

Null4u 웹페이지 만들고 DNS 추가해서 배포하기

김기현 양지호

목차

- 만든 이유
- 웹
- 코드
- 아쉬운 점
- 웹 배포 과정
- dns설정
- 아쉬운 점
- 마무리

만든 이유

웹페이지 한번 만들어 보고 싶었고 자연스럽게 aws와도 주제 연결돼서 만들게 되었습니다

공통 기능들

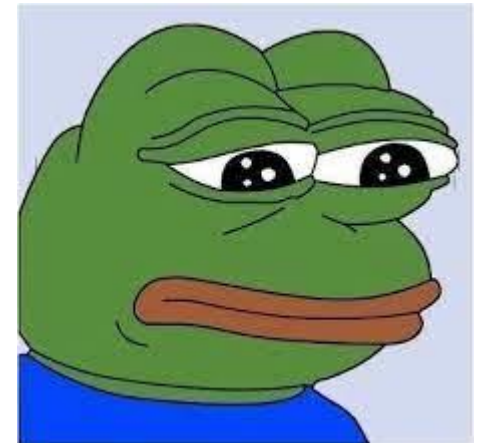
- 상단 메뉴
- 클릭시 랜덤이미지

상단 메뉴

페이지를 만들기 전에 여러 웹들을 둘러봤는데 거의 모든 웹이 상단에 메뉴바? 가 있었습니다



확실히 없으면 뭔가 허전한 느낌을 받았습니다



상단 메뉴 코드

```
<header>
  <h1><a href="test.html">NULL4U</a></h1>
  <nav>
    <ul>
      <li><a href="about.html">동아리소개</a></li>
      <li><a href="members.html">멤버</a></li>
      <li><a href="AWS.html">클라우드</a></li>
      <li><a href="https://smc-secu.net/">보안과</a></li>
    </ul>
  </nav>
</header>
```

```
header {
  position: fixed;
  width: 100%;
  top: 0;
  background-color: #fff;
  box-shadow: 0 2px 5px rgba(74, 190, 209);
  z-index: 1000;
  transition: background-color 0.3s ease, box-shadow 0.3s ease;
  padding: 10px;
}

header.scroll {
  background-color: #ddd;
  box-shadow: 0 4px 8px rgba(74, 190, 209);
}

header h1 {
  margin: 0;
  padding-right: 20px;
}

header a {
  text-decoration: none;
  color: black;
}

header nav {
  display: flex;
  justify-content: flex-end;
}

header ul {
  padding: 0;
  margin: 0;
  list-style: none;
  display: flex;
}

header ul li {
  margin-right: 20px;
}

header ul li a {
  padding: 5px;
  display: inline-block;
}

header ul li a:hover {
  background-color: #eee;
}
```

클릭시 랜덤이미지

그냥 페이지만 있으면 심심하기 때문에 마우스 클릭시 근처에 이미지나 나타나는 기능을 넣었습니다

전체 코드

```
<script>
  document.addEventListener('DOMContentLoaded', function () {
    var imageContainer = document.getElementById('image-container');
    var randomImage = document.getElementById('random-image');

    document.body.addEventListener('click', function (event) {
      var imagePaths = [
        'images\\umm.jpg',
        'images\\bqi.jpg',
        'images\\chipi.png'
      ];

      var randomIndex = Math.floor(Math.random() * imagePaths.length);
      var randomImagePath = imagePaths[randomIndex];

      var tempImage = new Image();

      tempImage.src = randomImagePath;

      tempImage.onload = function () {
        var mouseX = event.clientX;
        var mouseY = event.clientY;

        var imageWidth = tempImage.width;
        var imageHeight = tempImage.height;

        imageContainer.style.left = mouseX - imageWidth / 2 + 'px';
        imageContainer.style.top = mouseY - imageHeight / 2 + 'px';

        randomImage.src = randomImagePath;

        imageContainer.style.display = 'block';

        setTimeout(function () {
          imageContainer.style.display = 'none';
        }, 500);
      };
    });
  });
</script>
```


코드 설명

```
document.addEventListener('DOMContentLoaded', function () {  
  var imageContainer = document.getElementById('image-container'); // 이미지 컨테이너  
  var randomImage = document.getElementById('random-image'); // 랜덤 이미지 엘리먼트  
  
  document.body.addEventListener('click', function (event) {  
    // 랜덤 이미지 경로 배열  
    var imagePaths = ['images\\umm.jpg',  
                      'images\\bqi.jpg',  
                      'images\\chipi.png'];  
  });
```



