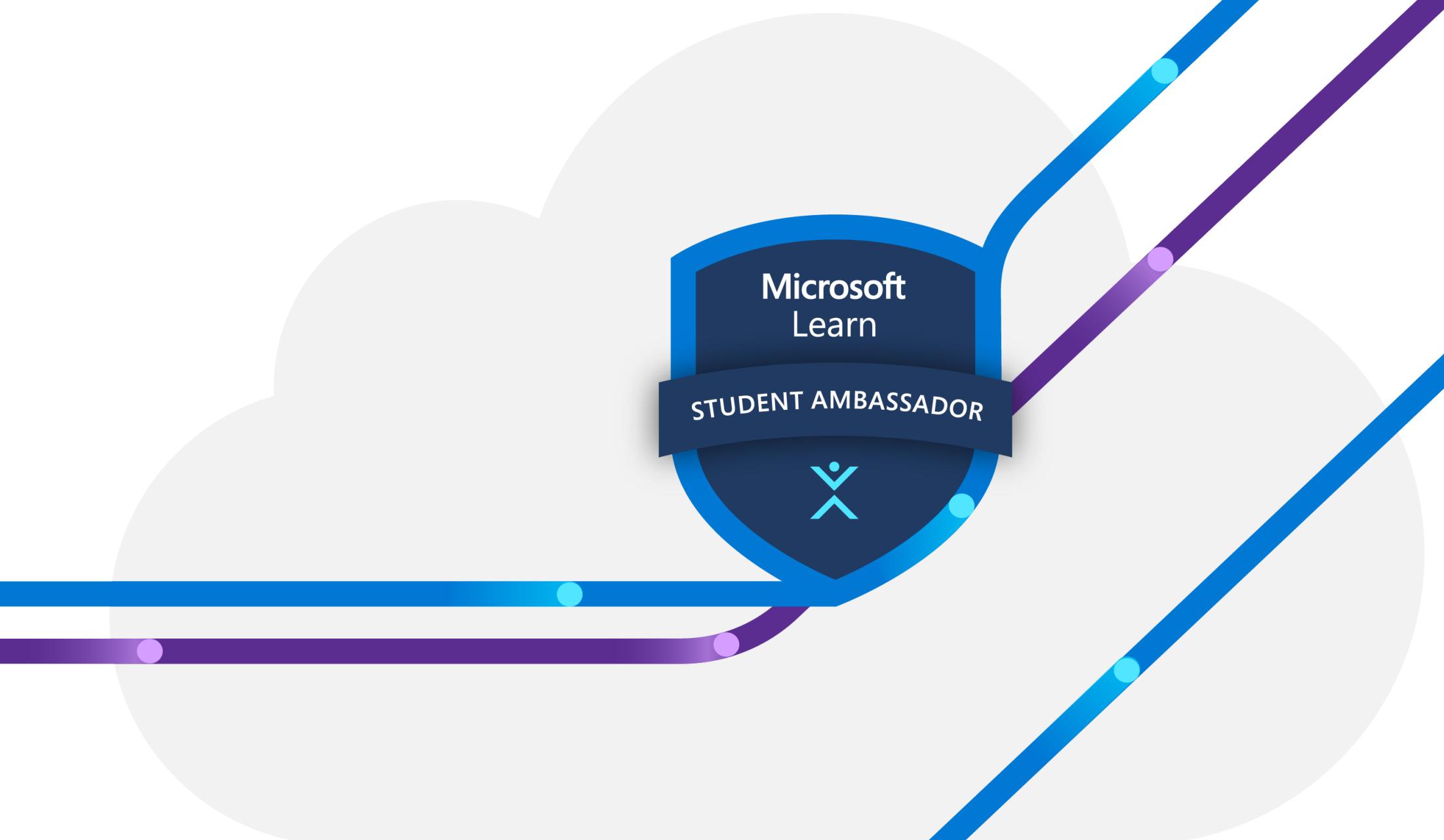


Azure에 SpringBoot 마이크로서비스 애플리케이션 배포 찍어 먹기

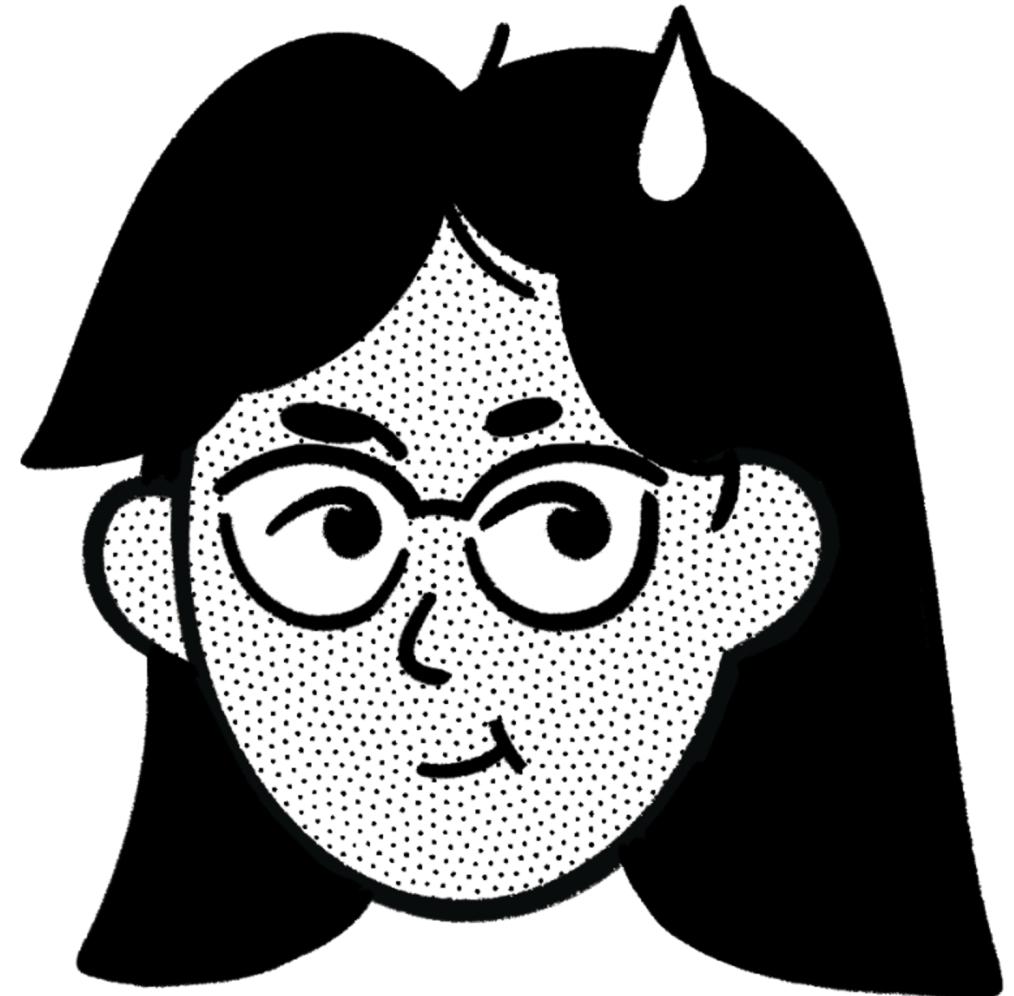
MSA Beta 오지수



Hi! I'm Jisoo!

