언은 hoisting 됨)

JS

Project2 : 아이디를 입력하세요! (공백 검사)

성과 이름



이메일 주소



비밀번호



이름, 이메일, 비밀번호를 입력하세요

준비물 :

- Bootstrap Form 제작
- Form에 name속성으로 이름쓰기

법카 사용 허락 프로그램

```
function 법카결제() {  
    return '승인'  
}
```

```
var 오늘점심값 = 6000;  
법카결제();
```

직원들이 너무 많이 카드를 긁자
이사님은 5000원의 금액제한을 두게 되는데..

if / else / else if 로 조건을 만족하거나 만족하지 않을 때 코드 실행이 가능

```
if ( 참/거짓 ) {  
    //조건이 참일 때 실행할 코드  
} else if (조건문2) {  
    //조건2 참일 때 실행할 코드  
} else {  
}
```

```
if ( true ) {  
    console.log('정답');  
} else {  
    console.log('틀림');  
}
```


if / else 조건문에 사용할 비교/논리 연산자 (Operator)

부등호

>

<

>=

<=

등호

2 == '2'

2 === '2'

2 != '2'

!==

논리

&&

||

!()

연산자를 사용하면
true 아니면 false를 반환합니다.

```
if ( 3 > 1 ) {  
    console.log('정답');  
} else {  
    console.log('틀림');  
}
```

기타 수학 연산자 (Operator)

+	1+1	1+'1'	'1'+ '1'
-	5-4	5-'4'	
/	5/4	5/'4'	
*	5*4	5*'4'	

+기호만 좀 이상함. +기호는 문자위주계산

```
if ( 3 > 1 ) {  
    console.log('정답');  
} else {  
    console.log('틀림');  
}
```

오늘의 HTML 상식:

<form onsubmit="return false">

<form onsubmit="return true">

form의 onsubmit 속성 내에 return false, return true를 적어서
폼을 정상적으로 전송시키거나 막을 수 있습니다.

(중수용 참고사항: submit 이벤트를 처리하는 addEventListener내부에 e.preventDefault()를 달아도 똑같음)

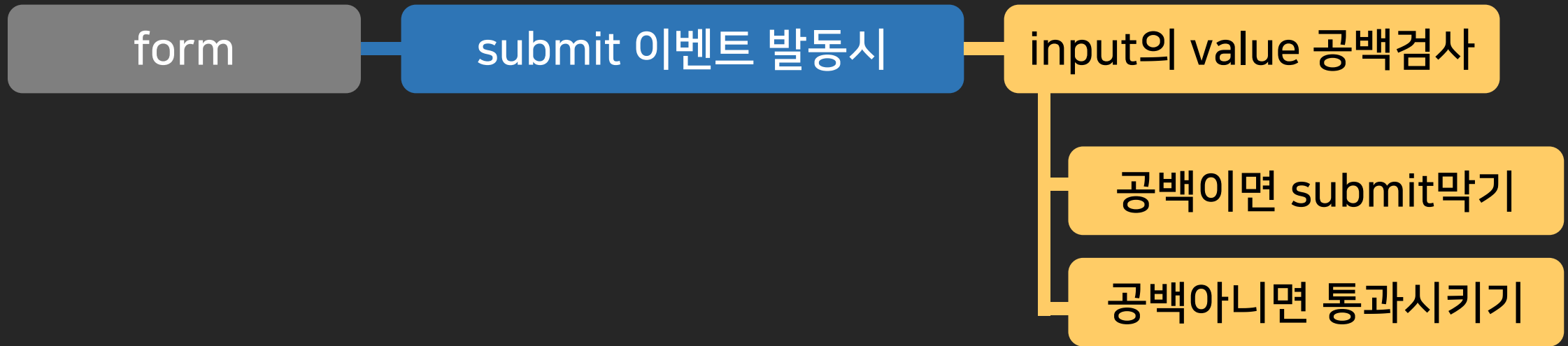
오늘의 HTML 상식2:

```
<form name="form1">  
  <input name="email">  
</form>
```

form태그나 input태그 안엔 **name**속성으로 이름을 자주 달아줍니다.