코드 설명

```
// 랜덤 이미지 선택
var randomIndex = Math.floor(Math.random() * imagePaths.length);
var randomImagePath = imagePaths[randomIndex];

// 새로운 이미지 객체 생성 및 로드
var tempImage = new Image();
tempImage.src = randomImagePath;

// 이미지가 로드된 후 실행되는 함수
tempImage.onload = function () {
  // 클릭한 마우스 위치
  var mouseX = event.clientX;
  var mouseY = event.clientY;

  // 이미지 컨테이너 위치 설정
  imageContainer.style.left = mouseX - tempImage.width / 2 + 'px';
  imageContainer.style.top = mouseY - tempImage.height / 2 + 'px';
}
```

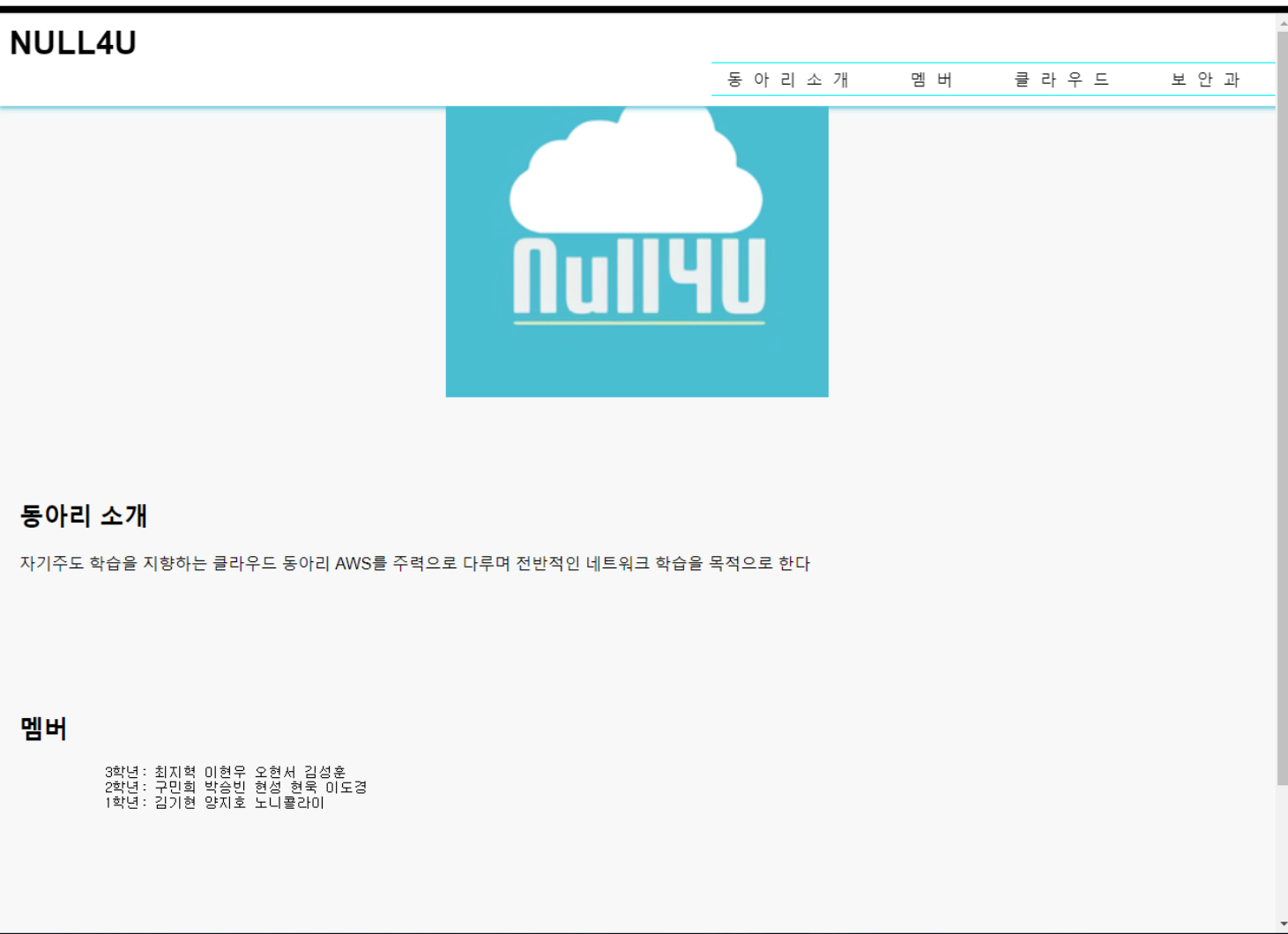
코드 설명

```
// 랜덤 이미지 설정 및 표시
randomImage.src = randomImagePath;
imageContainer.style.display = 'block';

// 일정 시간 후 이미지 숨김
setTimeout(function () {
    imageContainer.style.display = 'none';
}, 500);
```

