



[모음] 1~5주차 요약

▼ PDF 다운받기

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/facc07e3-d70a-4a1f-8abe-61677ebb70cd/1_5cheatsheet.pdf



모든 토글을 열고 닫는 단축키

Windows : **Ctrl** + **alt** + **t**

Mac : **⌘** + **option** + **t**



"이게 어디있었더라", "또 어떤 걸 할 수 있지?"

하는 분들을 위해 준비했습니다. 이름하여 치트시트!

코드 요약

HTML, CSS, Javascript

▼ 1) HTML 기본 골격



아무것도 표기되지 않은 가장 기본 형태입니다.

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>스파르타코딩클럽의 페이지</title>
  <!-- head 안에 들어 갈 어떤 요소들을 여기에 넣습니다 -->
</head>
<body>
  <!-- body 안에 들어 갈 어떤 요소들을 여기에 넣습니다 -->
</body>
</html>
```

▼ 2) 자주 쓰이는 HTML 요소

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>스파르타코딩클럽 | HTML 기초</title>
</head>

<body>
  <!-- 구역을 나누는 태그들 -->
  <div>나는 영역을 나누죠. 속성을 주어야 눈에 보이는 모양이 달라져요.</div>
  <p>나는 문단이에요</p>
  <ul>

    <li>bullet point!1</li>
    <li>bullet point!2</li>
  </ul>

  <!-- 구역 내 콘텐츠 태그들 -->
```

```

<h1>h1은 제목을 나타내는 태그입니다. 페이지마다 하나씩 꼭 써주는 게 좋아요. 그래야 구글 검색이 잘 되거든요.</h1>
<h2>h2는 소제목입니다. 숫자가 커지면 제목 크기가 작아집니다. </h2>
<h3>h3~h6도 각자의 역할이 있죠. 비중은 작지만..</h3>
<hr>
span 태그입니다: 특정 <span style="color:red">글자</span>를 꾸밀 때 써요
<hr>
a 태그입니다: <a href="http://naver.com/"> 하이퍼링크 </a>
<hr>
img(이미지) 태그입니다: 
<hr>
input 태그입니다: <input type="text"/>
<hr>
button 태그입니다:
<button> 버튼입니다</button>
<hr>
textarea 태그입니다: <textarea>나는 무엇일까요?</textarea>
</body>

</html>

```

▼ 3) HTML class 선택자, 중첩



선택자를 살펴볼 수 있습니다.

1. red-font라는 class가 style에서 어떻게 정의되었고, 어떤 HTML 요소에 이름 붙어 있는지 살펴봐주세요.
2. red-font이라는 class와 small-font 라는 class가 띄어쓰기로 중첩되었습니다.

```

<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>스파르타코딩클럽 | 로그인페이지</title>
  <style>
    .red-font {
      color: red;
    }

    .small-font {
      font-size: 10px;
    }
  </style>
</head>

<body>
  <h1 class="red-font">로그인 페이지</h1>
  <div class="red-font small-font">
    <p>
      ID: <input type="text" />
    </p>
    <p>
      PW: <input type="password" />
    </p>
  </div>
  <button>로그인하기</button>
</body>

</html>

```

▼ 4) 자주쓰이는 CSS 소개

참고: <http://www.walterz.net/2017/08/18/자주-사용되는-css-속성/>



배경관련

background-color
background-image
background-size

사이즈

width
height

폰트

font-size
font-weight
font-family
color

간격

margin
padding

▼ 5) Bootstrap 기본 골격



HTML 페이지를 시작할 때, 아래 코드를 복사해서 쓰세요!

```
<!doctype html>
<html lang="ko">

<head>
  <!-- Required meta tags -->
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
    integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
    integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibx39j7fakFPskvXusvfa0b4Q"
    crossorigin="anonymous"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
    integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl"
    crossorigin="anonymous"></script>

  <title>스파르타코딩클럽 | 부트스트랩 연습하기</title>
</head>

<body>
  <h1>이걸로 시작해보죠!</h1>
</body>

</html>
```

▼ 6) Bootstrap 가져다쓰기



참고: <https://getbootstrap.com/docs/4.0/components/alerts/>
components의 적절한 코드를 <body></body>에 넣으면 끝!

```
<div class="jumbotron">
  <h1 class="display-4">나홀로 링크 메모장!</h1>
  <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
  <hr class="my-4">
  <p class="lead">
```

```

    <a class="btn btn-primary btn-lg" href="#" role="button">포스팅박스 열기</a>
  </p>
</div>

```

▼ 7) 1주차 나홀로 메모장 완성 코드 - 부트스트랩으로 화면 완성하기

```

<!doctype html>
<html lang="ko">

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
    integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

  <!-- JS -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
    integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
    crossorigin="anonymous"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
    integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmY1"
    crossorigin="anonymous"></script>

  <!-- 구글폰트 -->
  <link href="https://fonts.googleapis.com/css?family=Stylish&display=swap" rel="stylesheet">

  <title>스파라코딩클럽 | 나홀로 메모장</title>

  <!-- style -->
  <style type="text/css">
    * {
      font-family: 'Stylish', sans-serif;
    }

    .wrap {
      width: 900px;
      margin: auto;
    }

    .comment {
      color: blue;
      font-weight: bold;
    }

    .post-box {
      width: 500px;
      margin: 20px auto;
      padding: 50px;
      border: black solid;
      border-radius: 5px;
    }
  </style>
  <script>
    function sayHello() {
      alert('안녕!');
    }
  </script>

</head>

<body>
<div class="wrap">
  <div class="jumbotron">
    <h1 class="display-4">나홀로 링크 메모장!</h1>
    <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
    <hr class="my-4">
    <p class="lead">
      <a class="btn btn-primary btn-lg" href="#" role="button">포스팅박스 열기</a>
    </p>
  </div>
  <div class="form-post post-box">
    <div>
      <div class="form-group">
        <label for="exampleFormControlInput1">아티클 URL</label>
        <input class="form-control" placeholder="">
      </div>
      <div class="form-group">
        <label for="exampleFormControlTextarea1">간단 코멘트</label>

```

```

        <textarea class="form-control" rows="2"></textarea>
    </div>
    <button onclick="sayHello()" type="button" class="btn btn-primary">기사저장</button>
</div>
</div>
<div class="card-columns">
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
    <div class="card">
        
        <div class="card-body">
            <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
            <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼천리 화려강
            <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
    </div>
</div>
</div>
</body>
</html>

```

Javascript, jQuery, Ajax

▼ 1) Javascript 기초문법 (변수, 자료형, 함수, 반복문, 조건문, + 결합)

- 변수 선언

```
let firstName = 'Harry' // 문자열
let age = 24 // 숫자
let wizardStatus = true // 참거짓
```

- 자료형

- 리스트

```
let fruits = ['감', '배', '귤']
console.log(fruits[0]) //'감' 출력
fruits.push('수박') // 요소 추가
console.log(fruits[3]) //'수박' 출력
```

- 딕셔너리

```
let wizard = {'name': 'Harry Potter', 'age': 40}; // 로 선언 가능
console.log(wizard['name']); // 'Harry Potter'를 출력
console.log(wizard['age']); // 40을 출력

wizard['height'] = 173; // 딕셔너리에 키:밸류 넣기
console.log(wizard); // {name: "Harry Potter", age: 40, height: 173}을 출력
```

- 함수



JS에서 기본적으로 제공하는 여러 함수들이 존재합니다.