(나중에 Javascript로 폼의 value를 가져올 때 코드가 훨씬 짧아집니다.
document.form1.email.value 이런식)

폼 공백검사 pseudo 코드:



응용 : 빈칸일때 경고 div박스를 띄우려면?

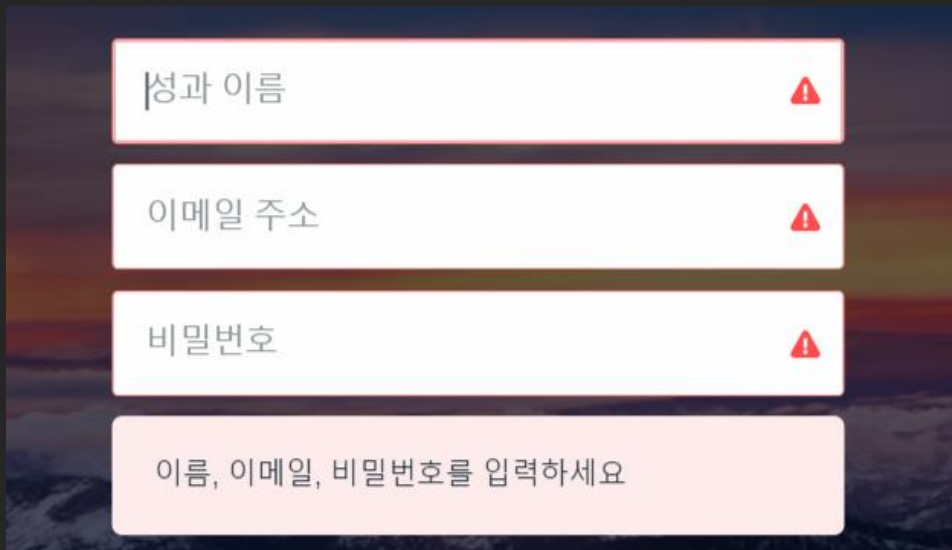
form

onsubmit 이벤트 발동시

input의 value 공백검사

공백이면 submit막기 + div

공백아니면 통과시키기



성과 이름

이메일 주소

비밀번호

이름, 이메일, 비밀번호를 입력하세요

응용 : 폼 전송시가 아니라 폼 입력중일 때 경고를 주려면?

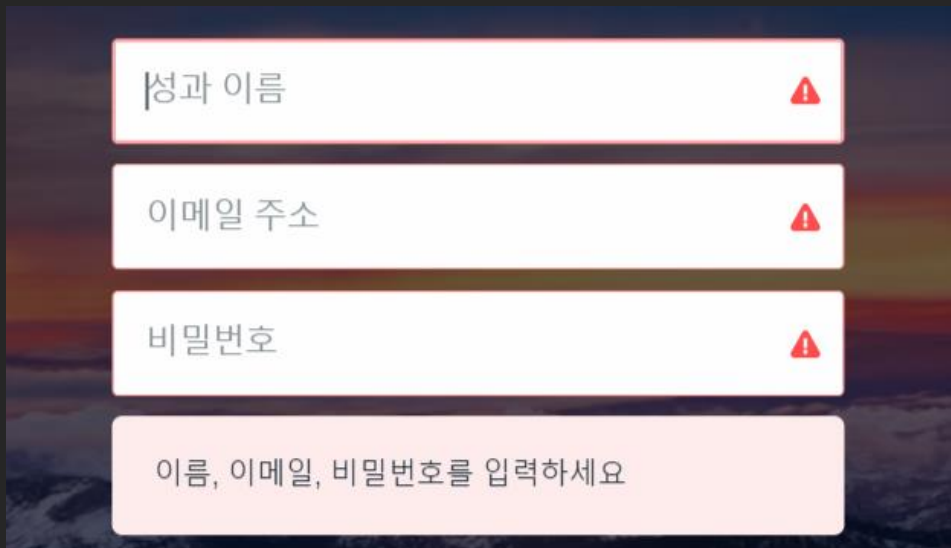
input 요소

?? 이벤트 발동시

input의 value 공백검사

공백이면 submit막기 + div

공백아니면 통과시키기 - div



성과 이름

이메일 주소

비밀번호

이름, 이메일, 비밀번호를 입력하세요

응용하면

- 빈칸 검증
- 이메일형식이 맞는지 검증
- 최대 최소값 검증
- 비밀번호에 영어대문자가 있는지 검증

.....