페이지 소개

메인(index) 페이지



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

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>NULL4U</title>
  <link rel="stylesheet" href="styles.css">
</head>

<body>
  <header>
    <h1><a href="http://www.n4uweb.site">NULL4U</a></h1>
    <nav>
      <ul>
        <li><a href="http://www.n4uweb.site/about.html">동아리소개</a></li>
        <li><a href="http://www.n4uweb.site/members.html">멤버</a></li>
        <li><a href="http://www.n4uweb.site/aws.html">클라우드</a></li>
        <li><a href="https://smc-secu.net/">보안과</a></li>
      </ul>
    </nav>
  </header>

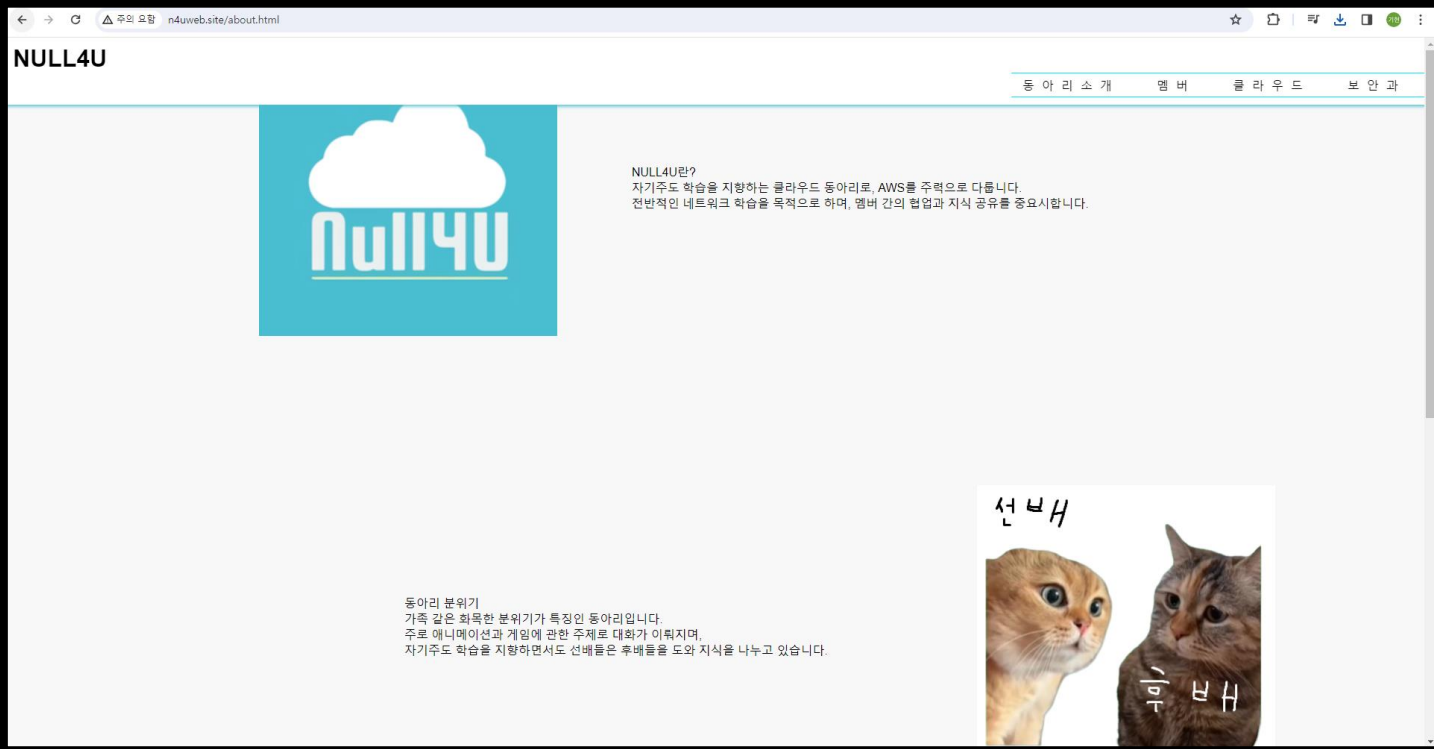
  <div class="image-background">
    
  </div>

  <section id="about">
    <h2>동아리 소개</h2>
    <p>자기주도 학습을 지향하는 클라우드 동아리 AWS를 주력으로 다루며 전반적인 네트워크 학습을 목적으로 한다</p>
  </section>

  <section id="members">
    <h2>멤버</h2>
    <pre>
      3학년: 최지혁 이현우 오현서 김성훈
      2학년: 구민희 박승빈 현성 현욱 이도경
      1학년: 김기현 양지호 노니콜라이
    </pre>
  </section>

  <section id="projects">
    <h2>클라우드</h2>
    <ul>
      <li>IaaS</li>
      <li>PaaS</li>
      <li>SaaS</li>
      <!-- 추가 프로젝트 목록 -->
    </ul>
  </section>
```

동아리소개



```
<body>
  <header>
    <h1><a href="http://www.n4uweb.site">NULL4U</a></h1>
    <nav>
      <ul>
        <li><a href="http://www.n4uweb.site/about.html">동아리소개</a></li>
        <li><a href="http://www.n4uweb.site/members.html">멤버</a></li>
        <li><a href="http://www.n4uweb.site/aws.html">클라우드</a></li>
        <li><a href="https://smc-secu.net/">보안과</a></li>
      </ul>
    </nav>
  </header>

  <div class="image-container">
    
    <div class="description">
      <p>NULL4U란?<br>
      자기주도 학습을 지향하는 클라우드 동아리로, AWS를 주력으로 다룹니다.<br>
      전반적인 네트워크 학습을 목적으로 하며, 멤버 간의 협업과 지식 공유를 중요시합니다.</p>
    </div>
  </div>

  <div id="image-container2">
    
    <div class="description">
      <p>동아리 분위기<br>
      가장 깊은 화목한 분위기가 특징인 동아리입니다.<br>
      주로 애니메이션과 게임에 관한 주제로 대화가 이뤄지며,<br>
      자기주도 학습을 지향하면서도 선배들은 후배들을 도와 지식을 나누고 있습니다.</p>
    </div>
  </div>

  <div class="image-container">
    
    <div class="description">
      <p>AWS란?<br>
      Amazon Web Services는 아마존(Amazon)에서 제공하는 클라우드 서비스로,<br>
      네트워킹을 기반으로 가상 컴퓨터, 스토리지, 네트워크 인프라 등 다양한 서비스를<br>
      제공합니다.<br>
      비즈니스와 개발자가 웹 서비스를 사용하여 확장 가능하고 정교한 애플리케이션을 구<br>
      축하는 데 도움을 줍니다.</p>
    </div>
  </div>
</body>
```

멤버

```
<body>
  <header>
    <h1><a href="http://www.n4uweb.site">NULL4U</a></h1>
    <nav>
      <ul>
        <li><a href="http://www.n4uweb.site/about.html">동아리소개</a></li>
        <li><a href="http://www.n4uweb.site/members.html">멤버</a></li>
        <li><a href="http://www.n4uweb.site/aws.html">클라우드</a></li>
        <li><a href="https://smc-secu.net/">보안과</a></li>
      </ul>
    </nav>
  </header>