왠지 이걸 있을 것 같은데?(예 - 특정 문자를 바꾸고 싶다 등) 싶으면 직접 만들지 말고 구글에서 먼저 검색해보세
요!

```
// 두 숫자를 입력받으면 더한 결과를 돌려주는(return) 함수
function sum(num1, num2) {
  console.log('num1: ', num1, ', num2: ', num2);
  return num1 + num2;
}

let sumResult;
sumResult = sum(3, 5); // 8
console.log(sumResult);

sumResult = sum(4, -1);
console.log(sumResult); // 3
```

- 반복문

```
let names = ['Bob', 'Kay', 'Juice']
for (let i = 0; i < names.length; i++) {
  console.log(names[i]); // i가 0부터 2까지 돌
}
```

- 조건문 - if, else if, else

```
// 조건을 여러 개 수행하고 싶을 때
function checkGeneration(age){
  if (age > 120) {
    alert('와 19세기에 태어나셨군요!')
  } else if (age >= 80) {
    alert('80세 이상! 인생은 여든부터!')
  } else {
    alert('젊으시군요! 장래희망이 뭔가요?')
  }
}

let myAge = 55;
checkGeneration(myAge);
```

- AND 조건은 "&&"로, OR 조건은 "||"로 표현

```
// AND 조건 : 모든 조건들이 true 여야 true
// 나이가 20과 같거나 크고 30보다 작은 경우 true
function isTwenty(person) {
  if (person['age'] >= 20 && person['age'] < 30) {
    return true;
  }
}
```

```

    } else {
        return false;
    }
}

// OR 조건 : 조건 중 하나라도 true이면, true!
// 나이가 20보다 작거나, 30보다 크면 true
function isNotTwenty(person) {
    if (person['age'] < 20 || person['age'] >= 30) {
        // not(!) 을 사용해 !(20대인 조건) 으로 판별식을 만들 수도 있습니다.
        // 결과를 얻는 방법은 여러 가지! 가독성을 고려해 판별식을 만들어보세요!
        // if (!(person['age'] >= 20 && person['age'] < 30)){

            return true;
        } else {
            return false;
        }
    }
}

```

• 반복문

```

// 만약에, 0부터 n-1까지 더하는 함수를 만들고 싶다면?

function sum(n) {
    let result = 0
    for (let i = 0; i < n; i++) {
        result += i;
        // 반복문을 돌면서, result에다 i를 더해줍니다.
        // result = result + i; 와 같습니다.
    }
    return result
}

console.log(sum(10)) // 45를 출력

```

- 반복문은 리스트와 단짝! 리스트와 함께 자주 쓰입니다.

```

// 리스트의 값을 모두 출력하기
let wizards = ['덤블도어', '맥고나걸', '스네이프', '해리', '허마이오니', '론']

// 리스트이름.length 기억나시죠? 리스트의 길이!
// i가 1씩 증가하면서, 마법사들 wizards의 원소를 차례대로 불러올 수 있게 됩니다.
// 이렇게 하면 리스트의 모든 원소를 한번에 출력할 수 있겠죠?
for (let i = 0 ; i < wizards.length ; i++) {
    console.log(wizards[i])
}

```

- 반복문과 조건문의 결합 - 70세 미만인 마법사의 이름만 출력하기

```

let wizardsInfo = [
    { 'name': '덤블도어', 'age': 116 },
    { 'name': '맥고나걸', 'age': 85 },
    { 'name': '스네이프', 'age': 60 },
    { 'name': '해리', 'age': 40 },
    { 'name': '허마이오니', 'age': 40 },
    { 'name': '론', 'age': 40 },
]

for (let i = 0 ; i < wizardsInfo.length ; i++) {
    if (wizardsInfo[i]['age'] < 70) {
        console.log(wizardsInfo[i]['name']);
    }
}

```

▼ 2) HTML 버튼과 Javascript 연결해보기



버튼(button)을 클릭(onclick)하면, 이름을 적어둔 함수(handleClick)가 실행됩니다.

- 버튼을 짝수 번 클릭했을 때는 " 짹짹 ", 홀수 번 눌렀을 때는 " 훌훌훌 " alert 창을 띄우는 버튼을 만들기

```

<!DOCTYPE html>
<html lang="ko">

```

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>스파르타코딩클럽 | 자바스크립트 연습</title>
  <script>
    let num = 0;
    function handleClick() {
      num += 1;
      if (num % 2 == 0) {
        alert('짝짝짝 🎉');
      } else {
        alert('홀홀홀 🙄');
      }
    }
  </script>
</head>

<body>
  <button onclick="handleClick()">이 버튼을 눌러보세요</button>
</body>

</html>

```

▼ 3) HTML 로딩 후 Javascript 실행하기



HTML 요소들이 다 로딩되면, `$(document).ready` 안의 함수가 실행됩니다.
페이지를 새로고침하면,

```

<script>

$(document).ready(function(){
  showAlert();
});

function showAlert() {
  alert('이 함수는 HTML 로딩 후 불러집니다.')
}

</script>

```

▼ 4) jQuery 임포트하기

- bootstrap 템플릿에는 이미 import하는 코드가 적혀있어요! 추가로 안 적어주셔도 됩니다. 아래 같은 코드가 없다면 `<head>` 와 `</head>` 사이에 넣기

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

- 참고! jQuery import하기: ([설명 링크](#)) ([공식 CDN 링크](#))

▼ 5) 자주 사용하는 jQuery 모음

▼ 1. 사용할 HTML 코드 [memo.html](#) [[코드 JQuery 01](#)]

```

<!doctype html>
<html lang="ko">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
      integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
      crossorigin="anonymous">

    <!-- JS -->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
      integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
      crossorigin="anonymous"></script>

    <!-- 구글폰트 -->
    <link href="https://fonts.googleapis.com/css?family=Stylish&display=swap" rel="stylesheet">

```



```

<title>스파르타코딩클럽 | 나홀로 메모장</title>

<!-- style -->
<style type="text/css">
    * {
        font-family: 'Stylish', sans-serif;
    }

    .wrap {
        width: 900px;
        margin: auto;
    }

    .comment {
        color: blue;
        font-weight: bold;
    }

    #post-box {
        width: 500px;
        margin: 20px auto;
        padding: 50px;
        border: black solid;
        border-radius: 5px;
    }
</style>
<script>
    function openClose() {
        // 여기에 jQuery를 이용해 코드를 짤 예정
    }
</script>

</head>

<body>
    <div class="wrap">
        <div class="jumbotron">
            <h1 class="display-4">나홀로 링크 메모장!</h1>
            <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
            <hr class="my-4">
            <p class="lead">
                <button onclick="openClose()" id="btn-post-box" type="button" class="btn btn-primary">포스팅 박스 열기
            </button>
            </p>
        </div>
        <div id="post-box" class="form-post">
            <div>
                <div class="form-group">
                    <label for="post-url">아티클 URL</label>
                    <input id="post-url" class="form-control" placeholder="">
                </div>
                <div class="form-group">
                    <label for="post-comment">간단 코멘트</label>
                    <textarea id="post-comment" class="form-control" rows="2"></textarea>
                </div>
                <button type="button" class="btn btn-primary">기사저장</button>
            </div>
        </div>
        <div id="cards-box" class="card-columns">
            <div class="card">
                
                <div class="card-body">
                    <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                    <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
                    <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
                </div>
            </div>
            <div class="card">
                
                <div class="card-body">
                    <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                    <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
                    <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
                </div>
            </div>
            <div class="card">
                
                <div class="card-body">
                    <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>

```

```

        <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
        <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
    </div>
</div>
<div class="card">
    
    <div class="card-body">
        <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
        <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
        <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
    </div>
</div>
<div class="card">
    
    <div class="card-body">
        <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
        <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
        <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
    </div>
</div>
<div class="card">
    
    <div class="card-body">
        <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
        <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
        <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
    </div>
</div>
</div>
</body>
</html>

```

▼ 2. input 박스의 값을 가져와보기

- `memo.html` 에서 형광펜 표시된 부분을 주목하세요! (`memo.html` 내에서 'post-url' 검색 - `ctrl + F` 또는 `cmd + F`)

```

<div class="form-post">
    <div>
        <div class="form-group">
            <label for="post-url">아티클 URL</label>
            <input id="post-url" class="form-control" placeholder="">
        </div>
        <div class="form-group">
            <label for="post-comment">간단 코멘트</label>
            <textarea id="post-comment" class="form-control" rows="2"></textarea>
        </div>
        <button type="button" class="btn btn-primary">기사저장</button>
    </div>
</div>

```

[ 코드 JQuery 02]  크롬 개발자도구창 `console` 탭 에서 아래 코드치기

- 실행 전, 먼저 화면의 아티클 URL 에 `www.spartacodingclub.shop` 을 입력해보세요.

