

# Contents

about	01
layout	02
structure	03
main	04
media	05
gallery	06
location	07
join	08

# about

title: JEEP

url: <https://fmdlivehj.github.io/portfolio-business/index.html>

skill: scss, js

# layout

- 반응형 미디어 쿼리

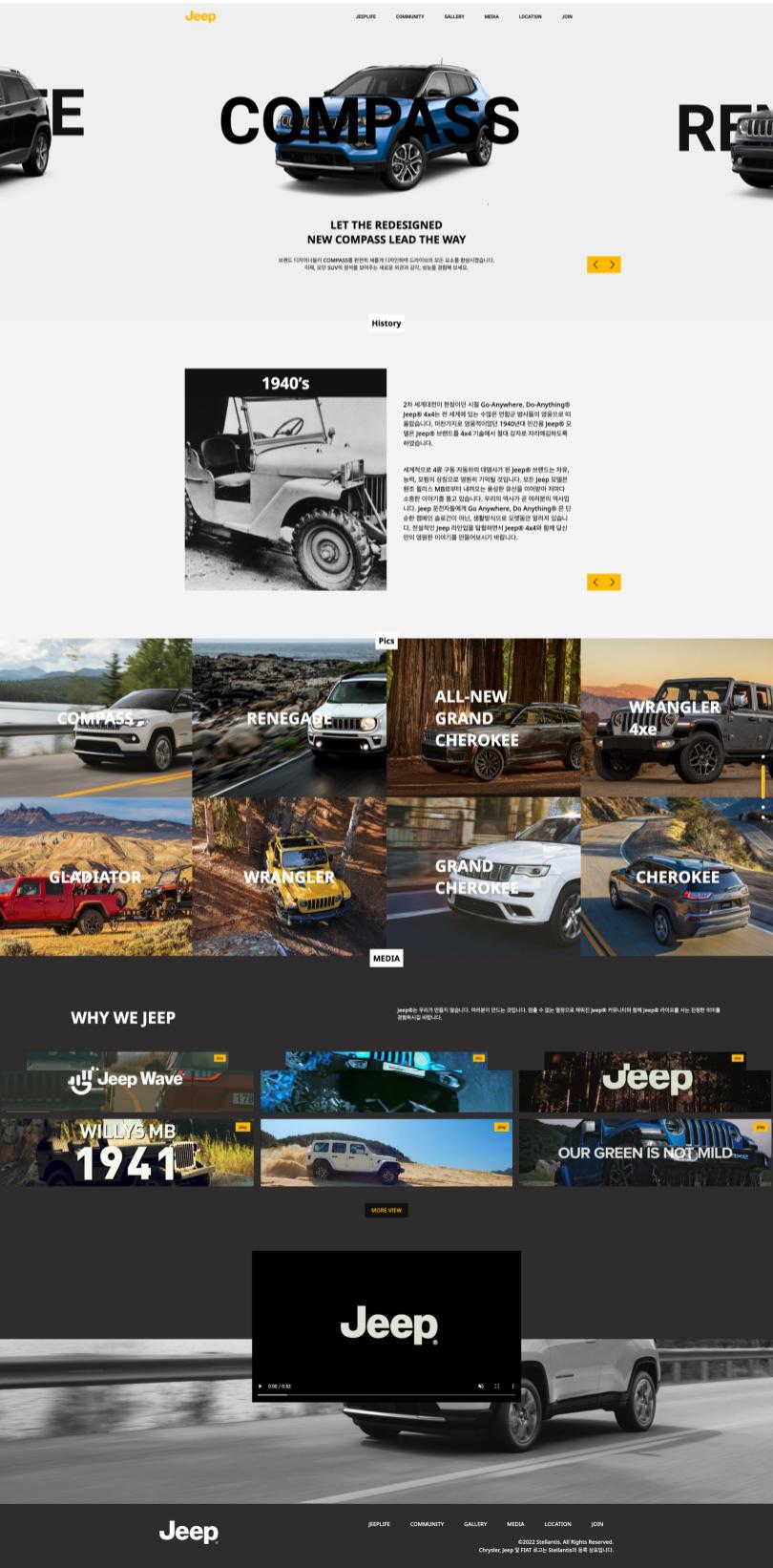
## 반응형 구간 설정

\$tablet : 1200px;