  <section id="members">
    <h2>멤버</h2>
    <h1><ul><li><a href="https://originalchoi.tistory.com/">선생님:최정훈</a></li>
  </ul></h1>
    <ul>
      <li>3학년: 최지혁 이현우 오현서 김성훈</li>
    </ul>
    <ul>
      <li>2학년: 구민희 박승빈 현성 현욱 이도경</li>
    </ul>
    <ul>
      <li>1학년: 김기현 양지호 노니콜라이</li>
    </ul>
  </section>
```

NULL4U

[동아리 소개](#) [멤버](#) [클라우드](#) [보안과](#)

멤버

선생님 : 최정훈

3 학년 : 최지혁 이현우 오현서 김성훈


2 학년 : 구민희 박승빈 현성 현욱 이도경

1 학년 : 김기현 양지호 노니콜라이

클라우드

NULL4U

동 아 리 소 개 멤 버 클 라 우 드 보 안 과



클라우드 컴퓨팅

인터넷을 통해서 원격에 있는 고성능의 컴퓨터 자원을 빌려서 IT 서비스를 할 수 있는 환경

클라우드 서비스를 이용하는 방식에 따라 크게 3가지로 구분된다

IaaS(Infrastructure as a Service) 인프라 서비스 cpu, memory, network, storage 같은 하드웨어적 장비를 가상화해서 유연하게 제공하는 서비스입니다	PaaS(Platform as a Service) 플랫폼 환경 제공 서비스 웹 서비스나 어플리케이션을 만들고 배포할 수 있는 개발 환경을 제공하는 서비스입니다	SaaS(Software as a Service) 응용프로그램 제공 서비스 클라우드에서 제공하는 웹 서비스나 어플리케이션을 그대로 이용하는 것 완전한 소프트웨어 솔루션을 사용하는 것이라고 생각하면 됩니다 구글의 gmail이나 Microsoft Office 365 같은 예입니다
--	---	---

```
<body>
  <header>
    <h1><a href="http://www.n4uweb.site">NULL4U</a></h1>
    <nav>
      <ul>
        <li><a href="http://www.n4uweb.site/about.html">동아리소개</a></li>
        <li><a href="http://www.n4uweb.site/members.html">멤버</a></li>
        <li><a href="http://www.n4uweb.site/aws.html">클라우드</a></li>
        <li><a href="https://smc-secu.net/">보안과</a></li>
      </ul>
    </nav>
  </header>

  <!-- 이미지 및 설명 컨테이너 -->
  <div class="image-container3">
    <!-- 이미지 -->
    
    <!-- 설명 -->
    <p>클라우드 컴퓨팅</p>
    <p>인터넷을 통해서 원격에 있는 고성능의 컴퓨터 자원을 빌려서 IT 서비스를 할 수 있는 환경</p>
    <p>클라우드 서비스를 이용하는 방식에 따라 크게 3가지로 구분된다</p>