JS Project2.5 : 승용차 브랜드명 집어넣기

Object 데이터를 HTML에 넣어보자

Syntax

Javascript 변수

= 데이터를 저장할 수 있는 공간!

```
var car = 'BMW' ;  
var carPrice = 30000 ;  
var carModel = '520' ;
```



Array 자료형

Javascript는 Array라는 데이터 저장형식을 제공합니다.
한 개의 변수(variable)에 데이터 여러개를 저장하기 위해 씁니다. 끝

```
var car = [ 'BMW', 520, 'white' ];
```



Object 자료형

역시 한 개의 변수(variable)에 데이터 여러개를 저장하기 위해 씁니다.

```
var car = {  
  type : 'BMW',  
  model : 520,  
  color : 'white'  
};
```



```
{ key : value, key : value ...}
```

key와 value 형태로 저장합니다.

Array vs Object 자료출력 방법

```
var car = [ 'BMW', 520, 'white' ];
```

→ `car[1]`

```
var car = {  
  type : 'BMW',  
  model : 520,  
  color : 'white'  
};
```

→ `car.model`
`car['model']`

승용차 브랜드명 집어넣기

`var cars = ?`

1. 현대, 소나타, 회색
2. BMW, 520d, 검정색



```
<div>  
  <h4>브랜드명1 </h4>  
  <p>모델명1</p>  
  
  <h4>브랜드명2 </h4>  
  <p>모델명2</p>  
</div>
```

Array 다루기

```
var data = ['apple', 'pear', 'grape']
```

자료 추가 : data.push('melon')

array에 array추가한 새 array : data.concat()

맨 뒤 자료 삭제 : data.pop()

맨 앞 자료 삭제 : data.shift()

중간에서 잘라 새 array생성 : data.slice()

중간에 데이터 추가 : data.splice()

Object 다루기

```
var obj = { key1:'apple', key2:'grape' }
```

자료 추가 : obj.key3 = 'melon'

자료 추가2 : obj['key3'] = 'melon'

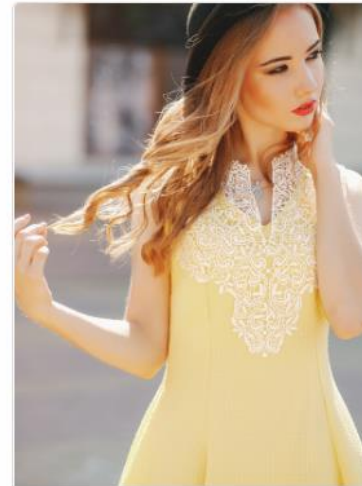
자료 삭제 : delete obj.key2

자료 삭제2 : delete obj['key2']

JS Project3 : 데이터 바인딩

Object 데이터를 HTML에 넣어보자

```
var list = [{  
  id: 1,  
  name: "blossom",  
  price: 299,  
  img: "img/pic1.jpg",  
  title: "Blossom dress",  
  text: "Do you like some random  
  for you."  
},
```



Blossom dress

Do you like some
random yellow stuff?
Here's one for you.

from \$299

Details



Springfield shirt

This shirt is from
Massachusetts, and we
don't mean any
springfield bolt-action
rifles.

from \$99

Details



JS

Project3 : 데이터 바인딩

서버언어로 하면되는데.. 왜 굳이 자바스크립트한테 시키냐면..

웹앱 형식의 사이트를 만들 수 있기 때문에!