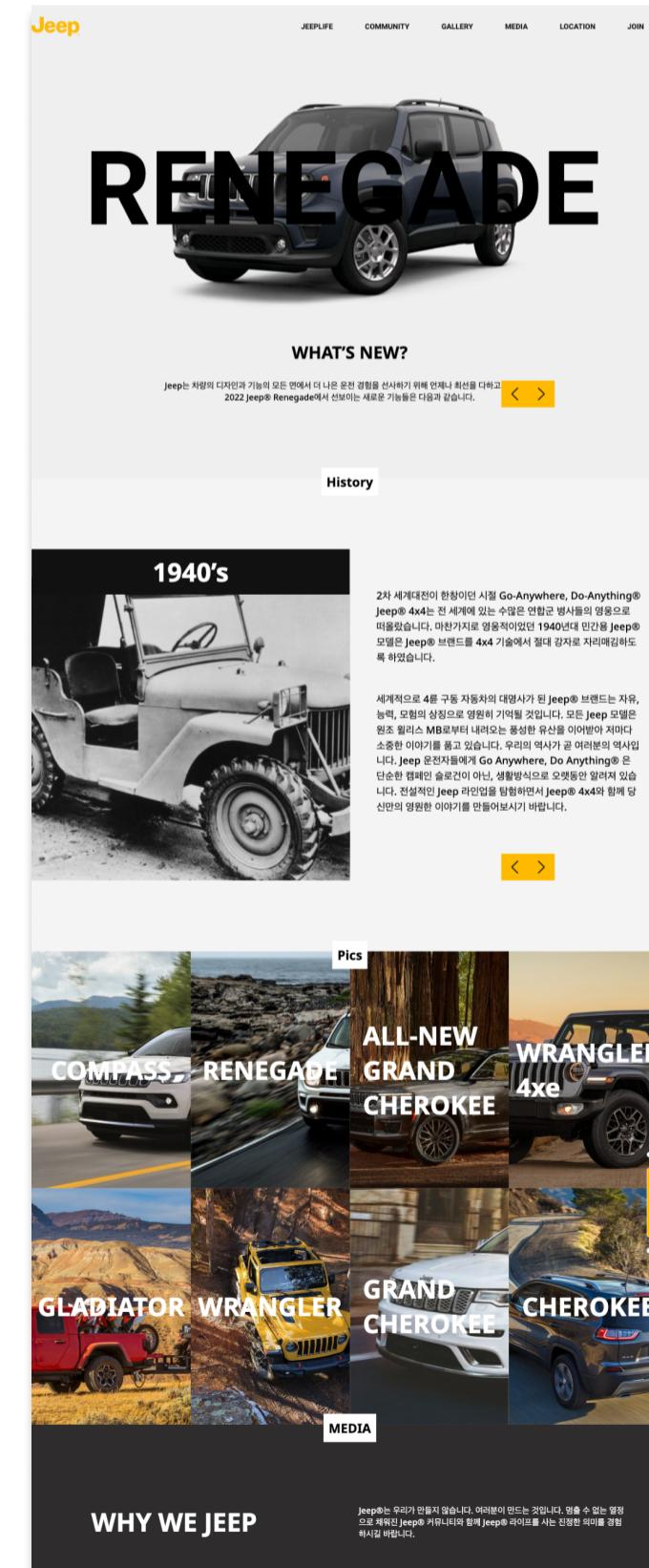
\$mini : 900px;