숙명여자대학교 컴퓨터과학전공

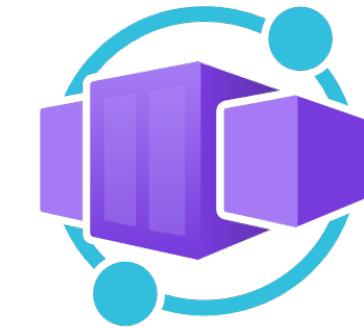
Microsoft Student Ambassador



주제

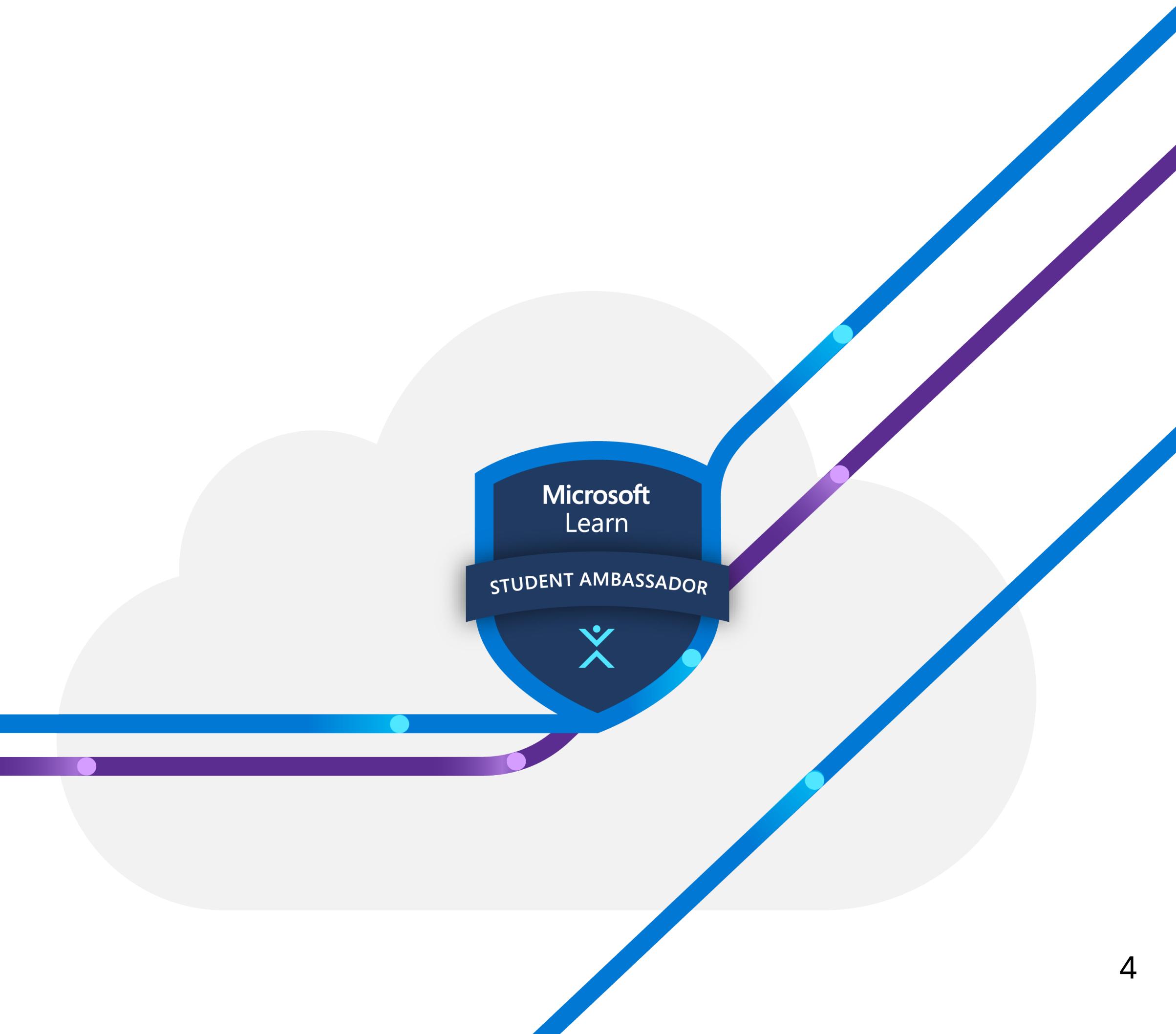
✗ Spring Cloud를 통해 MSA를 설계/제작하기 ✗

✓ Spring Cloud 프로젝트를 **Azure Container Apps**로 배포해보기! ✓



01.

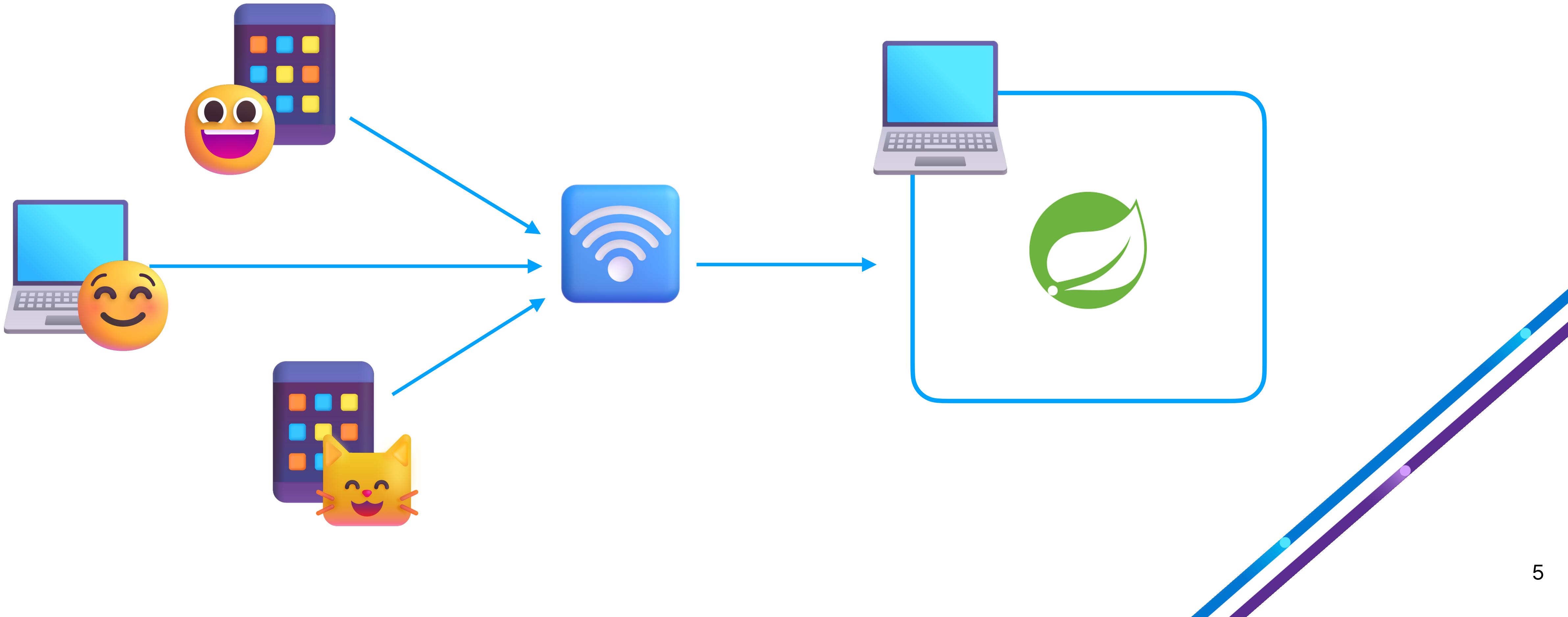
배경 지식



배포

웹 애플리케이션을 배포하는 것?