<https://www.flipkart.com/>

Syntax

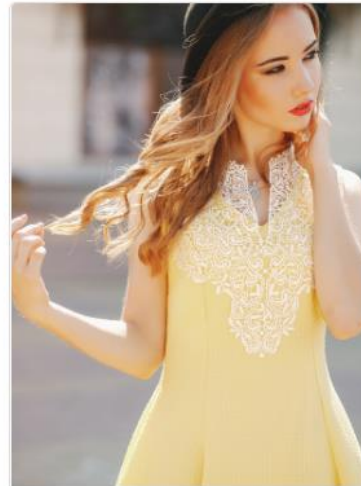
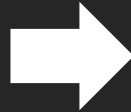
Javascript `querySelector`

```
.querySelector('#div li.list')
```

```
.querySelector('div.class')
```

Product 리스트에 데이터 바인딩을 해봅시다

```
var products = [  
  {  
    id: 1,  
    price: 299,  
    img: 'img/pic1.jpg',  
    title: 'Blossom Dress',  
    text: "Do you like some random yellow stuff?  
Here's one for you."  
  },  
  {  
    id: 2,  
    price: 199,  
    img: 'img/pic1.jpg',  
    title: 'Springfield Shirt',  
    text: "Stole it from my dad."  
  },  
  {  
    id: 3,  
    price: 399,  
    img: 'img/pic3.jpg',  
    title: 'Black Monastery',  
    text: "The best beer comes from the monastery."  
  }  
];
```



Blossom dress

Do you like some
random yellow stuff?
Here's one for you.

from \$299

Details



Springfield shirt

This shirt is from
Massachusetts, and we
don't mean any
springfield bolt-action
rifles.

from \$99

Details



Product 리스트에 데이터 바인딩을 해봅시다

1. 데이터 집어넣을 <HTML> 작성
2. Array / Object에서 원하는 데이터 뽑기
3. <HTML>에 데이터 집어넣기

JS

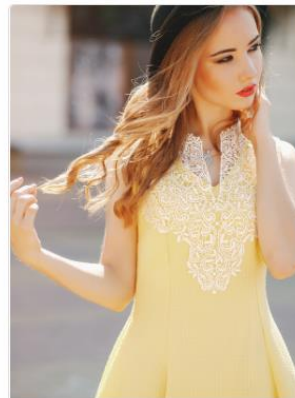
Project 3.5 : 상품 정렬 기능

'가격순 정렬 버튼'을 누르면 가격 순으로 정렬됨.

가격 정렬

이름 정렬

XX 정렬



Blossom dress

Do you like some random yellow stuff? Here's one for you.

from \$299

Details



Springfield shirt

This shirt is from Massachusetts, and we don't mean any springfield bolt-action rifles.

from \$99

Details



Array 정렬하기 : sort()

```
var array = [2, 3, 1, 5, 4];  
array.sort(function(a,b){  
    return a - b  
});
```

array 정렬함수 :

array내부의 두개의 값(a,b)을 뽑아 비교하는데,
a - b == 음수가 나오면 a가 앞으로
a - b == 양수가 나오면 a가 뒤로

Array 정렬하기 : 그럼 문자 정렬은?

```
var array = ['가다', '가나', '라', '하', '다'];  
array.sort();
```

```
array.sort(function(a,b){  
    if ( a < b ) {  
        return 1  
    } else if ( a > b ) {  
        return -1  
    }  
});
```

Array 전체를 규칙에 따라 바꾸기 : map() (역시 참고)

```
var array = [1, 2, 3, 4, 5];  
var array2 = array.map(function(a){  
    return a * 2  
});
```

모든 숫자에 2를 곱한
새로운 array2를 만든다.

//ES6

```
var array = [1, 2, 3, 4, 5];  
var array2 = array.map( a => a * 2 );
```


Array 데이터 필요한 것만 고르기: filter()

```
var array = [1, 2, 3, 4, 5];  
var array2 = array.filter( a => a > 2 );
```

함수에 따른
새로운 array2를 만든다.