#mobile : 539px

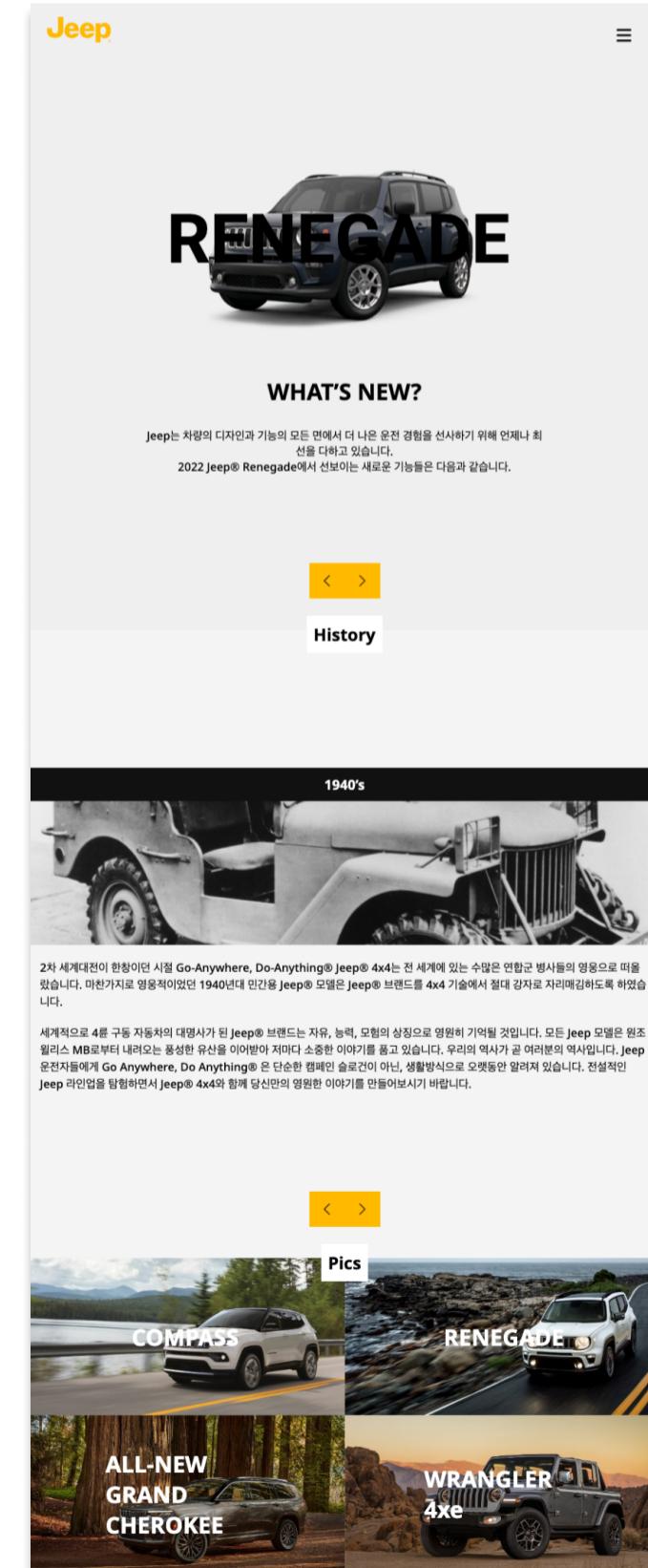
WEB



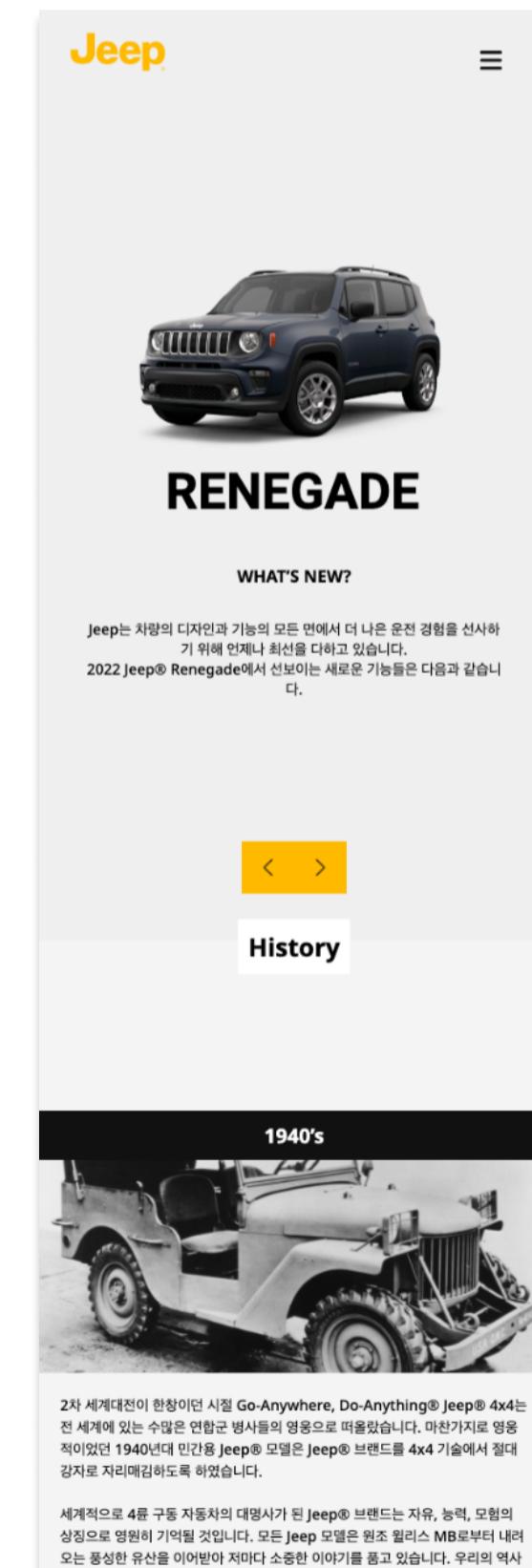
TABLET



MINI



MOBILE



# structure

- 폴더 구조



css



img



js



scss



video



about.html



community.html



gallery.html



index.html



join.html



media.html



result.html



css



img



video

style.css  
sub\_layout\_org.css

.jpg  
.png

.mp4



scss

\_about.scss  
\_community.scss  
\_gallery.scss  
\_join.scss  
\_location.scss  
\_media.scss  
\_reset.scss  
\_sub\_layout.scss  
style.scss



js

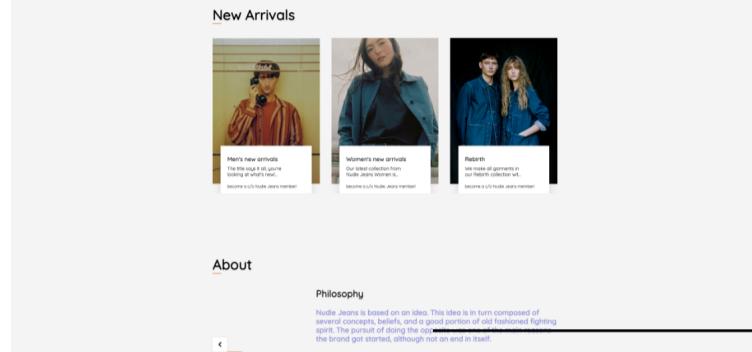
<b>JS</b>	about.js	<b>JS</b>	location.js
<b>JS</b>	anime.js	<b>JS</b>	main.js
<b>JS</b>	community.js	<b>JS</b>	media.js
<b>JS</b>	gallery.js	<b>JS</b>	sub.js
<b>JS</b>	join.js		