```

// id 값이 post-url인 곳을 가리키고, val()로 값을 가져온다.
$('#post-url').val(); // www.spartacodingclub.shop 이 뜨네요.

```

▼ 3. div 보이기 / 숨기기

- `memo.html` 에서 형광펜 표시된 부분을 주목하세요! (`memo.html` 내에서 'post-box' 검색 - `ctrl + F` 또는 `cmd + F`)

```

<div id="post-box" class="form-post">
    <div>
        <div class="form-group">
            <label for="post-url">아티클 URL</label>
            <input id="post-url" class="form-control" placeholder="">
        </div>
        <div class="form-group">

```

```

<label for="post-comment">간단 코멘트</label>
<textarea id="post-comment" class="form-control" rows="2"></textarea>
</div>
<button type="button" class="btn btn-primary">기사저장</button>
</div>
</div>

```

[ 코드 JQuery 03] 크롬 개발자도구창 console 탭 에서 아래 코드치기

```

// id 값이 post-box인 곳을 가리고, hide()로 안보이게 한다.
$('#post-box').hide();

// show()로 보이게 한다.(css의 display 값을 block으로 바꾼다)
$('#post-box').show();

```

▼ 4. css의 값 가져와보기 (우리는 display 속성 값을 가져와볼게요!)

- `$('#post-box').css('display')` 했을 때, 출력값이 어떻게 보이나요?

[ 코드 JQuery 04] 크롬 개발자도구창 console 탭 에서 아래 코드치기

```


$('#post-box').hide();
$('#post-box').css('display'); // 출력값: none
// hide()를 하면, css의 display 값을 none으로 바꾼다.
// id=post-box 요소의 css display 값이 none 이라서 눈에 안보였군요!

$('#post-box').show();
$('#post-box').css('display'); // 출력값: block
// show()를 하면, css의 display 값을 원래 값이 있던 block으로 바꾼다.
// id=post-box 요소의 css display 값이 block 이라서 눈에 잘 보였군요!

```

▼ 5. 태그 내 텍스트 입력하기


- 크롬 개발자도구창 console 탭 에서 아래 코드치기

[ 코드 JQuery 05-1] input box의 경우

```

$('#post-url').val('여기에 텍스트를 입력하면!');

```

[ 코드 JQuery 05-2] 그 외 - 예) 버튼의 텍스트 바꾸기

- `memo.html` 에서 형광펜 표시된 부분을 주목하세요! (`memo.html` 내에서 'btn-posting-box' 검색 - ctrl + F 또는 cmd + F)

```

// 가리고 싶은 버튼에 id 값을 준다음
<button id="btn-post-box" type="button" class="btn btn-primary">포스팅 박스 열기</button>

```

```

$('#btn-post-box').text('포스팅 박스 닫기');

```

▼ 6. 특정 HTML 요소 내 동적으로 요소(element) 만들기

- 포스팅하기 버튼을 누르면(동적으로) → `<div> ~ </div>` 안에 카드를 추가(요소 *element* 만들기)기능입니다.
- `memo.html` 에서 형광펜 표시된 부분을 주목하세요! (`memo.html` 내에서 'cards-box' 검색 - ctrl + F 또는 cmd + F 한 후, id 값 입력)

```

// 요소 element 를 넣고 싶은 태그에 id 값을 준 다음
<div id="cards-box" class="card-columns">
  <div class="card">
    
      <a href="https://naver.com/" class="card-title">여기 기사 제목이 들어가죠</a>
      <p class="card-text">여기에 기사 내용이 들어가겠쥬</p>
      <p class="card-text comment">여기엔 코멘트가 들어갑니다</p>
    </div>
  </div>
</div>

```

- 크롬 개발자도구]창 console 탭 에서 아래 코드치기

[코드 JQuery 06-1] 카드만들기 전 연습! 간단히 버튼을 넣어보기

```
let tempHtml = '<button>나는 추가될 버튼이다!</button>';
$('#cards-box').append(tempHtml);

// 버튼 코드 대신 다른 코드를 넣을 수도 있겠죠?
// '<div class='card'>카드 들어갈 자리</div>' 처럼요!
```

[코드 JQuery 06-2] 버튼 말고, 카드를 넣어보기

```
let imgUrl = 'https://movie-phinf.pstatic.net/20181019_236/1539926790655oHv5z_JPEG/movie_image.jpg';
let linkUrl = 'https://naver.com/';
let title = '해리포터와 마법사의 돌 영화 정보';
let desc = '기사의 요약 내용이 들어갑니다. 해리 포터(다니엘 래드클리프 분)는 위압적인 버논 숙부(리차드 그리피스 분)와 냉담한 이모 페투니아 (...';
let comment = '여기에 코멘트가 들어갑니다.';

// 주의: 홑따옴표(')가 아닌 backtick(`)으로 감싸야 합니다.
// Windows: 숫자 1번 키 왼쪽의 버튼을 누르면 backtick(`)이 입력됩니다.
// Mac : 영문으로 전환 후, ` 키를 누르면 됩니다.
// backtick을 사용하면 문자 중간에 Javascript 변수를 삽입할 수 있습니다.
let tempHtml = `<div class="card">
    
    <div class="card-body">
        <a href="${linkUrl}" class="card-title">${title}</a>
        <p class="card-text">${desc}</p>
        <p class="card-text comment">${comment}</p>
    </div>
</div>`;
$('#cards-box').append(tempHtml);
```

▼ 6) Ajax로 API 호출하기



[2주차] 데이터를 읽어오는(Read) GET 요청

```
$.ajax({
  type: "GET",
  url: "http://spartacodingclub.shop/post?author_give=스파르타코딩클럽",
  data: {},
  success: function(response){
    console.log(response)
  }
})
```



[4주차] 데이터를 새로 저장하는(Create) POST 요청

```
let url = 'https://www.nytimes.com/interactive/2016/08/25/universal/ko/36-hours-seoul-korean.html';
let comment = '코멘트 테스트';
let author = '스파르타코딩클럽';

$.ajax({
  type: "POST",
  url: "/post",
  data: { url_give:url, comment_give:comment, author_give:author },
  success: function(response){
    console.log(response)
  }
})
```

▼ 7) 2주차 나홀로 메모장 완성 코드 - 포스팅 박스 열고 닫기 기능 추가

```
<!Doctype html>
<html lang="ko">
```

```

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
    integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
    crossorigin="anonymous">

  <!-- JS -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
    integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
    crossorigin="anonymous"></script>

  <!-- 구글폰트 -->
  <link href="https://fonts.googleapis.com/css?family=Stylish&display=swap" rel="stylesheet">

  <title>스파르타코딩클럽 | 나홀로 메모장</title>

  <!-- style -->
  <style type="text/css">
    * {
      font-family: 'Stylish', sans-serif;
    }