유저들이 다른 곳에서도 내 앱에 접근할 수 있도록 하는 것!

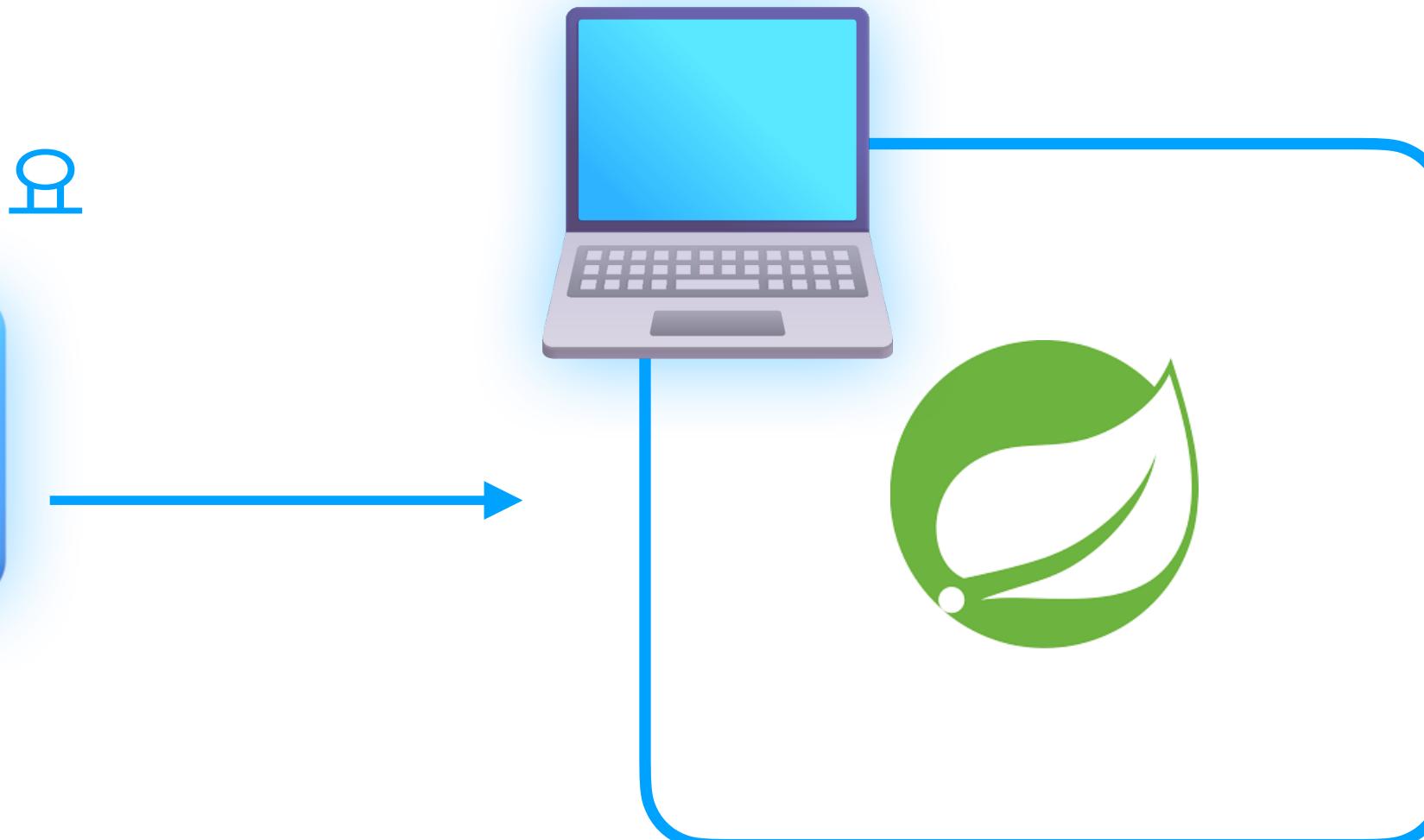


배포

웹 애플리케이션을 배포하는 것?

유저들이 다른 곳에서도 내 앱에 접근할 수 있도록 하는 것!

접근 가능한 링크 필요



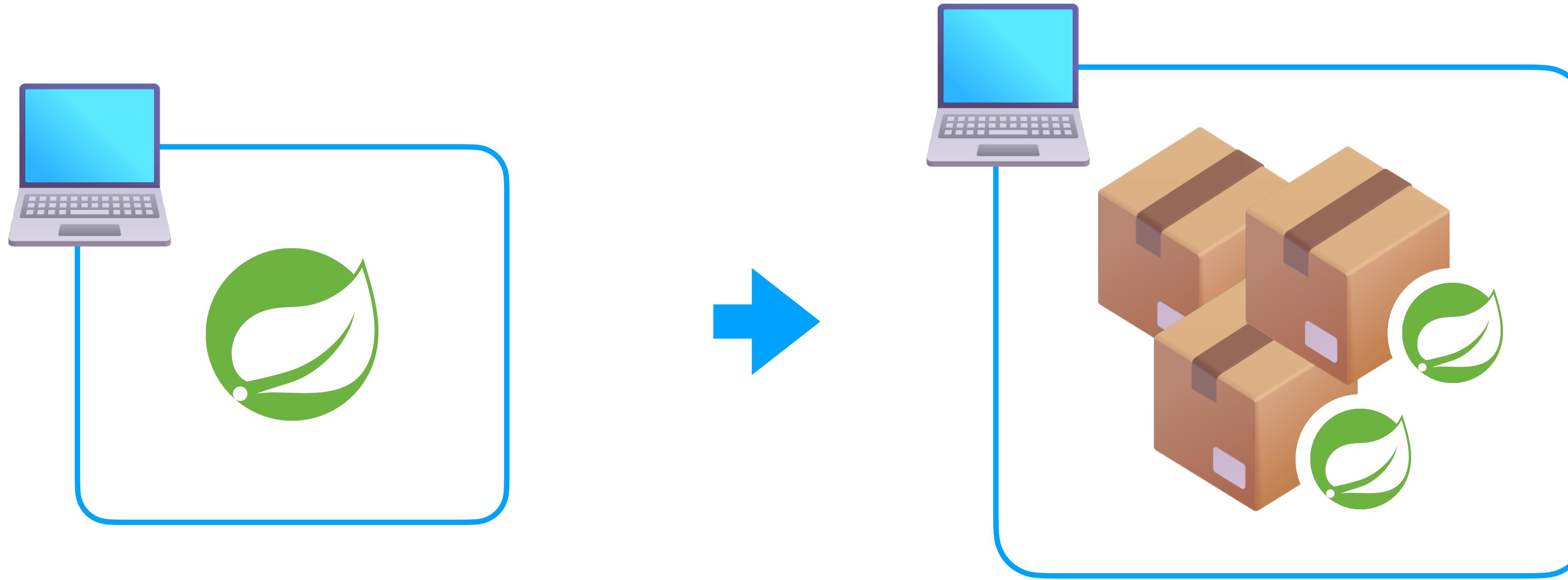
웹앱을 계속 실행할 컴퓨터(서버)가 필요

⇒ 클라우드 자원 사용: Virtual Machine

컨테이너

프로그램이 환경에 구애받지 않고 계속 실행될 수 있는 독립적인 공간

⇒ 운영체제에서 실행되는 프로세스를 '격리'하여 별도의 실행(운영) 환경을 제공하는 것.