# main

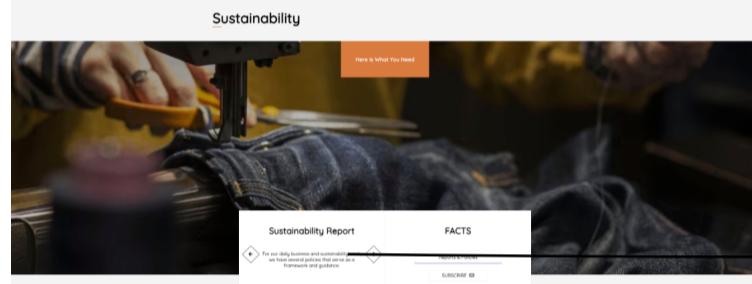


cookie popup

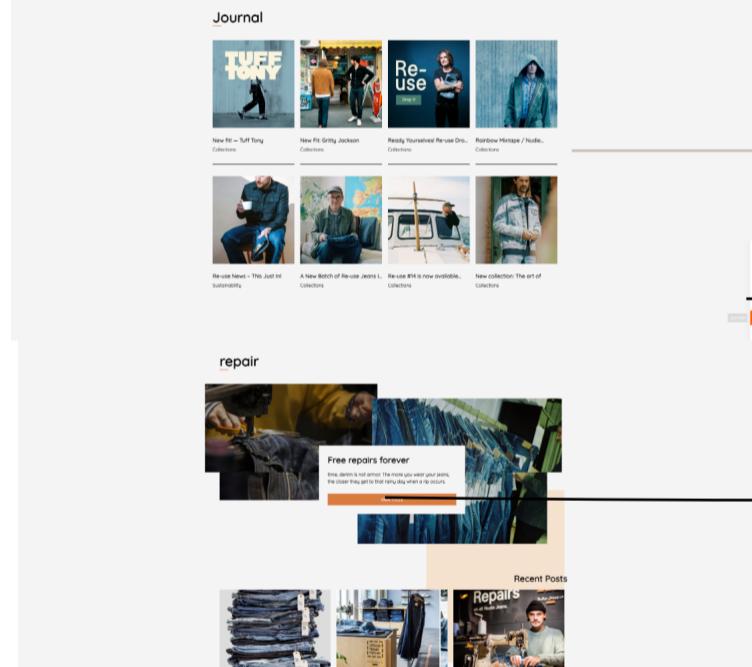
swiper slider



custom slider

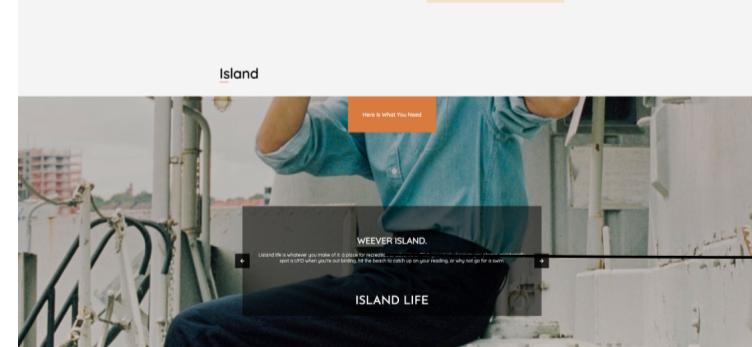


swiper slider

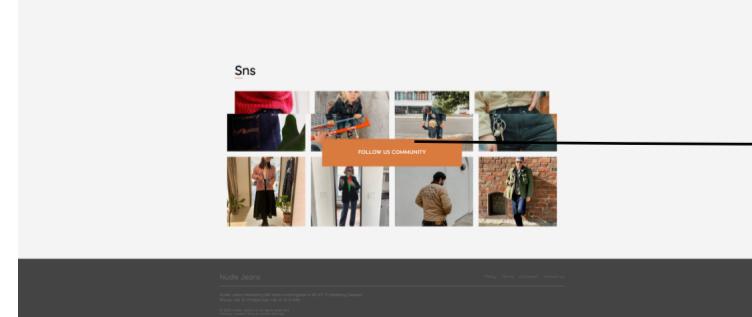


flickr api

scroll



swiper  
slider



gsap

- cookie popup

new Date() 생성자 사용  
브라우저 Application에  
쿠키생성

- swiper slider

swiper slider 플러그인 사용  
loop, speed , pagination 사용

- custom slider

Anime.js 사용

- flickr api

flickr api 가져옴  
검색버튼 클릭해서 callData 호출  
fetch json으로 flickr 데이터 가져옴

- scroll

각 section 에 높이값을 정해주고  
높이값의 1/3 에 스크롤이 왔을때  
on 클래스 추가