    .wrap {
      width: 900px;
      margin: auto;
    }

    .comment {
      color: blue;
      font-weight: bold;
    }

    #post-box {
      width: 500px;
      margin: 20px auto;
      padding: 50px;
      border: black solid;
      border-radius: 5px;
    }
  </style>
  <script>
    function openClose() {
      // id 값 post-box의 display 값이 block 이면(= 눈에 보이면)
      if ($('#post-box').css('display') == 'block') {
        // post-box를 가리고
        $('#post-box').hide();
        // 다시 버튼을 클릭하면, 박스 열기를 할 수 있게 텍스트 바꿔두기
        $('#btn-post-box').text('포스팅 박스 열기');
      } else {
        // 아니면(눈에 보이지 않으면) post-box를 펴라
        $('#post-box').show();
        // 다시 버튼을 클릭하면, 박스 닫기를 할 수 있게 텍스트 바꿔두기
        $('#btn-post-box').text('포스팅 박스 닫기');
      }
    }
  </script>
</head>

<body>
  <div class="wrap">
    <div class="jumbotron">
      <h1 class="display-4">나홀로 링크 메모장!</h1>
      <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
      <hr class="my-4">
      <p class="lead">
        <button onclick="openClose()" id="btn-post-box" type="button" class="btn btn-primary">포스팅 박스 열기
      </button>
      </p>
    </div>
    <div id="post-box" class="form-post" style="display:none">
      <div>
        <div class="form-group">
          <label for="post-url">아티클 URL</label>
          <input id="post-url" class="form-control" placeholder="">
        </div>
        <div class="form-group">
          <label for="post-comment">간단 코멘트</label>
          <textarea id="post-comment" class="form-control" rows="2"></textarea>
        </div>
        <button type="button" class="btn btn-primary">기사저장</button>
      </div>
    </div>
  </div>

```

```

        </div>
    </div>
    <div id="cards-box" class="card-columns">
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
        <div class="card">
            
            <div class="card-body">
                <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
                <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
                <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
            </div>
        </div>
    </div>
</body>
</html>

```

▼ 8) 3주차 나홀로 메모장 완성 코드 - 포스팅 가져오기(Read) API 사용하기

```

<!doctype html>
<html lang="ko">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
        integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
        crossorigin="anonymous">

```

```

<!-- JS -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
        integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
        crossorigin="anonymous"></script>

<!-- 구글폰트 -->
<link href="https://fonts.googleapis.com/css?family=Stylish&display=swap" rel="stylesheet">

<title>스파르타코딩클럽 | 나홀로 메모장</title>

<!-- style -->
<style type="text/css">
    * {
        font-family: "Stylish", sans-serif;
    }

    .wrap {
        width: 900px;
        margin: auto;
    }

    .comment {
        color: blue;
        font-weight: bold;
    }

    #post-box {
        width: 500px;
        margin: 20px auto;
        padding: 50px;
        border: black solid;
        border-radius: 5px;
    }
</style>
<script>
    $(document).ready(function () {
        $("#cards-box").empty();
        showPost();
    });

    function showPost() {
        $.ajax({
            type: "GET",
            url: "http://spartacodingclub.shop/post",
            data: {},
            success: function (response) {
                let articles = response["articles"];
                for (let i = 0; i < articles.length; i++) {
                    let article = articles[i];
                    makeCard(article["image"], article["url"], article["title"], article["desc"], article["comment"])
                }
            }
        });
    }

    function makeCard(image, url, title, desc, comment) {
        let tempHtml = `<div class="card">
            
            <div class="card-body">
                <a href="${url}" target="_blank" class="card-title">${title}</a>
                <p class="card-text">${desc}</p>
                <p class="card-text comment">${comment}</p>
            </div>
        </div>`;
        $("#cards-box").append(tempHtml);
    }

    function openClose() {
        // id 값 post-box의 display 값이 block 이면(= 눈에 보이면)
        if ($("#post-box").css("display") == "block") {
            // post-box를 가리고
            $("#post-box").hide();
            // 다시 버튼을 클릭하면, 박스 열기를 할 수 있게 텍스트 바꿔두기
            $("#btn-post-box").text("포스팅 박스 열기");
        } else {
            // 아니면(눈에 보이지 않으면) post-box를 펴라
            $("#post-box").show();
            // 다시 버튼을 클릭하면, 박스 닫기를 할 수 있게 텍스트 바꿔두기
            $("#btn-post-box").text("포스팅 박스 닫기");
        }
    }
</script>
</head>

```

```

<body>
  <div class="wrap">
    <div class="jumbotron">
      <h1 class="display-4">나홀로 링크 메모장!</h1>
      <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
      <hr class="my-4">
      <p class="lead">
        <button onclick="openClose()" id="btn-post-box" type="button" class="btn btn-primary">포스팅 박스 열기
      </button>
      </p>
    </div>
    <div id="post-box" class="form-post" style="display:none">
      <div>
        <div class="form-group">
          <label for="post-url">아티클 URL</label>
          <input id="post-url" class="form-control" placeholder="">
        </div>
        <div class="form-group">
          <label for="post-comment">간단 코멘트</label>
          <textarea class="form-control" rows="2"></textarea>
        </div>
        <button type="button" class="btn btn-primary">기사저장</button>
      </div>
    </div>
    <div id="cards-box" class="card-columns">
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁화 삼?
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
    </div>
  </div>
</div>

```



```
</body>

</html>
```

Python, MongoDB, 스크래핑(크롤링)

▼ 1) 파이썬 기초 문법



파이썬에서는 들여쓰기(indent)로 블록 단위를 나눕니다(JS에서는 `{ }` 를 썼었죠?).
들여쓰기를 잘못하면 들여쓰기 에러(indentation error) 가 발생하죠! 들여쓰기가 매우 중요합니다.

▼ 변수 & 기본연산

```
a = 3 # 3을 a에 넣는다
print(a)

b = a # a를 b에 넣는다
print(b)

a = a + 1 # a+1을 다시 a에 넣는다
print(a)

num1 = a * b # a*b의 값을 num1이라는 변수에 넣는다
print(num1)

num2 = 99 # 99의 값을 num2이라는 변수에 넣는다
print(num2)
```

▼ 자료형

[📄 코드 자료형 01] 숫자, 문자형

```
name = 'Harry' # 변수에는 문자열이 들어갈 수도 있고,
print(name)

num = 12 # 숫자가 들어갈 수도 있고,
print(num)

number_status = True # True 또는 False -> "Boolean"형이 들어갈 수도 있습니다.
print(number_status)
```

[📄 코드 자료형 02] 리스트 형 (Javascript의 배열형과 동일)

```
a_list = [] # 비어있는 리스트 만들기
a_list.append(1) # 리스트에 값을 넣는다
print(a_list)

a_list.append([2, 3]) # 리스트에 [2,3]이라는 리스트를 다시 넣는다
print(a_list)

# print 로 값 확인해보기
# a_list의 값은? [1, [2,3]]
# a_list[0]의 값은? 1
# a_list[1]의 값은? [2,3]
# a_list[1][0]의 값은? 2
```

[📄 코드 자료형 03] Dictionary 형 (Javascript의 dictionary형과 동일)

```
a_dict = {} # 비어있는 딕셔너리 만들기

wizard = {'name': 'Harry Potter', 'age': 40}
print(wizard)

wizard['height'] = 173
print(wizard)

# print 로 값 확인해보기
# wizard의 값은? {'name': 'Harry Potter', 'age': 40, 'height': 173}
# wizard['name']의 값은? 'Harry Potter'
# wizard['age']의 값은? 40
# wizard['height']의 값은? 173
```

【코드 자료형 04】 Dictionary 형과 List형의 조합

```
wizards = [{'name': 'Harry Potter', 'age': 40}, {'name': 'Ron Weasley', 'age': 40}]
print(wizards)

# print 로 값 확인해보기
# wizards[0]['name']의 값은? 'Harry Potter'
# wizards[1]['name']의 값은? 'Ron Weasley'

new_wizard = {'name': 'Albus Potter', 'age': 14}
wizards.append(new_wizard)
print(wizards)