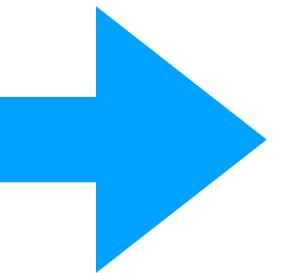


컨테이너

'이미지'를 빌드 명령을 통해 컨테이너를 생성할 수 있음.



스프링부트 이미지



스프링부트 컨테이너

이미지

컨테이너 내에서 애플리케이션을 실행하는 데 필요한
소스코드 및 라이브러리, 환경 설정, 컴파일된 실행파일 등 모든 것을 포함하는 템플릿



이미 만들어진 것을 사용



Docker file을 통해 직접 생성



Azure Spring Apps

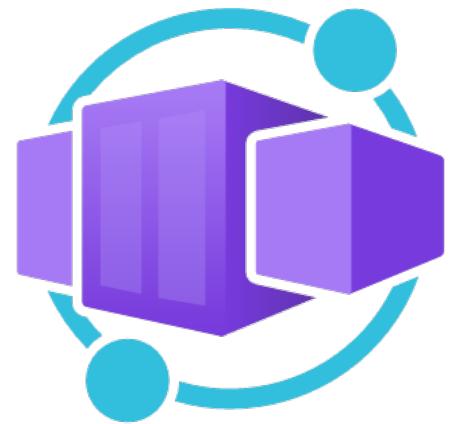
Spring 개발자가 인프라가 아닌 코드에 집중할 수 있도록 도와주는 완전 관리형 서비스

Kubernetes 없이 웹앱, 마이크로 서비스, 서비스를 비롯한 모든 유형의 Spring 앱을 배포할 수 있도록 지원

Azure Spring Apps 사용 중지 공지

2025. 02. 25.

Azure Spring Apps는 Microsoft와 VMware by Broadcom이 공동으로 빌드한 Java Spring 애플리케이션을 실행하기 위한 완전 관리형 서비스입니다. 신중하게 고려하고 분석한 후 Microsoft와 Broadcom은 Azure Spring Apps 서비스를 사용 중지하기로 어려운 결정을 내렸습니다. Azure Spring Apps에서 실행되는 워크로드 마이그레이션을 위한 대체 서비스로 Azure Container Apps 및 AKS(Azure Kubernetes Service)를 사용하는 것이 좋습니다. Microsoft는 서비스 종료를 통해 마이그레이션 노구, 전문가 리소스 및 기술 지원을 통해 장기 플랫폼으로 사용자를 지원하기 위해 최선을 다하고 있습니다.



Azure Container Apps

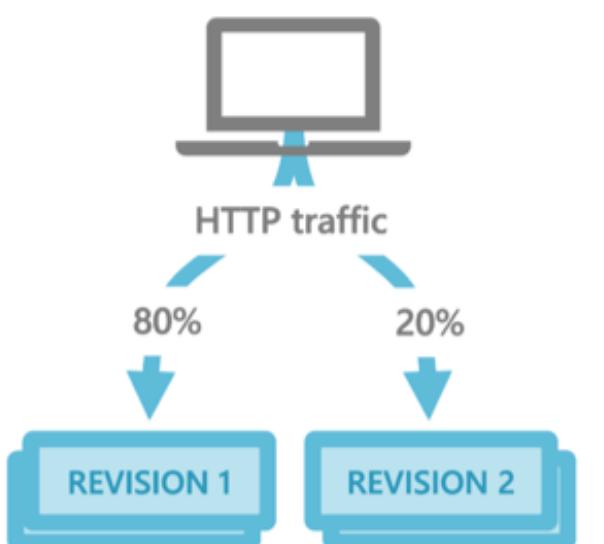
컨테이너화된 애플리케이션을 실행하는 동안 인프라를 줄이고 비용을 절감할 수 있는 서비스 플랫폼

→ 자동 크기 조정을 제공하고, 운영 복잡성을 줄이며, **인프라가 아닌 애플리케이션에 집중할 수 있도록 함**

일반적인 용도

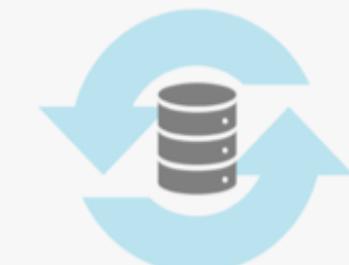
1. API 엔드포인트 배포
2. 후순위 처리 작업 호스팅
3. 이벤트 기반 처리 수행
4. 마이크로서비스 실행

PUBLIC API ENDPOINTS



HTTP requests are split between two versions of the container app where the first revision gets 80% of the traffic, while a new revision receives the remaining 20%.

BACKGROUND PROCESSING



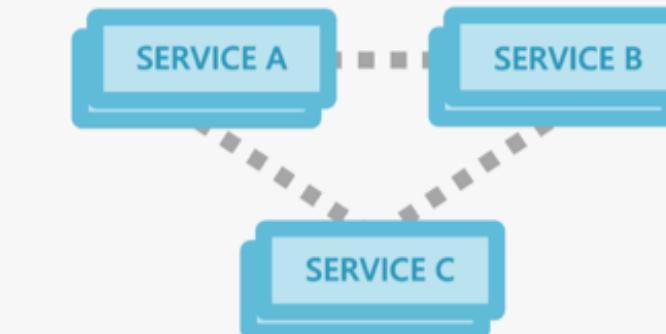
A background job that transforms data in a database.

EVENT-DRIVEN PROCESSING



A queue reader application that processes messages as they arrive in a queue.

MICROSERVICES



Deploy and manage a microservices architecture with the option to integrate with Dapr.

AUTO-SCALE CRITERIA

Scaling is determined by the number of concurrent HTTP requests.

AUTO-SCALE CRITERIA

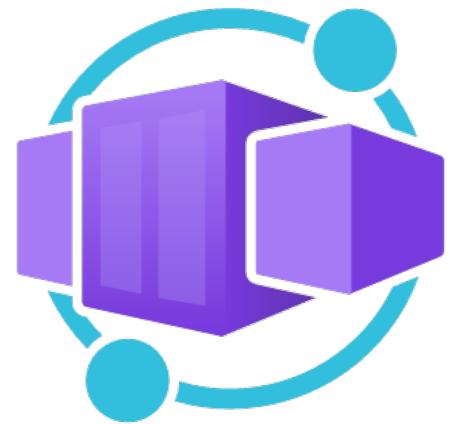
Triggered by a schedule, on demand, or based on events.

AUTO-SCALE CRITERIA

Scaling is determined by the number of messages in the queue.

AUTO-SCALE CRITERIA

Individual microservices can scale according to any KEDA scale triggers.