    <div class="description">
      <div>
        IaaS(Infrastructure as a Service)<br>
        인프라 서비스<br>
        cpu, memory, network, storage 같은 하드웨어적 장비를<br>
        가상화해서 유연하게 제공하는 서비스입니다
      </div>
      <div>
        PaaS(Platform as a Service)<br>
        플랫폼 환경 제공 서비스<br>
        웹 서비스나 어플리케이션을 만들고 배포할 수 있는<br>
        개발 환경을 제공하는 서비스입니다
      </div>
      <div>
        SaaS(Software as a Service)<br>
        응용프로그램 제공 서비스<br>
        클라우드에서 제공하는 웹 서비스나 어플리케이션을 그대로 이용하는 것<br>
        완전한 소프트웨어 솔루션을 사용하는 거라고 생각하면 됩니다<br>
        구글의 gmail이나 Microsoft Office 365 같은 예입니다
      </div>
    </div>
  </div>
</body>
```


아쉬웠던점

웹 이란걸 처음 만들어보기도 하고 미숙해서 웹에 내용이나 기능이 부족한 것 같았습니다

웹디자인을 알았던거 같았습니다



웹 만들기 전



만들고 난 뒤

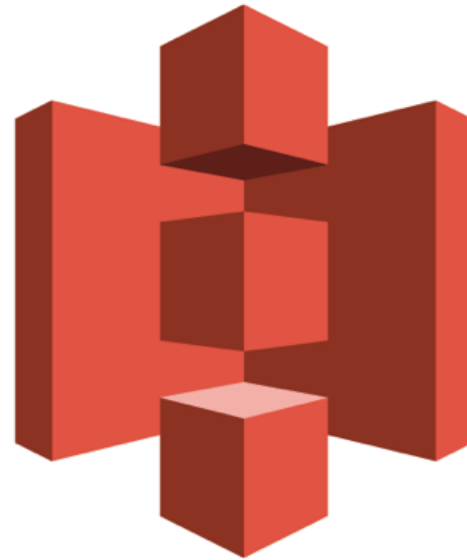
웹 배포 과정

배포 수단



S3란?

S3란 Simple Storage Service의 약자로 확장성, 데이터 가용성 및 보안과 성능을 제공하는 객체 스토리지 서비스입니다.



Amazon S3

Bucket이란?

버킷이란 S3 저장 서비스의 기본 저장 단위입니다. 버킷은 FTP 서버 등과 비슷하게 서버 내에 여러 폴더나 객체 (파일) 등을 담을 수 있는 구조입니다.

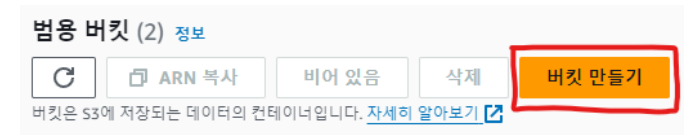
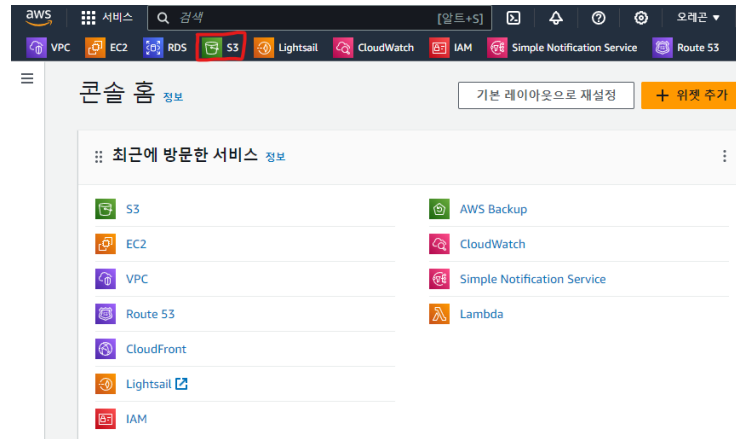


Route53이란?

Route53은 AWS에서 제공하는 DNS 서비스로
특징으로는 특이 레코더인 ALIAS(별칭) 이 있습니다.

배포 과정

1. 버킷 생성



배포 과정

2. 세부 설정

이처럼 설정해야 웹을 퍼블릭으로 배포할 수 있습니다.

이 버킷의 퍼블릭 액세스 차단 설정

퍼블릭 액세스는 ACL(액세스 제어 목록), 버킷 정책, 액세스 지점 정책 또는 모두를 통해 버킷 및 객체에 부여됩니다. 이 버킷 및 해당 객체에 대한 퍼블릭 액세스가 차단되었는지 확인하려면 모든 퍼블릭 액세스 차단을 활성화합니다. 이 설정은 이 버킷 및 해당 액세스 지점에만 적용됩니다. AWS에서는 모든 퍼블릭 액세스 차단을 활성화하도록 권장하지만, 이 설정을 적용하기 전에 퍼블릭 액세스가 없어도 애플리케이션이 올바르게 작동하는지 확인합니다. 이 버킷 또는 내부 객체에 대한 어느 정도 수준의 퍼블릭 액세스가 필요한 경우 특정 스토리지 사용 사례에 맞게 아래 개별 설정을 사용자 지정할 수 있습니다. [자세히 알아보기](#)

모든 퍼블릭 액세스 차단