Javascript 이벤트 동작방식 심화

난 버튼을 눌렀는데

```
<div>
```

```
<div>
```

```
<button>
```

```
</div>
```

```
</div>
```



button 누른건가요
div 누른건가요

```
<div>
```

```
<div>
```

```
<button>
```

```
</div>
```

```
</div>
```



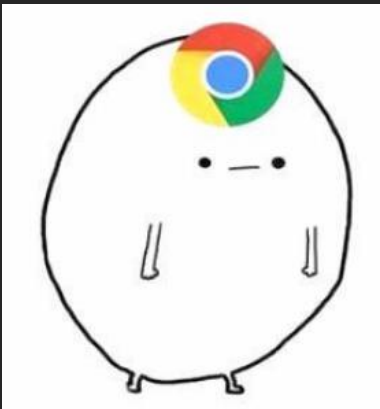
div누른지 button누른지
내가어찌알어
그냥 다 클릭한걸로
치셈



버튼을 클릭 시 상위 div요소들도 클릭을 감지하는 예제

<https://codepen.io/codingapple/pen/wmjOvR>

크롬이는 클릭이 일어나면
부모 <tag>들을 전부 다 탐색하면서 물어봅니다.



“button씨 클릭 이벤트 처리하세여?”
“div씨 클릭이벤트 처리하세여?”
...

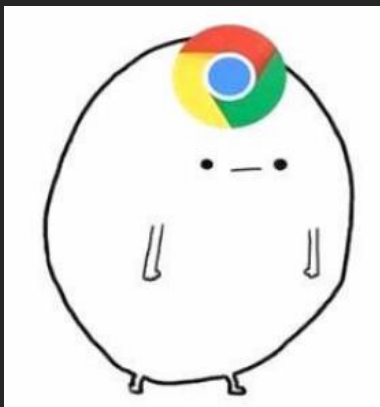
```
<div>
```

```
<button>
```

그런데 물어보는 순서가 있음.

<button>을 눌렀을 때

1. div -> button 순으로 물어보거나 (부모먼저)
2. button -> div 순으로 물어보거나 (자식먼저)



1. 부모님 먼저
2. 자식 먼저

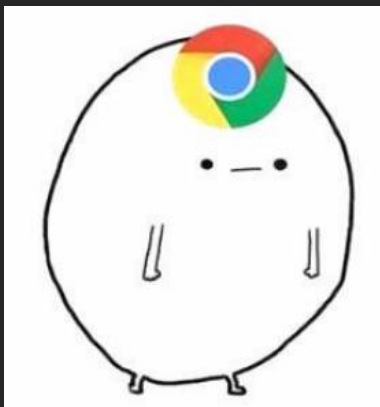
```
<div>
```

```
<button>
```

그런데 물어보는 순서가 있음.

<button>을 눌렀을 때

1. div -> button 순으로 물어보거나
2. button -> div 순으로 물어보거나



1. 부모님 먼저 : capture

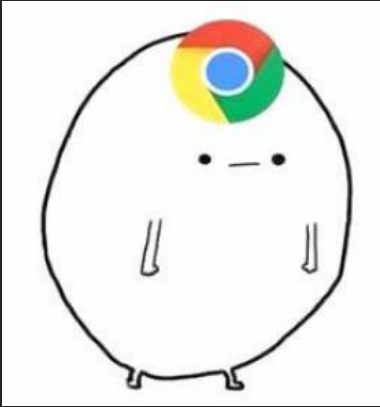
2. 자식 먼저 : bubble

```
<div>
```

```
<button>
```

모던 브라우저에선

button -> div 순으로 자식먼저 물어보는 **bubble** 형태가 디폴트



```
<div>
```

```
<button>
```