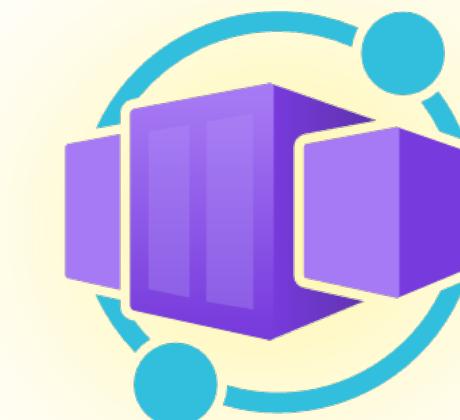
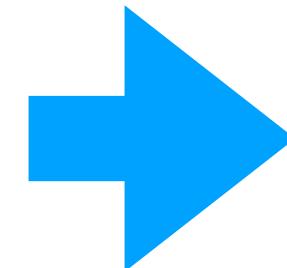


Azure Container Apps

컨테이너화된 애플리케이션을 실행하는 동안 인프라를 줄이고 비용을 절감할 수 있는 서비스 플랫폼

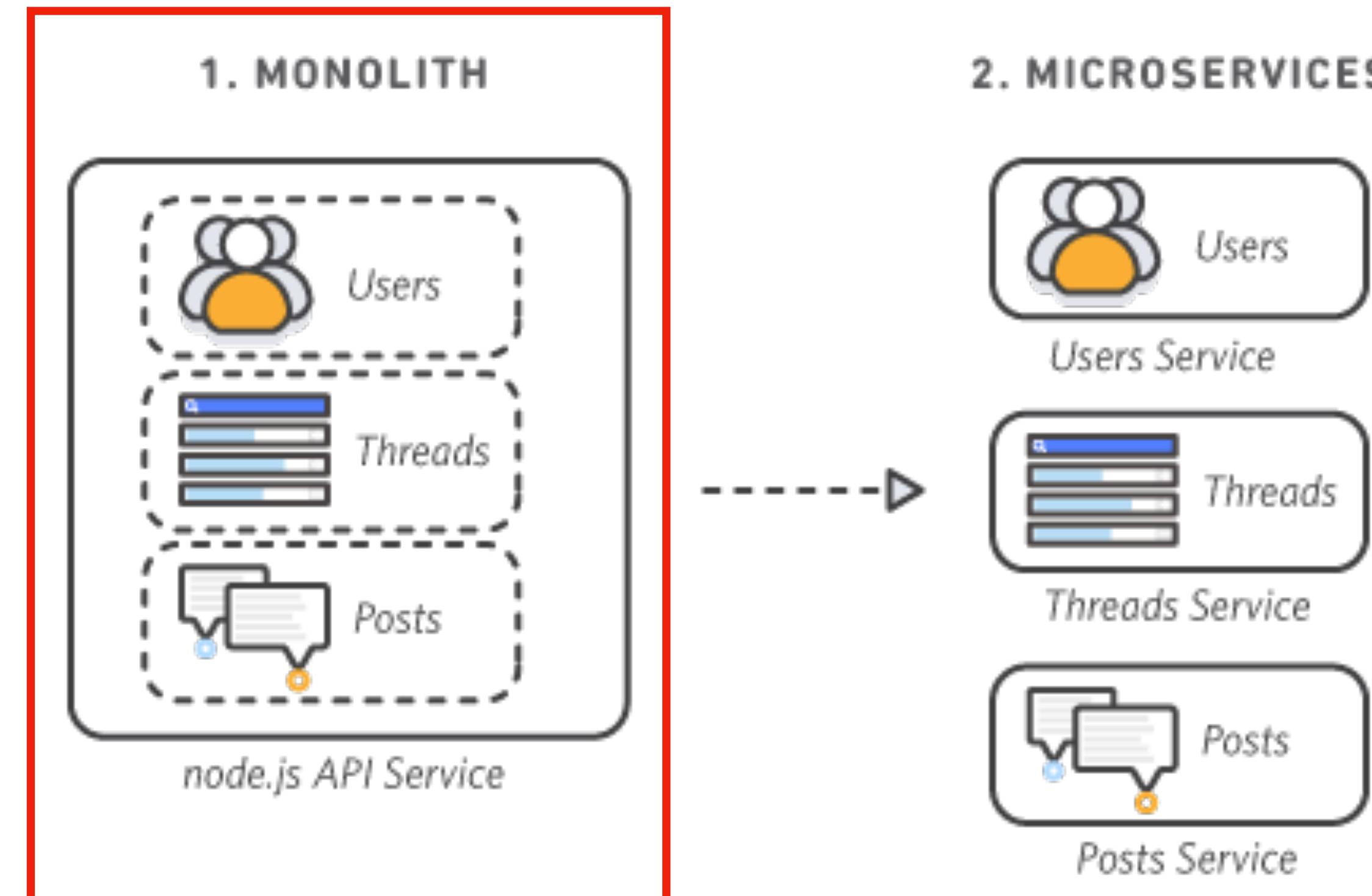
→ 자동 크기 조정을 제공하고, 운영 복잡성을 줄이며, **인프라가 아닌 애플리케이션에 집중할 수 있도록 함**