# print 로 값 확인해보기
# wizards의 값은? [{'name': 'Harry Potter', 'age': 40}, {'name': 'Ron Weasley', 'age': 40}, {'name': 'Albus Potter', 'age': 14}]
# wizards[2]['name']의 값은? 'Albus Potter'
```

▼ 함수



리마인드!

왠지 이걸 있을 것 같은데?(예 - 특정 문자를 바꾸고 싶다 등) 싶으면 직접 만들지 말고 **구글에서 먼저 검색해보세요!**

• 함수 기본 생김새

```
# 만들기
def 함수이름(필요한 변수들) :
    내릴 명령어들을 순차적으로 작성

# 사용하기
함수이름(필요한 변수들)
```

• 예시

```
def sum_all(a, b, c):
    return a + b + c

def mul(a, b):
    return a * b

result = sum_all(1, 2, 3) + mul(10, 10)

# print 로 값 확인해보기
# result라는 변수의 값은?
```

▼ 조건문 - ! 패턴 ! 에 집중하세요

```
# 조건을 여러 개 사용하고 싶을 때
def check_generation(age):
    if age > 120:
        print('와 19세기에 태어나셨군요!')
    elif age >= 80:
        print('80세 이상! 인생은 여든부터!')
    else:
        print('젊으시군요! 장래희망이 뭔가요?')

my_age = 55
check_generation(my_age)
```

▼ 반복문 - ! 패턴 ! 에 집중하세요



Python에서의 반복문은, 리스트나 문자열의 요소들을 하나씩 꺼내쓰는 형태입니다. 즉, 임의의 열(sequence, 리스트나 문자열처럼)의 항목들을 그 순서대로 꺼내어 반복합니다.

우리 수업에서는 리스트와 함께 씁니다! 아래 **! 패턴 !** 을 잘 기억해주세요!

```
professor_wizards = [
    {'name': '덤블도어', 'age': 116},
    {'name': '맥고나걸', 'age': 85},
    {'name': '스네이프', 'age': 60},
]

# 이번엔, 반복문과 조건문을 응용한 함수를 만들어봅시다.
# 마법사의 이름을 받으면, age를 리턴해주는 함수
def get_age(name, wizards):
    # wizards! 잊 줄 함수 선언에서 사용한 변수죠? 함수 사용하는 쪽에서 쓰는 변수명 아닙니다!
    for wizard in wizards:
        if wizard['name'] == name:
            return wizard['age']
    return '해당하는 이름이 없습니다'

print(get_age('덤블도어', professor_wizards))
print(get_age('맥고나걸', professor_wizards))
```

▼ 2) 웹스크래핑 기본 세팅

- 가상환경에 패키지 설치하기
 - beautifulsoup4** (줄여서 bs4): HTML 파싱을 편리하게 하는 패키지
 - requests** : URL 요청(Request)할 때 사용하는 패키지
- 스크래핑 코드 기본 세팅

```
import requests
from bs4 import BeautifulSoup

# 타겟 URL을 읽어서 HTML을 받아오고,
headers = {'User-Agent' : 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.
data = requests.get('https://movie.naver.com/movie/sdb/rank/rmovie.nhn?sel=pnt&date=20200716', headers=headers)

# HTML을 BeautifulSoup이라는 라이브러리를 활용해 검색하기 용이한 상태로 만들
# soup이라는 변수에 "파싱 용이해진 html"이 담긴 상태가 됨
# 이제 코딩을 통해 필요한 부분을 추출하면 된다.
soup = BeautifulSoup(data.text, 'html.parser')

#####
# (입맛에 맞게 코딩)
#####
```

- 웹 스크래핑은 웹 페이지 구조에 따라 조금씩 방법이 다르기 때문에 처음에는 좀 까다롭게 느껴질 수도 있어요. 하지만 여러 번 스크래핑 하다보면 여러분만의 팁이 생길 거예요!
- 필요할 때마다 bs4 함수를 검색해 활용해보세요.



스크래핑 팁 세 가지!

1. 문제 분석(화면 분석)

원하는 데이터가 화면 어디에 있는지 확인. selector 기억나시나요? 스크래핑도 원하는 부분을 가리켜서 정보를 가져오죠? 페이지 구조를 파악하는게 필요합니다.

2. 부모 요소부터 접근

내가 원하는 데이터의 부모(조상) 요소 부터 가져와보세요. 부모 요소부터 페이지 구조를 살펴보다보면 규칙을 찾을 수 있을 거예요.

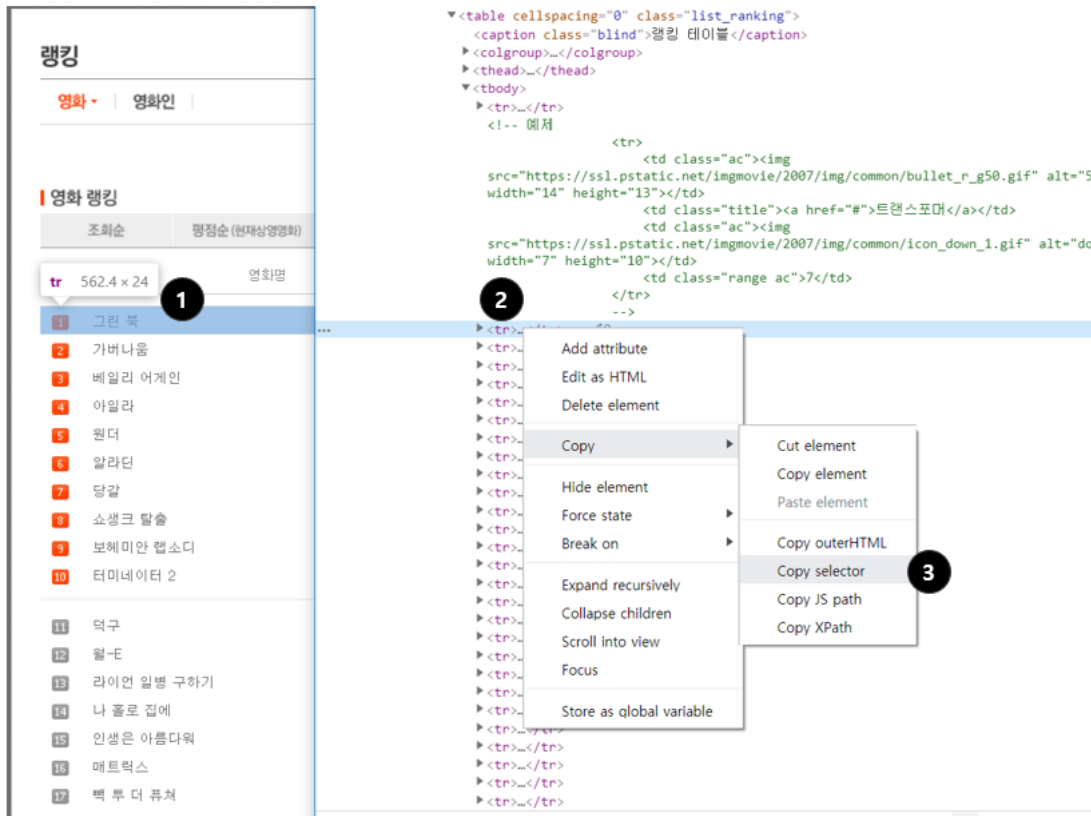
3. 값 출력하기

값을 제대로 가져오고 있는지 중간중간 출력(print)

▼ 3) 웹 페이지에서 가져오고 싶은 요소 분석하기 + beautifulsoup4의 select 사용

- 항상 정확하지는 않으나, 크롬 개발자도구를 참고할 수도 있습니다.

1. 원하는 부분에서 마우스 오른쪽 클릭 → 검사
2. 원하는 태그에서 마우스 오른쪽 클릭
3. Copy → Copy selector로 선택자를 복사할 수 있음



- BeautifulSoup 내 select에 미리 정의된 다른 방법을 알아봅시다
 - select는 여러개가 리스트에 담겨서, select_one은 한 개만 출력