이 설정을 활성화하면 아래 4개의 설정을 모두 활성화한 것과 같습니다. 다음 설정 각각은 서로 독립적입니다.

새 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3은 새로 추가된 버킷 또는 객체에 적용되는 퍼블릭 액세스 권한을 차단하며, 기존 버킷 및 객체에 대한 새 퍼블릭 액세스 ACL 생성을 금지합니다. 이 설정은 ACL을 사용하여 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 권한을 변경하지 않습니다.

임의의 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 모든 ACL을 무시합니다.

새 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 새 버킷 및 액세스 지점 정책을 차단합니다. 이 설정은 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 정책을 변경하지 않습니다.







임의의 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 및 교차 계정 액세스 차단

S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 정책을 사용하는 버킷 또는 액세스 지점에 대한 퍼블릭 및 교차 계정 액세스를 무시합니다.

배포 과정

3. 그 후 버킷에 웹을 구성하는 파일 등을 업로드 합니다.

EX)

<input type="checkbox"/>	이름 ▲	유형 ▼	마지막 수정 ▼	크기 ▼	스토리지 클래스 ▼
<input type="checkbox"/>	 about.html	html	2024. 1. 4. pm 4:40:48 PM KST	5.3KB	Standard
<input type="checkbox"/>	 aws.html	html	2024. 1. 4. pm 4:40:51 PM KST	6.6KB	Standard
<input type="checkbox"/>	 index.html	html	2024. 1. 4. pm 4:40:50 PM KST	3.5KB	Standard
<input type="checkbox"/>	 members.html	html	2024. 1. 4. pm 4:40:50 PM KST	2.9KB	Standard
<input type="checkbox"/>	 null4uweppage/	폴더	-	-	-
<input type="checkbox"/>	 styles.css	css	2024. 1. 4. pm 4:44:06 PM KST	5.9KB	Standard

배포 과정

이러한 버킷 정책은 버킷을 퍼블릭으로 전환하기 위한 것입니다.

버킷 정책

[편집](#)[삭제](#)

JSON으로 작성된 버킷 정책은 버킷에 저장된 객체에 대한 액세스 권한을 제공합니다. 버킷 정책은 다른 계정이 소유한 객체에는 적용되지 않습니다. [자세히 알아보기](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadGetObject",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::www.n4uweb.site/*"
    }
  ]
}
```