- gsap

gsap 플러그인 사용  
scale, toggle효과 추가

# subpage: media

## HTML

```
<div class="vCon">
  <div class="vid_con"></div>
</div>
```

## JS

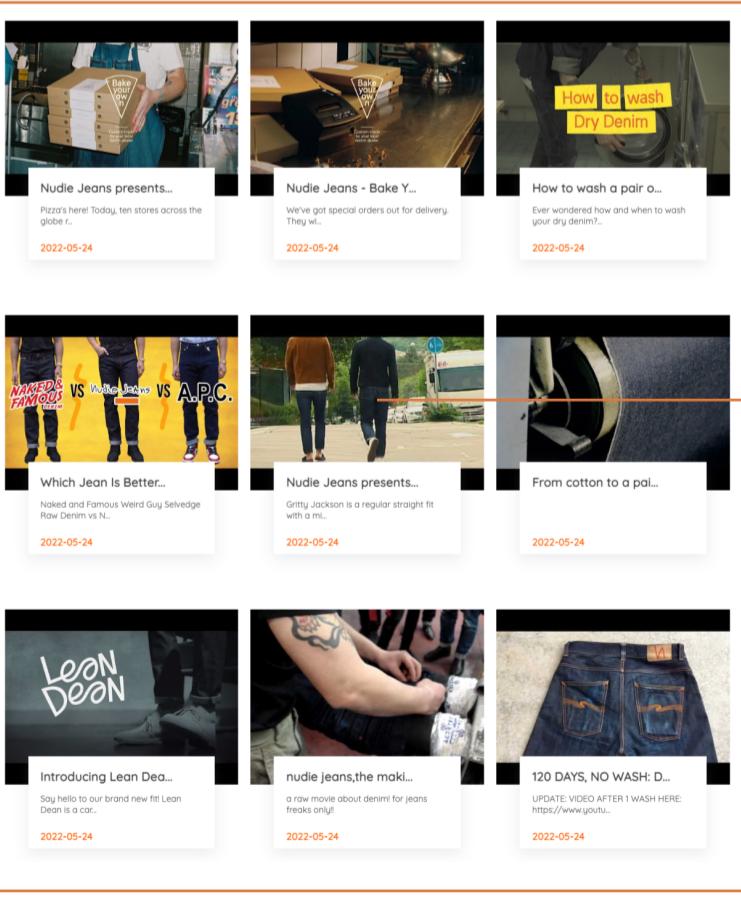
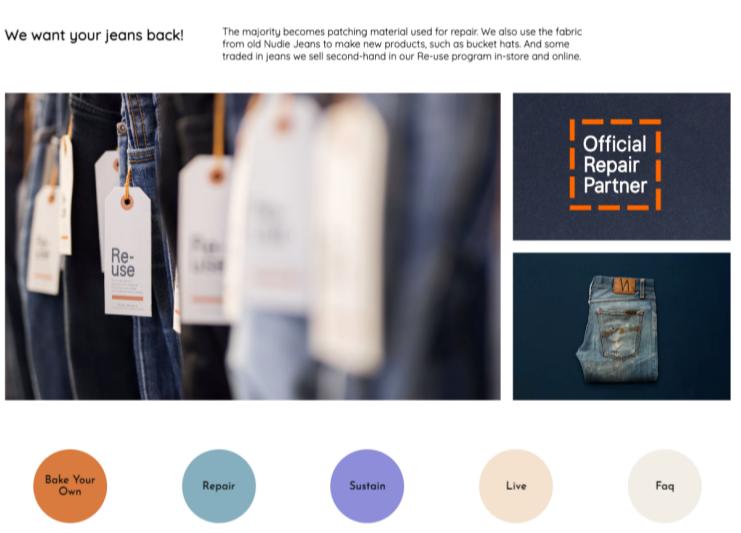
```
function createList(url) {
  fetch(url)
    .then(data => {
      return data.json();
    })
    .then(json => {
      let items = json.items;
      console.log(items);
      let result = "";
      items.map(item => {

        let title = item.snippet.title;
        if (title.length > 20) title = title.substr(0, 20) + "...";

        let desc = item.snippet.description;