이벤트를 여러 개 달려면 Capturing / Bubbling을 이해하자

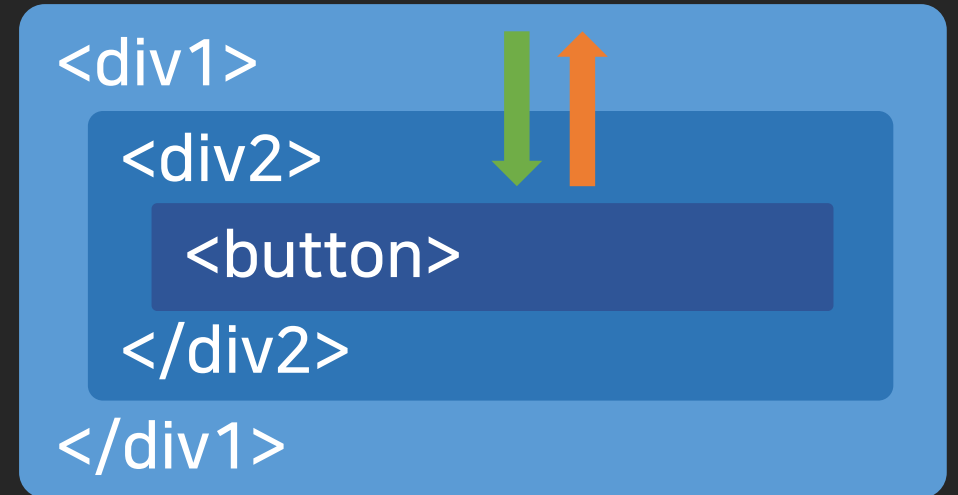
브라우저는 Event가 일어나면

capture div1, div2, button 순으로 event를 체크하기

bubble button, div1, div2 순으로 event를 체크하기

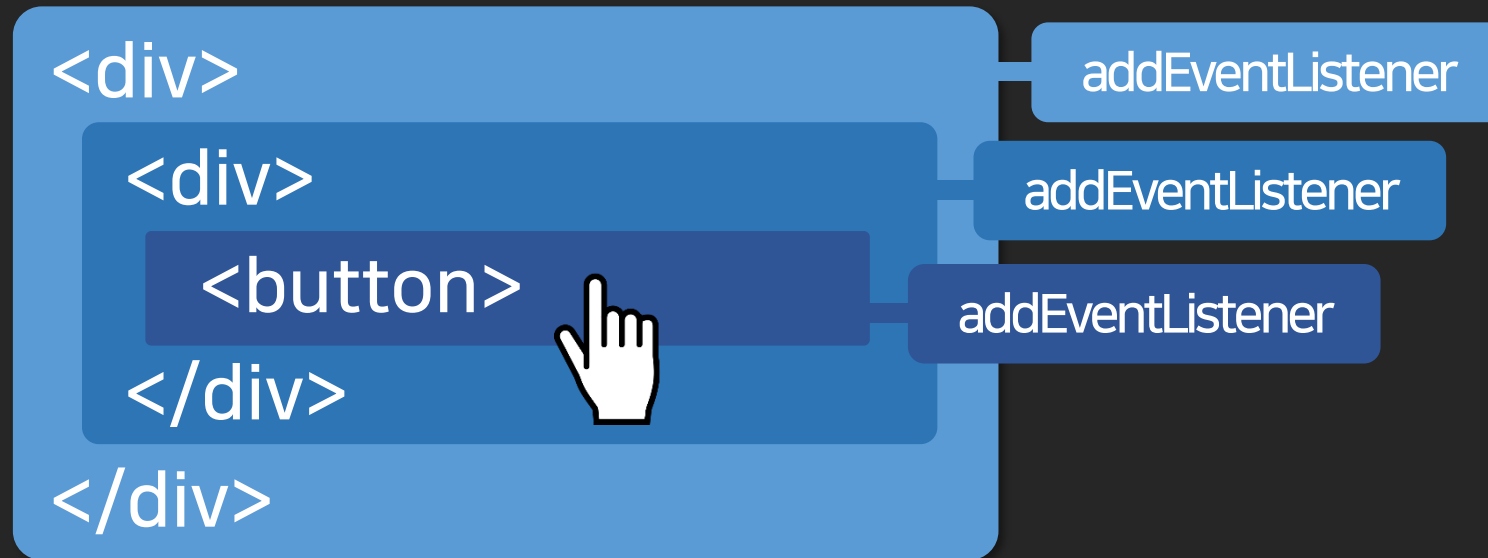
둘 중 하나의 방법으로 이벤트를 체크함

.addEventListener()로 이벤트를 달면 **bubble** 형태로 이벤트를 체크함.
(capturing은 잘 안써요)



그니까 개발할 때

이벤트 핸들러를 계속 달면 bubbling 때문에 분명 문제생기겠죠?