1. VM 생성 후, 접속
2. 환경 세팅 (docker 설치 등)
3. Docker 로그인하고,
이미지 가져오기
4. 컨테이너 실행
5. 네트워크/방화벽 설정
6. 등등..



모놀리식 아키텍처

하나의 코드 베이스를 사용하여 여러 비즈니스 기능을 수행하는 전통적인 소프트웨어 개발 모델

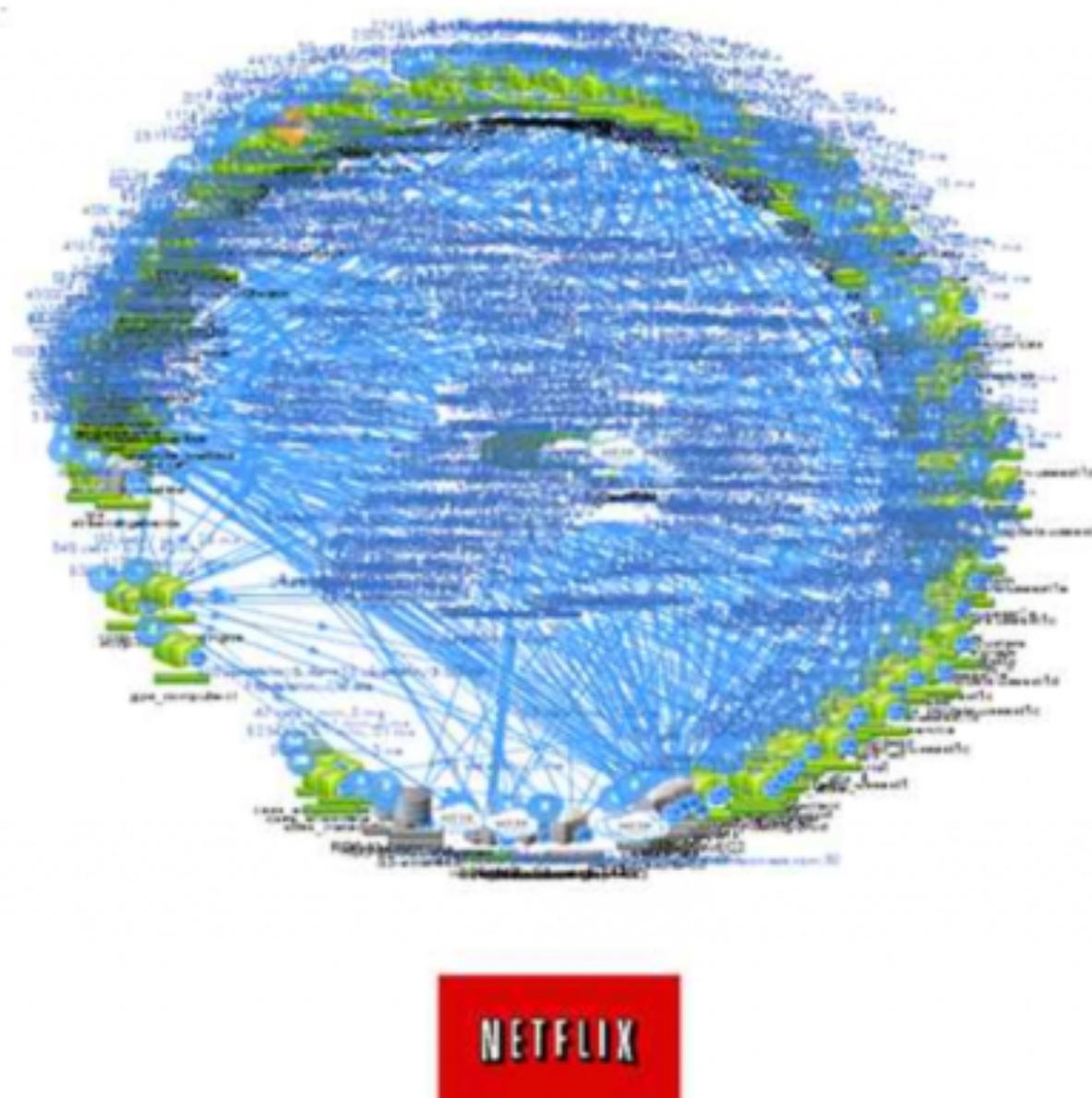
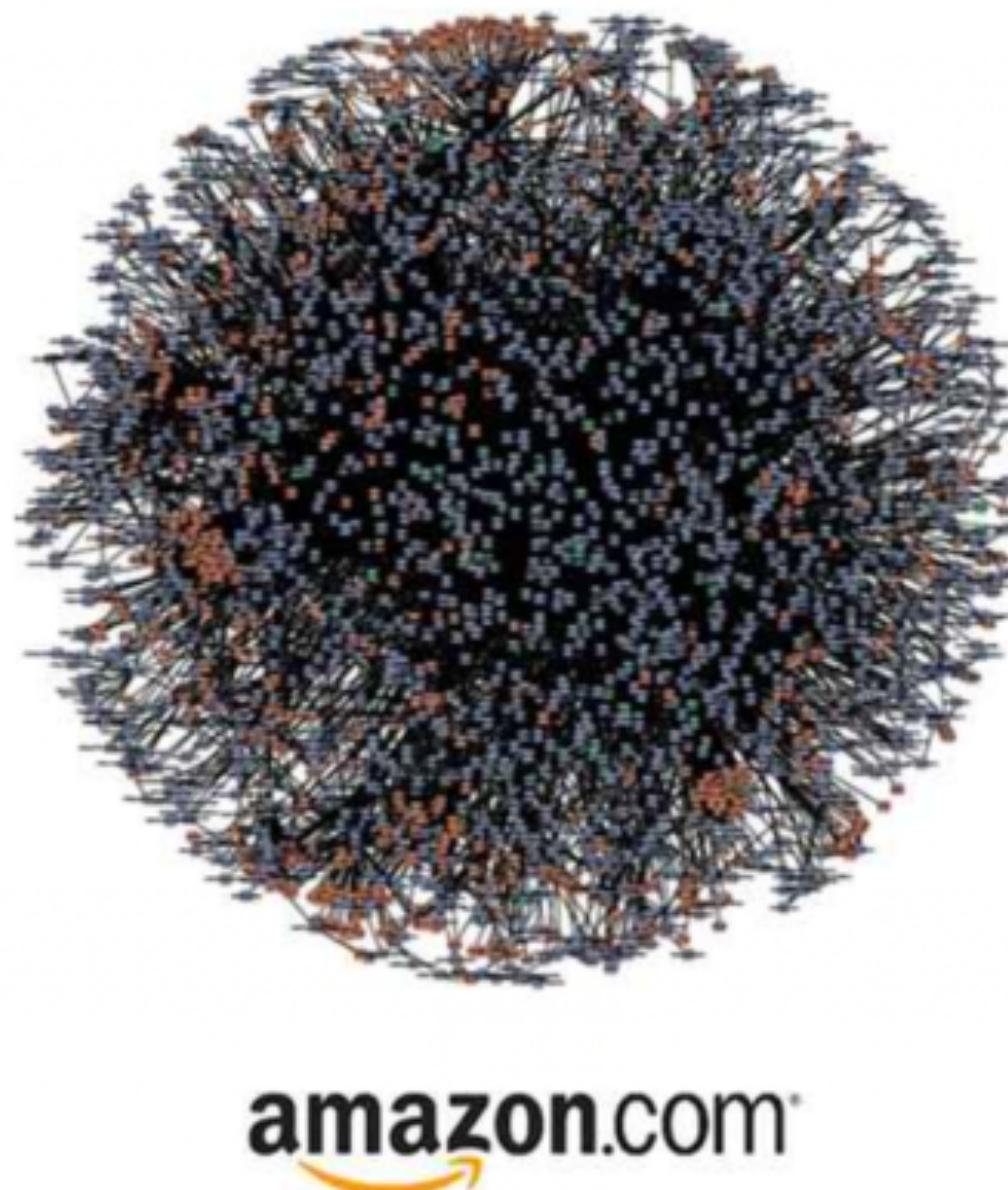


한 서버가 여러 기능 수행

마이크로서비스 아키텍처

소프트웨어를 작은 독립 구성 요소 또는 서비스로 구성하는 아키텍처 접근 방식

→ 하나의 애플리케이션을 다수의 독립적인 서비스들의 집합으로 구성



- 각 서비스는 **하나의 기능을 수행하고**,
잘 정의된 **인터페이스**를 통해 다른 서비스와 통신함.
- **독립적으로 실행됨**
 - 필요에 따라 서비스를 업데이트, 수정, 배포 가능
 - 한 요소의 오류로 인해 전체 앱이 중단되지 않음

모놀리식 vs 마이크로서비스

모놀리식은 간단하지만 수정, 크기 조정에 불리하고,
마이크로서비스는 유연성이 좋지만, 복잡함.

⇒ 따라서 장, 단점을 확인 후 도입을 고려해야 함.

	모놀리식	마이크로서비스
개발 과정	사전 계획이 많이 필요하지 않음. → 비교적 간단함.	많은 계획 및 설계 필요
배포	코드 베이스, 종속성을 단일 환경에 설치 → 배포가 간단함.	독립적으로 배포 가능해야 함. → 컨테이너화 후 배포 함.
디버깅	동일한 환경 내에서 검사 가능	연결된 서비스들을 모두 살펴봐야 함.
수정	코드가 긴밀하게 연결 → 여러 기능이 영향을 받음. 수정 후 전체 시스템 테스트 필요	특정 기능만 수정하고, 특정 서비스만 독립적 배포 가능 → 잦은 변경이 필요한 서비스에 유리
크기 조정	전체 애플리케이션의 크기 조정 필요	독립적인 크기 조정이 가능함

02.