```
# 여러개를 가져오고 싶은 경우
soup.select('태그명')
soup.select('.클래스명')
soup.select('#아이디명')

soup.select('상위태그명 > 하위태그명 > 하위태그명')
soup.select('상위태그명.클래스명 > 하위태그명.클래스명')

soup.select('태그명[속성="값"]')

# 한 개만 가져오고 싶은 경우
soup.select_one('위와 동일')
```

▼ 4) 네이버영화 스크래핑 코드

```
import requests
from bs4 import BeautifulSoup

# URL을 읽어서 HTML을 받아오고,
headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36'
}
data = requests.get('https://movie.naver.com/movie/sdb/rank/rmovie.nhn?sel=pnt&date=20200716', headers=headers)

# HTML을 BeautifulSoup이라는 라이브러리를 활용해 검색하기 용이한 상태로 만들
soup = BeautifulSoup(data.text, 'html.parser')

# select를 이용해서, tr들을 불러오기
movies = soup.select('#old_content > table > tbody > tr')
```

```
# movies (tr들) 의 반복문을 돌리기
for movie in movies:
    # movie 안에 a 가 있으면,
    a_tag = movie.select_one('td.title > div > a')
    if a_tag is not None:
        rank = movie.select_one('td:nth-child(1) > img')['alt'] # img 태그의 alt 속성값을 가져오기
        title = a_tag.text # a 태그 사이의 텍스트를 가져오기
        star = movie.select_one('td.point').text # td 태그 사이의 텍스트를 가져오기
        print(rank, title, star)
```

▼ 5) mongoDB 조작하기

- mongoDB 사용 전 DB 가 켜져있는지 확인

- 크롬 창에 localhost:27017 이라고 쳤을 때, 아래와 같은 메시지가 나오면 mongoDB가 동작하고 있는 것입니다.

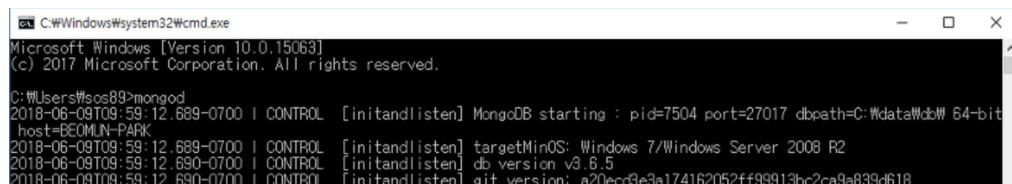
It looks like you are trying to access MongoDB over HTTP on the native driver port.

▼ Windows 에서 위 메시지가 안 뜰 때 빠른 해결방법

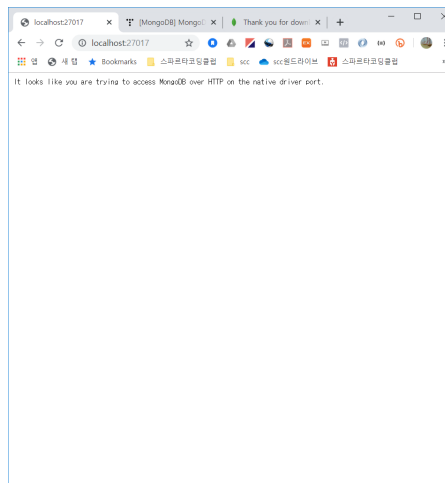
▼ mongoDB 실행시키기

1. 윈도우 키 + R 을 누른 후 cmd 를 입력하고 엔터를 누릅니다. 까만 터미널 창이 뜨면, 그리고 다음 명령어를 복사/붙여넣기하여 입력합니다.

```
mongod
```



2. 브라우저에서 localhost:27017 주소로 이동한 뒤 다음과 같은 화면이 뜨는 것을 확인하면 실행 완료!



3. 터미널 창을 종료하지 말아주세요. 터미널 창을 끄면 mongoDB 실행도 종료됩니다.

▼ 언제나 자동으로 실행되게 하고 싶다면 서비스 등록

1. 윈도우 키 + r 키 누른 후, cmd 입력하여 아래 코드를 복사/붙여넣기하여 입력하시면 서비스 등록 됩니다!

```
mongod --install --serviceName MongoDB --serviceDisplayName MongoDB --dbpath C:\data\db --logpath C:\data\log
```

- 만약 위 명령어 실행에도 작동이 안된다면, 다음 스텝을 따라보세요.

1. 관리자 권한으로 cmd를 실행합니다.
2. 아래 명령어를 입력합니다

```

mongod --remove

mongod --dbpath=C:\data\db --logpath=C:\data\db\log\log.txt --install

```

- 가상환경에 `pymongo` 패키지 설치하기
- DB연결하기

```

from pymongo import MongoClient # pymongo를 임포트 하기(패키지 인스톨 먼저 해야겠죠?)

client = MongoClient('localhost', 27017) # mongoDB는 27017 포트로 돌아갑니다.
db = client.dbsparta # 'dbsparta'라는 이름의 db를 사용합니다. 'dbsparta' db가 없다면 새로 만듭니다.

```



5분꿀팁에서 배운 CRUD 기억나시죠?

기능 기본 단위! Create(생성), Read(조회), Update(업데이트), Delete(삭제)

딱 네 가지 기능 어떻게 쓰는지 정리하면 아래와 같습니다.

```

# Create(생성)
user1 = {'name': '론', 'age': 40}
user2 = {'name': '해리', 'age': 40}
db.users.insert_one(user1)
db.users.insert_one(user2)

# Read(조회) - 한 개 값만
user = db.users.find_one({'name': '론'})

# Read(조회) - 여러 값 ( _id 값은 제외하고 출력)
same_ages = list(db.users.find({'age': 40}, {'_id': False}))

# Update(업데이트) - 여러 값
db.people.update_many({'age': 40}, {'$set': {'age': 70}})

# Update(업데이트) - 하나 값
db.users.update_one({'name': '론'}, {'$set': {'age': 116}})

# Delete(삭제)
db.users.delete_one({'name': '론'})

```

▼ 6) 네이버영화 스크래핑 결과를 mongoDB 에 저장하기

```

import requests
from bs4 import BeautifulSoup
from pymongo import MongoClient # pymongo를 임포트 하기(패키지 인스톨 먼저 해야겠죠?)

client = MongoClient('localhost', 27017) # mongoDB는 27017 포트로 돌아갑니다.
db = client.dbsparta # 'dbsparta'라는 이름의 db를 사용합니다. 'dbsparta' db가 없다면 새로 만듭니다.

# 타겟 URL을 읽어서 HTML를 받아오고,
headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36'
}
data = requests.get('https://movie.naver.com/movie/sdb/rank/rmovie.nhn?sel=pnt&date=20200716', headers=headers)

# HTML을 BeautifulSoup이라는 라이브러리를 활용해 검색하기 용이한 상태로 만들
# soup이라는 변수에 "파싱 용이해진 html"이 담긴 상태가 됨
# 이제 코딩을 통해 필요한 부분을 추출하면 된다.
soup = BeautifulSoup(data.text, 'html.parser')

# select를 이용해서, tr들을 불러오기
movies = soup.select('#old_content > table > tbody > tr')

# movies (tr들)의 반복문을 돌리기
for movie in movies:
    # movie 안에 a 가 있으면,
    a_tag = movie.select_one('td.title > div > a')
    if a_tag is not None:
        # a의 text를 찍어본다.
        rank = movie.select_one('td:nth-child(1) > img')['alt'] # img 태그의 alt 속성값을 가져오기
        title = a_tag.text # a 태그 사이의 텍스트를 가져오기
        star = movie.select_one('td.point').text # td 태그 사이의 텍스트를 가져오기
        # print(rank, title, star)
        doc = {
            'rank': rank,

```