        if (desc.length > 50) desc = desc.substr(0, 50) + "...";

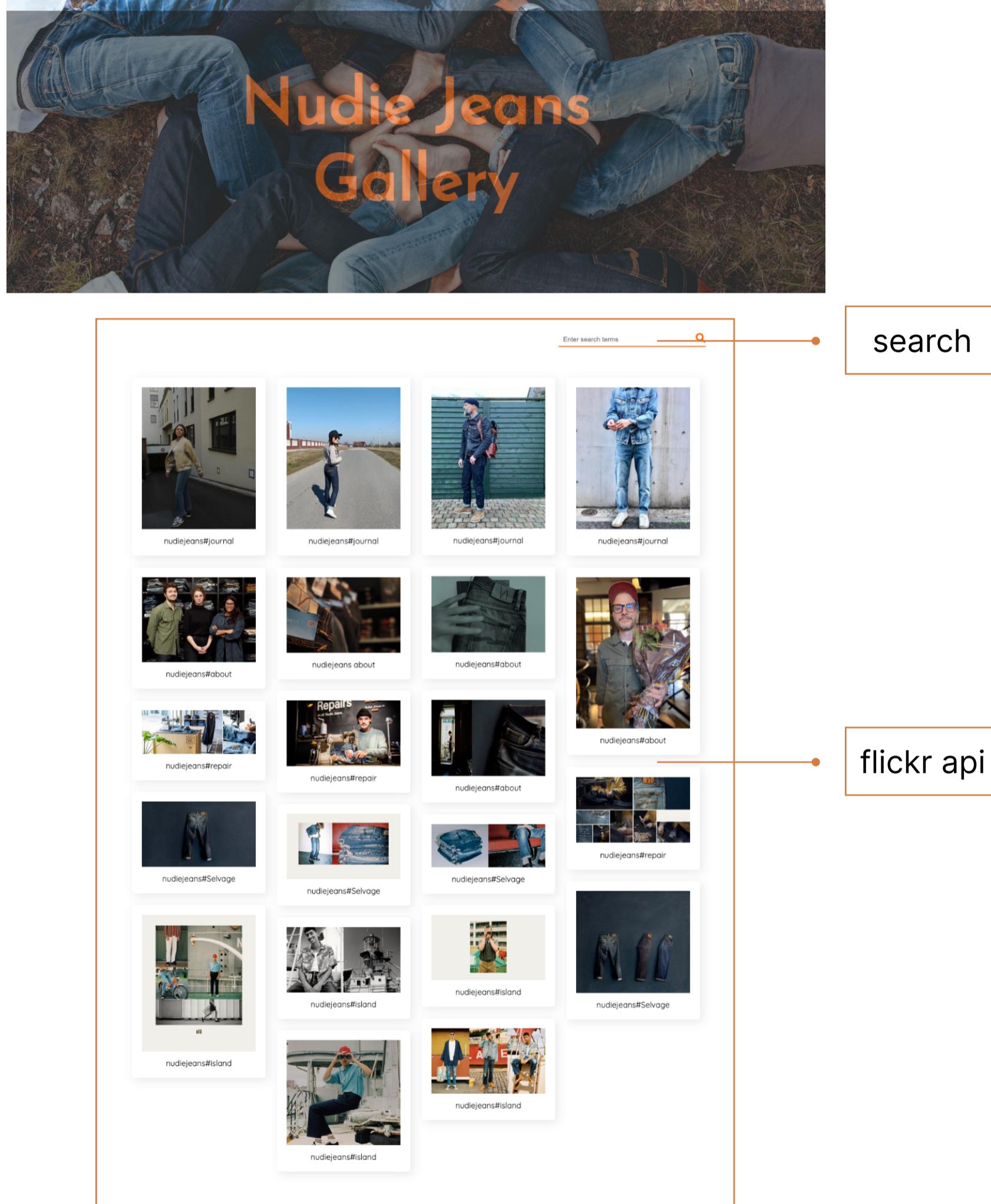
        let date = item.snippet.publishedAt;
        date = date.split("T")[0];
        result += `
          <article>
            <a href="#" data-vid="${item.snippet.resourceId.videoId}" class="pic">
              
              <i></i>
            </a>
            <div class="con">
              <h2>${title}</h2>
              <p>${desc}</p>
              <span>${date}</span>
            </div>
          </article>
        `;
      });
      vidCon.innerHTML = result;
    });
}
```