예시 프로젝트 - Pet Clinic



예시 애플리케이션

PetClinic 애플리케이션

스프링 공식 예제 프로젝트

→ '동물 병원'을 예시로 한 간단한 웹 애플리케이션

<https://github.com/spring-projects/spring-petclinic>

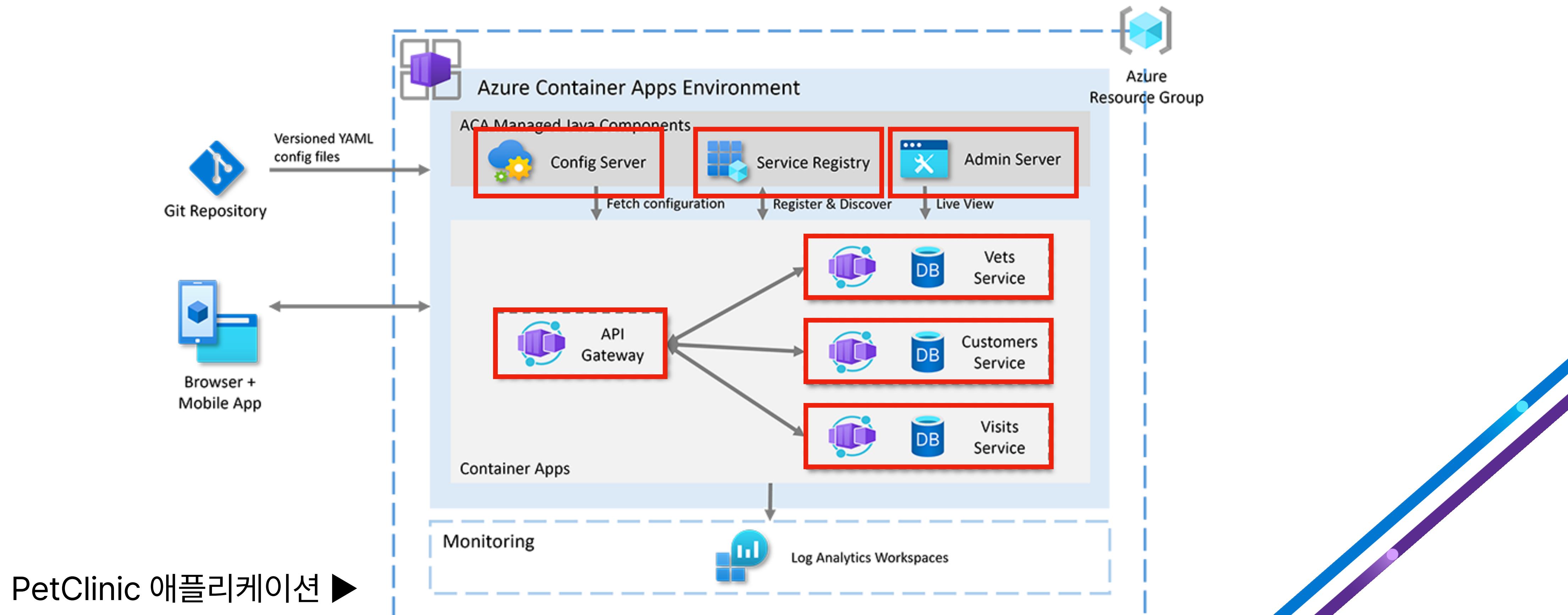
The screenshot shows the GitHub repository page for 'spring-projects/spring-petclinic'. The repository is public and has 368 watchers, 26.4k forks, and 8.5k stars. It contains 8 branches and 1 tag. The main branch has 985 commits. The repository description is 'A sample Spring-based application'. The 'About' section includes links to Readme, Apache-2.0 license, Code of conduct, Security policy, Activity, Custom properties, 8.5k stars, 368 watching, 26.4k forks, and a Report repository. The commit history shows several recent updates related to Maven and Gradle version upgrades, Dockerfile additions, and database upgrades.

Commit	Message	Date
.devcontainer	Add a Dockerfile for dev environments other than codeship	9 months ago
.github	Add dco config	6 months ago
.mvn/wrapper	Maven and Gradle version upgrades	last month
gradle/wrapper	Maven and Gradle version upgrades	last month
k8s	Upgraded databases	2 months ago
src	Fix typo in visits form which caused button to miss its text	2 months ago
.editorconfig	Add Gradle files indentation to .editorconfig	last year

Pet Clinic 앱

Config / Eureka / Admin Server와

API Gateway, Vets / Customers / Visits Service 요소가 독립적인 프로젝트를 만들어볼 예정!



사전 준비

Azure CLI 설정 과정

```
brew update && brew install azure-cli
```

Azure CLI 설치

```
az version
```

설치 확인

```
[jisoo@ojisuui-MacBookPro ~ % az version
{
    "azure-cli": "2.76.0",
    "azure-cli-core": "2.76.0",
    "azure-cli-telemetry": "1.1.0",
    "extensions": {
        "containerapp": "1.2.0b3"
    }
}
jisoo@ojisuui-MacBookPro ~ %
```

사전 준비

Azure Login

```
az login
```

로그인 후 구독/테넌트 선택하기

```
jisoo@ojisuui-MacBookPro ~ % az login
[A web browser has been opened at https://login.microsoftonline.com/organizations/oauth2/v2.0/authorize. Please continue to the login in the web browser. If no web browser is available or if the web browser fails to open, use device code flow with 'az login --use-device-code'.
Retrieving tenants and subscriptions for the selection...
[Tenant and subscription selection]
No Subscription name Subscription ID Tenant
----- ----- -----
[1] * Visual Studio Enterprise 구독 Student Ambassadors
The default is marked with an *: the default tenant is 'Student Ambassadors' and subscription is 'Visual Studio Enterprise 구독'
Select a subscription and tenant (Type a number or Enter for no changes):
Tenant: Student Ambassadors
Subscription: Visual Studio Enterprise 구독
[Announcements]
With the new Azure CLI login experience, you can select the subscription you want to use more easily. Learn more about it and its configuration at https://go.microsoft.com/fwlink/?linkid=2271236
If you encounter any problem, please open an issue at https://aka.ms/azclibug
[Warning] The login output has been updated. Please be aware that it no longer displays the full list of available subscriptions by default.
jisoo@ojisuui-MacBookPro ~ %
```

```
az extension add --name containerapp --upgrade --allow-preview
```

Azure Container Apps CLI Extension 설치

리소스 그룹 만들기

리소스 그룹 ? Azure 솔루션과 관련된 리소스를 보관하는 컨테이너



리소스 그룹 만들기

기본 태그 검토 + 만들기

리소스 그룹 - Azure 솔루션의 관련 리소스를 보관하는 컨테이너입니다. 리소스 그룹에 솔루션의 모든 리소스를 포함할 수도 있고 그룹으로 관리할 리소스만 포함할 수도 있습니다. 무엇이 조직에 가장 적합한지에 따라 리소스 그룹에 리소스를 할당할 방법을 결정합니다. [자세한 정보](#)

구독 * ⓘ Visual Studio Enterprise 구독

리소스 그룹 이름 * ⓘ first-msa

지역 * ⓘ (Asia Pacific) Korea Central

```
az group create \
--name first-msa \
--location koreacentral
```

리소스 그룹 만들기

리소스 그룹 ? Azure 솔루션과 관련된 리소스를 보관하는 컨테이너

The screenshot shows the Microsoft Azure Resource Groups management interface. On the left, there is a sidebar titled "리소스 그룹" (Resource Groups) showing a list of existing resource groups: "appsvc_windows_centralus", "appsvc_windows_centralus_basic", "coaster", "first-msa", "first-msa", and "NetworkWatcherRG". The second "first-msa" entry is highlighted with a blue arrow pointing to it. On the right, the main pane displays the "first-msa" resource group details. The top navigation bar includes links for "리소스 그룹 관리", "리소스 그룹 삭제", "새로 고침", "CSV로 내보내기", "쿼리 열기", "태그 할당", "이동", and "삭제". Below the navigation bar, there are sections for "개요" (Overview), "필수" (Required), and "리소스" (Resources). The "리소스" section lists various resources such as "활동 로그" (Activity Log), "액세스 제어(IAM)", "태그", "리소스 시각화 도우미", "이벤트", "설정", "비용 관리", "모니터링", "자동화", and "도움말". There are also filter options for "유형" (Type) and "위치" (Location).