[복사](#)

배포 과정

이 설정을 통해 정적
웹 사이트 호스팅을
활성화 합니다.

정적 웹 사이트 호스팅

이 버킷을 사용하여 웹 사이트를 호스팅하거나 요청을 리디렉션합니다. [자세히 알아보기](#)

정적 웹 사이트 호스팅

- 비활성화
- 활성화

호스팅 유형

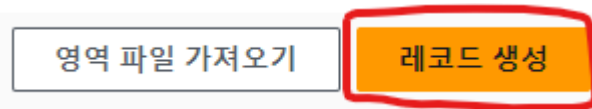
- 정적 웹 사이트 호스팅
버킷 엔드포인트를 웹 주소로 사용합니다. [자세히 알아보기](#)
- 객체에 대한 요청 리디렉션
요청을 다른 버킷 또는 도메인으로 리디렉션합니다. [자세히 알아보기](#)

배포 과정

1. 호스팅 영역을 생성합니다



2. 그 후 레코드를 생성 해줍니다.



배포 과정

3. 레코드 유형은 A - IPv4 주소 및 일부 AWS 리소스... 를 선택하여 주고 별칭을 활성화합니다.

레코드 유형 | 정보

A - IPv4 주소 및 일부 AWS 리소스로 트래픽 라우팅

4. 그 후 트래픽 라우팅 대상에서 S3를 선택한 후 본인이 S3를 만든 리전을 선택하고 본인의 S3 엔드포인트를 선택합니다.

트래픽 라우팅 대상 | 정보

S3 웹 사이트 엔드포인트에 대한 별칭

미국 서부(오레곤)

Q S3 엔드포인트 입력

DNS 추가

DNS란?

DNS란? Domain Name System의 약어로 사용자가 숫자로 된 인터넷 프로토콜 주소 대신 우리들이 흔히 알고 있는 EX) www.naver.com와 같은 문자로 된 주소로 바꾸어 주는 서비스입니다.

DNS 추가 과정

1. DNS 구매 및 설정
2. S3 엔드포인트와 DNS 연결 과정

DNS 구매 및 설정

도메인을 구매할 수 있는 사이트에서 도메인을 구매해주고 ex) 가비아 등등

도메인을 구매한 사이트에서 도메인 관리에 들어가 추가적인 설정을 해줍니다.

DNS 네임서버 설정

도메인을 구매한 사이트에서 도메인의 네임서버를 설정합니다.

네임서버 **설정**

1차	ns- [redacted]
2차	ns- [redacted]
3차	ns- [redacted]
4차	ns- [redacted]
5차	데이터 없음
6차	데이터 없음
7차	데이터 없음

1차 ~ 4차 네임서버에는 Route 53을 생성할 때 만들어졌던 레코드에 설정 되어있던 값을 입력합니다.



n4uweb.site

NS

단순

-

아니요

ns- [redacted]
ns- [redacted]
ns- [redacted]
ns- [redacted]