이벤트 bubbling을 중단하려면

여러가지 HTML요소에 event를 많이 달아줄 경우

bubbling 때문에 혼선이 오게 됨.

event가 상위 요소까지 퍼지는 bubbling 을 막으려면?

```
.addEventListener('click', function(e){  
    e.stopPropagation();  
});
```

event 객체 안에있는
event bubbling을 막는 함수

인자를 아무거나 넣으면
event 객체를 다룰 수 있음

이벤트 `bubbling`을 중단하려면

디버깅 팁

`e.currentTarget` 은 현재 이벤트가 달린 곳을 알려줌

`e.target` 은 처음 실제 클릭한 곳을 알려줌

- 각각 출력해보면 나중에 디버깅 쉽게 가능

이벤트 bubbling을 중단해보자

버튼을 클릭 시 상위 div요소들도 클릭을 감지하는 예제

<https://codepen.io/codingapple/pen/wmjOvR>

변태처럼 **capturing**을 쓰고싶다면

```
.addEventListener('click', function( ){  
    doSomething()  
}, true);
```

← true 넣으면 **capture** 식으로 이벤트를 체크함.

Bubbling응용 :
배경을 눌러서 모달창 닫기

지금 어떤 요소를 클릭했는가

```
.addEventListener('click', function(e){  
    e.stopPropagation();  
});
```

event 객체 안에있는
event bubbling을 막는 함수

인자를 아무거나 넣으면
event 객체를 다룰 수 있음

```
.addEventListener('click', function(e){  
    e.target;  
    e.currentTarget;  
});
```

- 이벤트가 발동한 HTML 요소
- 현재 eventListener 달린 HTML 요소

배경을 눌러서 모달창 닫기

이벤트 리스너 - 만약, 실제 클릭한 Element가 배경일 경우 - 모달창 닫기

Misc Javascript 형변환

숫자끼리 + - 는 잘 연산하는데
문자가 들어가면 복잡해집니다.

+는 문자를 만나면 문자화되고
-는 문자를 만나면 숫자가됩니다.



The Practical Dev
@ThePracticalDev



1 + 2

👉 3

1 + "2"

👉 "12"

1 - "2"

👉 -1

1 + 2 + "3"

👉 "33"

"1" + 2 - "3"

👉 9

1 - 2 + "3"

👉 "-13"

1 - 2 - "3"

👉 -4

웹사이트 발행 및 마케팅 업무

- 구글 & 네이버 웹마스터 도구로 사이트 등록
- 구글 애널리틱스로 사이트 접속자 분석
- 검색엔진 최적화 (SEO) 작업
- 사이트 배너광고는 Adwords
- 사이트 광고달기는 Adsense
- 마케팅을 위한 facebook/google 추적 코드 심기

Misc 개발자라면 알아야할 웹용량 최소화

1. minified JS & CSS (+automation task)
2. 이미지 최적화 (Progressive JPG, Sprite)
3. 한글폰트 사이즈 축소
4. 서버에게 request 최소화
5. 콘텐츠발행 + 광고수입 사이트를 위한 AMP
6. Bootstrap grid only
7. Chrome Audit tools

Misc

외부 API 사용한 기능개발

Facebook 개발자 도구 - 댓글기능 만들기

<https://developers.facebook.com/docs/plugins/comments>

네이버 개발자 도구 - 지도API 이용

<https://developers.naver.com/docs/map/javascriptv3/>

Misc

Stackoverflow 프로그래머 통계 모음

<https://insights.stackoverflow.com/survey/2016#work-salary>