youtube api



html vidCon 안에 vidCon 설정해주고  
fetch, .then json으로 백틱``을 사용하여  
youtube list를 넣어줌

# subpage: gallery



HTML

```
<ul class="list"></ul>
```

JS

키값 설정

```
const key = "ac2a0a14b2c15e96b9b417eabf3b2694";
const base = "https://www.flickr.com/services/rest/?";
const photoGet = "flickr.people.getPublicPhotos";
const method_interest = "flickr.interestingness.getList";
const method_search = "flickr.photos.search";
```

```
fetch(url)
  .then(data => {
    return data.json();
  })
  .then(json => {
    let items = json.photos.photo;
    console.log(items);
  })
```

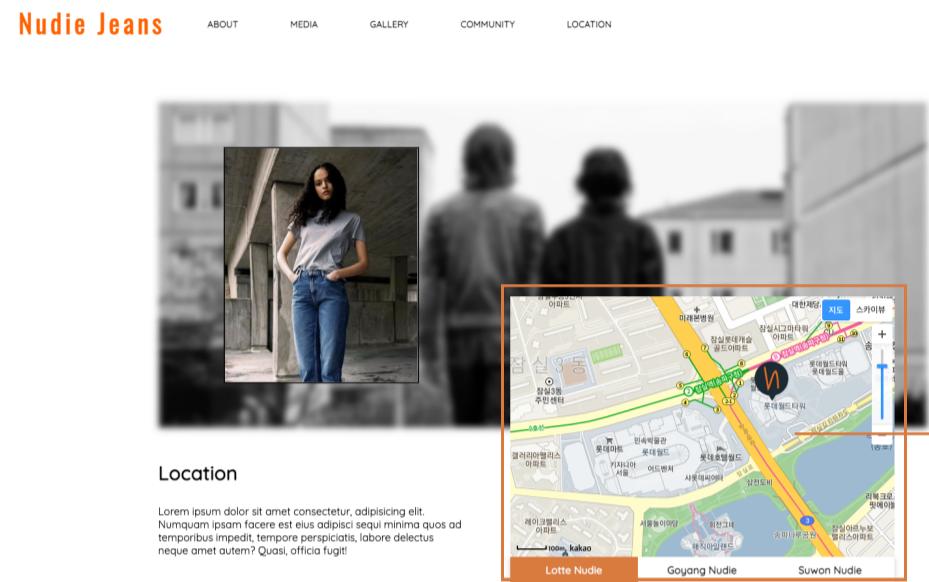
```
if (items.length > 0) {
  createList(items);
  imgLoaded();
} else {
  const errMsg = document.createElement("p");
  errMsg.append("No search results found");
  input.parentElement.append(errMsg);
```

```
frame.classList.remove("on");
loading.classList.add("off");
frame.innerHTML =
  <li class="no_search">
    <p>
      No search results foundz
    </p>
  </li>;
frame.style.height = "auto";
```

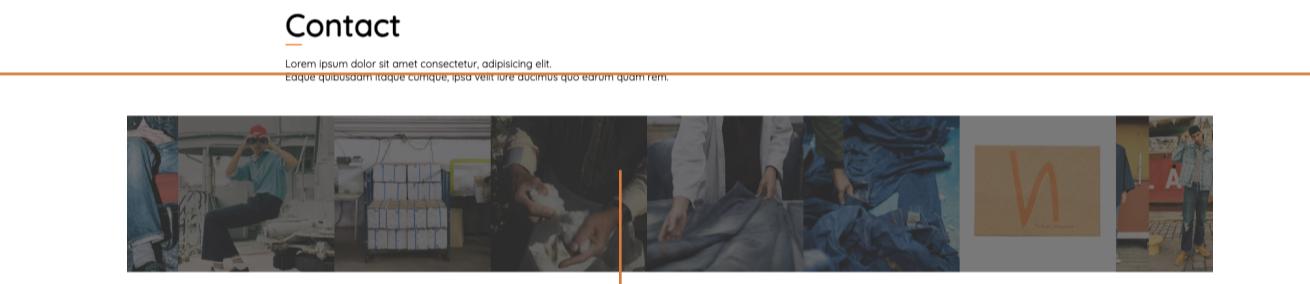
1. html list 안에 vidCon 설정해주고  
fetch, .then json으로 백틱 ``을 사용하여  
flickr list를 넣어줌

2. 버튼 클릭과 enter키 입력으로 설정된 tag 리스트 호출

# subpage: location



kakao api



rolling banner

```
fetch(url)
  .then(data => {
    return data.json();
  })
  .then(json => {
    let items = json.imgSrc;
    console.log(items);
    let tags = "";
    items.forEach(item => {
      tags += `
        <li>
          <a href=${item.pic}>
            <img src=${item.thumb}>
          </a>
        </li>
      `;
    });
    list.innerHTML = tags;
  });
}
```

data.json의 데이터를  
배열에 있는 객체를  
fetch를 이용하여 data.json에 요청해서  
함수로 호출

HTML

```
<div class="right">
  <div id="map"></div>
  <ul class="map_list">
    <li class="on"><a href="#">Lotte Nudie</a></li>
    <li><a href="#">Goyang Nudie</a></li>
    <li><a href="#">Suwon Nudie</a></li>
  </ul>
</div>
```