```
'title': title,
'star': star # DB에는 숫자처럼 생긴 문자열 형태로 저장됩니다.
}
db.movies.insert_one(doc)
```

▼ 7) meta 태그 스크래핑하기



[4주차] meta태그 중 property가 og:image인 태그의 content 값을 가져오기

```
import requests
from bs4 import BeautifulSoup

url = 'https://platum.kr/archives/120958'

headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safa
data = requests.get(url, headers=headers)

soup = BeautifulSoup(data.text, 'html.parser')

og_image = soup.select_one('meta[property="og:image"]')
og_title = soup.select_one('meta[property="og:title"]')
og_description = soup.select_one('meta[property="og:description"]')

# 값을 잘 가져오는지 출력해서 확인하기 - 사용할 땐 주석 해제, 사용 안하면 주석 처리
# print(og_image)
# print(og_title)
# print(og_description)

url_image = og_image['content']
url_title = og_title['content']
url_description = og_description['content']

print(url_image)
print(url_title)
print(url_description)
```

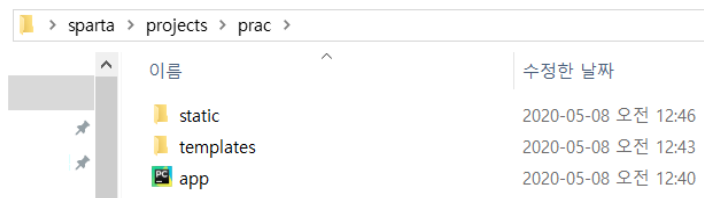
API 만들고 사용하기

▼ 1) Flask: 기본 구조 + URL 새롭게 부여하기(routing)



Flask 를 사용할 때는 기본 설정을 지켜주어야 합니다.
프로젝트 폴더 안에, 폴더를 만들어주세요!

static 폴더 : 이미지, css파일을 넣어둡니다.
templates 폴더 : html파일을 넣어둡니다.
app.py 파일



```
# app.py 파일입니다.
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello World!'

@app.route('/mypage')
```

```
def my_page():
    return 'This is My Page!'

if __name__ == '__main__':
    app.run('0.0.0.0', port=5000, debug=True)
```

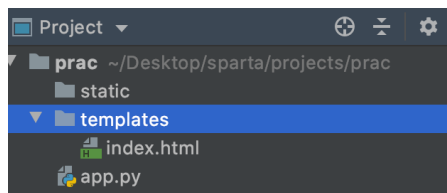
▼ 2) Flask 로 화면 띄우기 - HTML 파일 사용



templates 폴더에는

HTML 파일을 담아둡니다. 실행할 때에는 이 폴더에서 화면을 불러오죠.

1. 간단한 index.html 파일을 templates 안에 만들기



```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <title>첫 Flask 프로젝트</title>
  </head>
  <body>
    <h1>서버를 만들었다!</h1>
  </body>
</html>
```

2. HTML 파일 불러오기

- `app.py` 파일을 아래 분홍 형광펜 부분을 수정/추가하고 실행해보세요.



Flask 내장 함수 `render_template`를 이용해 HTML 파일 불러옵니다.
바로 이게 프레임워크의 위력!

```
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def home(): # 함수명 수정 - 이름만 보고 접속되는 페이지를 확인할 수 있게!
    return render_template('index.html')

@app.route('/mypage')
def my_page():
    return 'This is My Page!'

if __name__ == '__main__':
    app.run('0.0.0.0', port=5000, debug=True)
```

▼ 3) Flask 로 화면 띄우기 - HTML 화면에 이미지나 CSS 파일을 불러오기

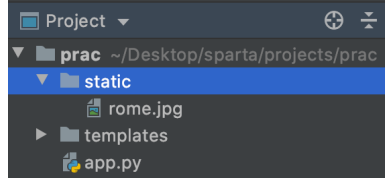


static 폴더는

이미지나 css파일과 같은 정적 파일을 담아두는 역할을 하지요!

1. 아래 이미지 다운로드 받고 static 폴더에 넣습니다.

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/bd59a681-63a4-4dab-86cb-eacfc40b0b82/rome.jpg>



2. Flask 에서 HTML 을 부를 때, 저장된 이미지 파일을 불러오기



이 HTML 파일은 Flask에서 읽어주기 때문에,
Flask에 미리 정의된 규칙대로 경로를 입력해줘야합니다.

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <title>Title</title>
  </head>
  <body>
    <h1>서버를 만들었다!</h1>
    
  </body>
</html>
```

- css 파일을 불러오고 싶을 때도 사용방법은 똑같습니다. static 폴더에 css 파일(파일명 mystyle)을 넣고, 아래 처럼 가져오기

```
<link rel="stylesheet" href="{{ url_for('static', filename='mystyle.css') }}">
```

▼ 4) Flask: GET, POST API 만들고 사용하기



[4주차] prac 폴더에 만든 기초 API 만들고 사용하기



[API 정보]

1. GET API

A. 요청 정보

- 요청 URL = `/test`, 요청 방식 = `GET`
- 요청 데이터 : 없음

B. 서버가 제공할 기능 : 클라이언트에게 정해진 메시지를 보냄

C. 응답 데이터 : (JSON 형식) 'result'= 'success', 'msg'= '이 요청은 GET!'

2. POST API

A. 요청 정보

- 요청 URL = `/test`, 요청 방식 = `POST`
- 요청 데이터 : 없음

B. 서버가 제공할 기능 : 클라이언트에게 정해진 메시지를 보냄

C. 응답 데이터 : (JSON 형식) 'result'= 'success', 'msg'= '이 요청은 POST!'

```

from flask import Flask, render_template, jsonify, request

app = Flask(__name__)

## HTML을 주는 부분
@app.route('/')
def home():
    return render_template('index.html')

## API 역할을 하는 부분
@app.route('/test', methods=['GET'])
def test_get():
    title_receive = request.args.get('title_give')
    print(title_receive)
    return jsonify({'result': 'success', 'msg': '이 요청은 GET!'})

@app.route('/test', methods=['POST'])
def test_post():
    title_receive = request.form['title_give']
    print(title_receive)
    return jsonify({'result': 'success', 'msg': '이 요청은 POST!'})

if __name__ == '__main__':
    app.run('0.0.0.0', port=5000, debug=True)

```

▼ Ajax 사용하기 위해 아래 HTML 을 띄운 화면에서 크롬 개발자도구 - console창 을 열어 API 를 사용하기

```

<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <title>첫 Flask 프로젝트</title>
  </head>
  <body>
    <h1>서버를 만들었다!</h1>
  </body>
</html>

```

▼ GET API 사용하기

- Ajax 사용하기 위해 index.html 파일 띄운 화면에서 크롬 개발자도구 - console창 을 열어 아래 코드를 실행해주세요.

```

$.ajax({
  type: "GET",
  url: "/test?title_give=봄날은간다",
  data: {},
  success: function(response){
    console.log(response)
  }
})

```

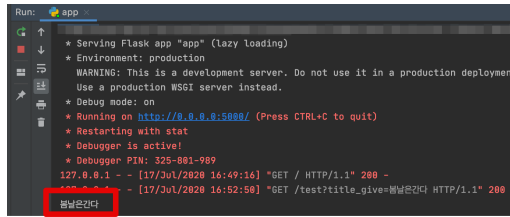
- 크롬 console 화면에 뜨는 메시지

```

> $.ajax({
  type: "GET",
  url: "/test?title_give=봄날은간다",
  data: {},
  success: function(response){
    console.log(response)
  }
})
< {readyState: 1, getResponseHeader: f, getAllRe
rs: f, setRequestHeader: f, overrideMimeType:
  > {msg: "이 요청은 GET!", result: "success"}