네임서버 설정 이유

위와 같이 네임서버 설정을 변경한 이유는

외부 도메인 업체에서 구매한 도메인을

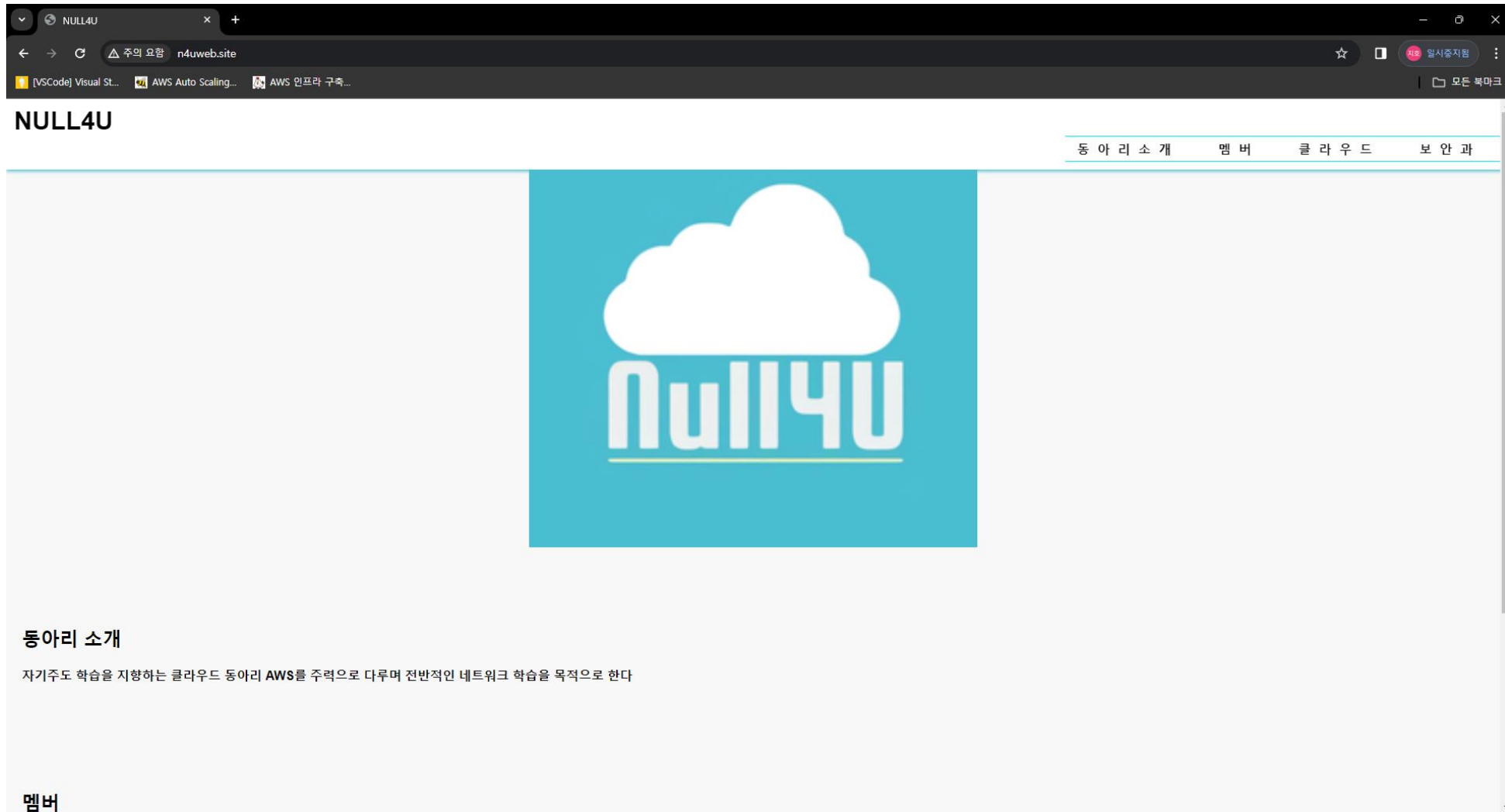
Route53으로 이전하기 위해서 입니다.

결과

모든 설정을 완료한 후 일반적으로 24 ~ 48 TTL(Time To Live) 정도가 지나면 적용됩니다.

TTL : 컴퓨터나 네트워크에서 데이터의 유효 기간을 나타내기 위한 방법

실제 실행 화면



아쉬웠던 점 & 개선할 점

1. EC2 인스턴스에서 HTTPD를 이용해 서버를 구축한 후 배포를 시도해 봤지만 실패하였습니다.
2. 이번 프로젝트를 통해 지금으로써는 더 좋은 배포 방식을 사용하기에는 실력이 부족하다는 것을 깨달았습니다.
3. HTTPS로 배포한 것이 아니기에 보안적으로 취약하다는 것이 아쉬웠고 개선해야할 점이라 생각합니다.

마무리 인사

부족한 발표를 들어주셔서 감사합니다.