JS

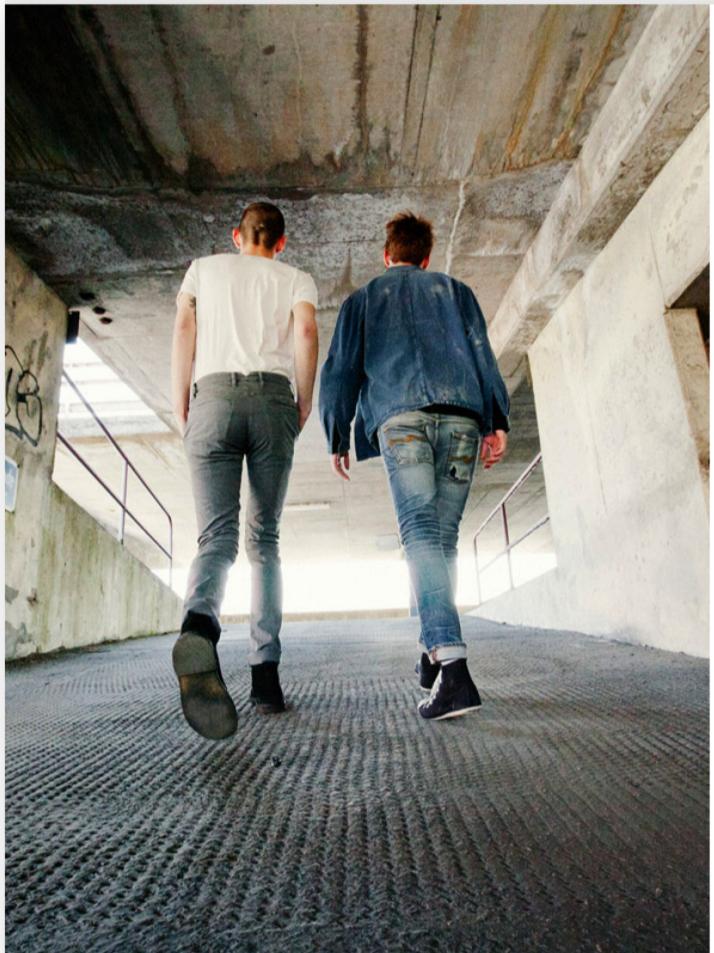
```
var mapOption = {
  center: new kakao.maps.LatLng(37.5121, 127.0997),
  level: 4
};
var map = new kakao.maps.Map(mapContainer, mapOption);

let drag = true;
let zoom = true;

let markerOptions = [
  {
    title:"본점",
    latlng:new kakao.maps.LatLng(37.5121, 127.0997),
    imgSrc:"img/location/marker1.png",
    imgSize:new kakao.maps.Size(52, 60),
    imgPos : {offset: new kakao.maps.Point(-80 ,100 )},
    button: branch_btns[0]
  },
]
```

html map 안에 vidCon 설정해주고  
fetch, .then json으로 백틱 ``을 사용하여  
youtube list를 넣어줌

# subpage: join



## Join Us

USER ID  
아이디를 입력해 주세요  
Please enter at least 5 characters for the entry item

PASSWORD  
비밀번호를 입력해 주세요  
Password must be at least 5 characters and must be entered in the same way, including numbers

HOBBY  
 GAME  MUSIC  READING  
Please check the required entry item.

GENDER  
 MALE  FEMALE  
Please check the required entry item.

E-MAIL  
이메일주소를 입력해 주세요  
Please enter an email address with at least 5 characters including '@'

## HTML

```
<form action="result.html" method="post" id="form">
  <fieldset>
    <legend class="h">회원가입양식폼</legend>
    <table>
      <caption class="h">회원정보 입력양식 테이블</caption>
      <tr scope="row">
        <th><label for="id">USER ID</label></th>
        <td>
          <input type="text" name="userid" id="id" placeholder="아이디를 입력해 주세요" >
        </td>
      </tr>
      <tr>
        <th scope="row"><label for="pwd1">PASSWORD</label></th>
        <td>
          <input type="password" name="pwd1" id="pwd1" placeholder="비밀번호를 입력해 주세요" >
        </td>
      </tr>
```

## JS

```
btnSubmit.addEventListener("click", e => {
  if (!isTxt("userid", 5)) e.preventDefault();
  if (!isEmail("email", 5)) e.preventDefault();
  if (!isCheck("hobby")) e.preventDefault();
  if (!isCheck("gender")) e.preventDefault();
  if (!isPwd("pwd1", 5)) e.preventDefault();
});

function isTxt(name, len) {
  let input = form.querySelector(`[name=${name}]`);
  let txt = input.value;
  if (txt.length >= len) {
    const errMsgs = input.closest("td").querySelectorAll("p");
    if (errMsgs.length > 0) input.closest("td").querySelector("p").remove();
    return true;
  } else {
    const errMsgs = input.closest("td").querySelectorAll("p");
    if (errMsgs.length > 0) input.closest("td").querySelector("p").remove();
    const errMsg = document.createElement("p");
    errMsg.append(`Please enter at least ${len} characters for the entry item`);
    input.closest("td").append(errMsg);
    return false;
  }
}
```

input submit에 클릭 이벤트를 걸고 전체 input값에 해당하는 텍스트가 있는지 없는지 확인함