```

- Flask 프로젝트 화면에는 뜨는 메시지

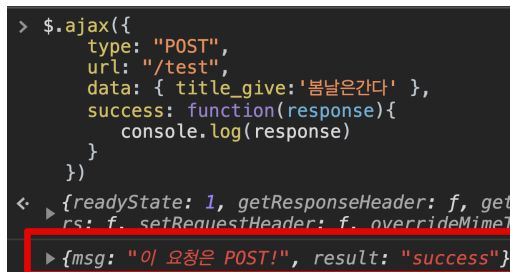


▼ POST API 사용하기

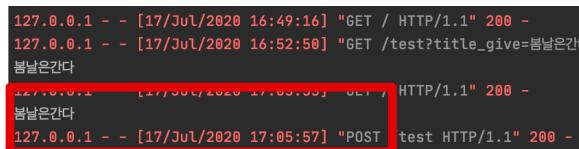
- Ajax 사용하기 위해 `index.html` 파일 띄운 화면에서 크롬 개발자도구 - console창을 열어 아래 코드를 실행해주세요.

```
$.ajax({
  type: "POST",
  url: "/test",
  data: { title_give:"불날은간다" },
  success: function(response){
    console.log(response)
  }
})
```

- 크롬 console 화면에 뜨는 메시지



- Flask 프로젝트 화면에는 뜨는 메시지



▼ 5) Flask: 나홀로메모장 전체 코드

- ▼ 클라이언트 코드 `index.html`

```
<!doctype html>
<html lang="ko">

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
    integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAw1GgFAW/dAiS6JXm"
    crossorigin="anonymous">

  <!-- JS -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
    integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
    crossorigin="anonymous"></script>

  <!-- 구글폰트 -->
  <link href="https://fonts.googleapis.com/css?family=Stylish&display=swap" rel="stylesheet">

  <title>스파르타코딩클럽 | 나홀로 메모장</title>
```

```

<!-- style -->
<style type="text/css">
    * {
        font-family: "Stylish", sans-serif;
    }

    .wrap {
        width: 900px;
        margin: auto;
    }

    .comment {
        color: blue;
        font-weight: bold;
    }

    #post-box {
        width: 500px;
        margin: 20px auto;
        padding: 50px;
        border: black solid;
        border-radius: 5px;
    }
</style>
<script>
    $(document).ready(function () {
        $("#cards-box").html("");
        showArticles();
    });

    function openClose() {
        // id 값 post-box의 display 값이 block 이면(= 눈에 보이면)
        if ($("#post-box").css("display") == "block") {
            // post-box를 가리고
            $("#post-box").hide();
            // 다시 버튼을 클릭하면, 박스 열기를 할 수 있게 텍스트 바뀌두기
            $("#btn-post-box").text("포스팅 박스 열기");
        } else {
            // 아니면(눈에 보이지 않으면) post-box를 펴라
            $("#post-box").show();
            // 다시 버튼을 클릭하면, 박스 닫기를 할 수 있게 텍스트 바뀌두기
            $("#btn-post-box").text("포스팅 박스 닫기");
        }
    }

    function postArticle() {
        let url = $("#post-url").val();
        let comment = $("#post-comment").val();

        // 2. memo에 POST 방식으로 메모 생성 요청하기
        $.ajax({
            type: "POST", // POST 방식으로 요청하겠다.
            url: "/memo", // memo라는 url에 요청하겠다.
            data: {url_give: url, comment_give: comment}, // 데이터를 주는 방법
            success: function (response) { // 성공하면
                if (response["result"] == "success") {
                    alert("포스팅 성공!");
                    // 3. 성공 시 페이지 새로고침하기
                    window.location.reload();
                } else {
                    alert("서버 오류!")
                }
            }
        })
    }

    function showArticles() {
        $.ajax({
            type: "GET",
            url: "/memo",
            data: {},
            success: function (response) {
                let articles = response["articles"];
                console.log(articles);
                for (let i = 0; i < articles.length; i++) {
                    makeCard(articles[i]["image"], articles[i]["url"], articles[i]["title"], articles[i]["desc"]
                )
                }
            }
        })
    }

    function makeCard(image, url, title, desc, comment) {
        let tempHtml = `<div class="card">
            
            <div class="card-body">
                <a href="${url}" target="_blank" class="card-title">${title}</a>
                <p class="card-text">${desc}</p>

```

```

<p class="card-text comment">${comment}</p>
</div>
</div>;
$("##cards-box").append(tempHtml);
}
</script>

</head>

<body>
  <div class="wrap">
    <div class="jumbotron">
      <h1 class="display-4">나홀로 링크 메모장!</h1>
      <p class="lead">중요한 링크를 저장해두고, 나중에 볼 수 있는 공간입니다</p>
      <hr class="my-4">
      <p class="lead">
        <button onclick="openClose()" id="btn-post-box" type="button" class="btn btn-primary">포스팅 박스 열기
      </button>
      </p>
    </div>
    <div id="post-box" class="form-post" style="display:none">
      <div>
        <div class="form-group">
          <label for="post-url">아티클 URL</label>
          <input id="post-url" class="form-control" placeholder="">
        </div>
        <div class="form-group">
          <label for="post-comment">간단 코멘트</label>
          <textarea id="post-comment" class="form-control" rows="2"></textarea>
        </div>
        <button type="button" class="btn btn-primary" onclick="postArticle()">기사저장</button>
      </div>
    </div>
    <div id="cards-box" class="card-columns">
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
      <div class="card">
        
        <div class="card-body">
          <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
          <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
          <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
      </div>
    </div>
  </div>

```

```

        src="https://www.eurail.com/content/dam/images/eurail/Italy%20CP%20Promo%20Block.adaptive.767
        alt="Card image cap">
        <div class="card-body">
        <a href="#" class="card-title">여기 기사 제목이 들어가죠</a>
        <p class="card-text">기사의 요약 내용이 들어갑니다. 동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라만세 무궁:
        <p class="card-text comment">여기에 코멘트가 들어갑니다.</p>
        </div>
        </div>
        </div>
        </div>
        </body>

</html>

```

▼ 서버 코드 `app.py`

```

from flask import Flask, render_template, jsonify, request
import requests
from bs4 import BeautifulSoup
from pymongo import MongoClient # pymongo를 임포트 하기(패키지 인스톨 먼저 해야겠죠?)

app = Flask(__name__)

client = MongoClient('localhost', 27017) # mongoDB는 27017 포트로 돌아갑니다.
db = client.dbsparta_ninework # 'dbsparta'라는 이름의 db를 만들거나 사용합니다.

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/memo', methods=['POST'])
def post_article():
    # 1. 클라이언트로부터 데이터를 받기
    url_receive = request.form['url_give'] # 클라이언트로부터 url을 받는 부분
    comment_receive = request.form['comment_give'] # 클라이언트로부터 comment를 받는 부분

    # 2. meta tag를 스크래핑하기
    headers = {
        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683
    data = requests.get(url_receive, headers=headers)
    soup = BeautifulSoup(data.text, 'html.parser')

    og_image = soup.select_one('meta[property="og:image"]')
    og_title = soup.select_one('meta[property="og:title"]')
    og_description = soup.select_one('meta[property="og:description"]')

    url_title = og_title['content']
    url_description = og_description['content']
    url_image = og_image['content']

    article = {'url': url_receive, 'title': url_title, 'desc': url_description, 'image': url_image,
               'comment': comment_receive}

    # 3. mongoDB에 데이터를 넣기
    db.articles.insert_one(article)

    return jsonify({'result': 'success'})

@app.route('/memo', methods=['GET'])
def read_articles():
    # 1. mongoDB에서 _id 값을 제외한 모든 데이터 조회해오기 (Read)
    result = list(db.articles.find({}, {'_id': 0}))
    # 2. articles라는 키 값으로 article 정보 보내주기
    return jsonify({'result': 'success', 'articles': result})

if __name__ == '__main__':
    app.run('0.0.0.0', port=5000, debug=True)

```