Container App 환경 만들기

Container Apps 및 작업을 둘러싼 보안 경계 → 관리의 용이성

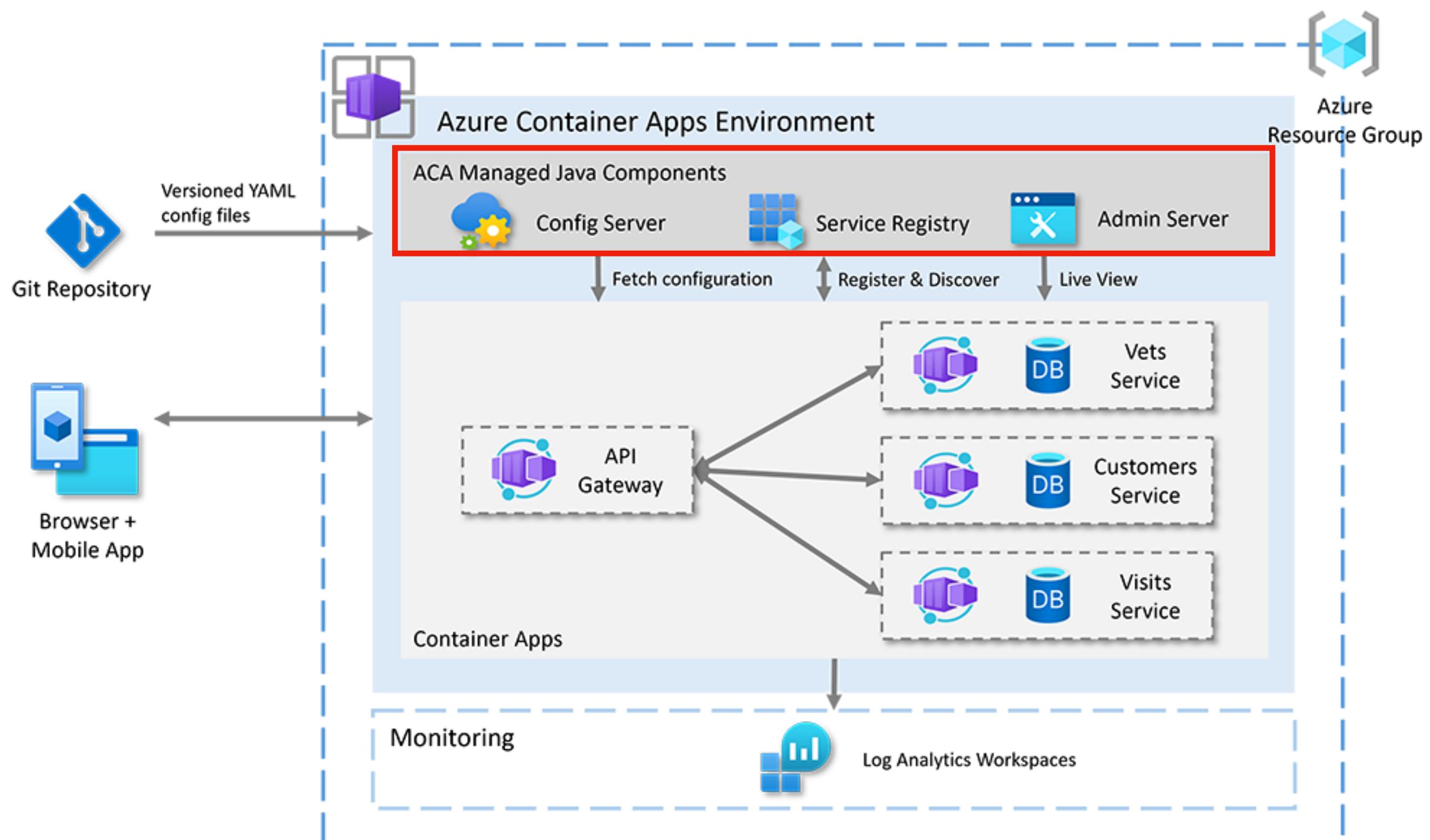
```
az containerapp env create \
--resource-group first-msa \
--name first-msa-container-env \
--location koreacentral
```

The screenshot shows the Azure portal interface for the 'first-msa' resource group. The top navigation bar includes 'Home', 'first-msa' (selected), 'Azure Marketplace', 'Star', '...', and 'Help & support'. Below the navigation is a search bar and a toolbar with '만들기' (Create), '보기 관리' (View Management), '리소스 그룹 삭제' (Delete Resource Group), '새로 고침' (Refresh), 'CSV로 내보내기' (Export to CSV), '쿼리 열기' (Open Query), '태그 할당' (Assign Tags), '이동' (Move), and '삭제' (Delete). On the left, a sidebar lists '개요' (Overview) as the selected tab, along with '활동 로그' (Activity Log), '액세스 제어(IAM)', '태그', '리소스 시각화 도우미', '이벤트', '설정', '비용 관리', and '모니터링'. The main content area displays the '리소스' (Resources) section, which lists two resources: 'first-msa-container-env' (Container Apps 환경, Korea Central) and 'workspace-firstmsaqPSj' (Log Analytics 작업 영역, Korea Central). Each resource has a checkbox, a name, a type, and a location.

리소스	유형	위치
first-msa-container-env	Container Apps 환경	Korea Central
workspace-firstmsaqPSj	Log Analytics 작업 영역	Korea Central

서버 구성 요소 추가

App을 지원하는 Java 구성 요소를 추가!



Config ; 마이크로서비스 앱의 설정을 관리하는 서버

Eureka ; 서비스들이 어디서 실행 중인지 등록하고,
서로 검색 가능하게 해주는 서버(서비스 레지스트리).

Admin ; 각 서비스의 동작 상태나 성능 수치(메트릭)를
모니터링하고 관리하는 서버.

서버 구성 요소 추가

Spring Config Server 추가하기

The screenshot shows the Azure Container Apps service configuration interface for the 'first-msa-container-env' service. On the left, the service navigation bar includes options like '설정', '앱', '서비스', '모니터링', '자동화', and '도움말'. The '서비스' tab is selected. In the center, there's a search bar and a toolbar with '새로 고침', '+ 구성', '삭제', and '연결 끊기'. A tooltip for the '+ 구성' button indicates it adds a Java Configuration Source. Below the toolbar, a message about the public preview add-ons feature being retired on 2025.9.29 is displayed. On the right, the 'Java 구성 요소 구성' (Configure Java Configuration Source) dialog is open. It has two tabs: '구성' (Configure) and '검토' (Review). The '구성' tab is active, showing a dropdown for 'Java 구성 요소 형식' set to '봄용 Config Server', a field for 'Java 구성 요소 이름' containing 'configserver', and a 'Git 리포지토리' section with a URI field containing 'https://github.com/spring-petcli...'. The 'Review' tab is partially visible on the right. At the bottom of the dialog are '닫기' (Close) and '다음' (Next) buttons.

서버 구성 요소 추가

Spring Config Server 추가하기

Git 리포지토리 추가

유형 *

기본값

URI *

https://github.com/spring-petclinic/spring-pet...

패턴

{application} 또는 {application}/{profile}

분기 이름

검색 경로

쉼표로 구분된 검색 경로 입력

인증

일반

Java 구성 요소 구성

✓ 구성 ② 검토

기본

Java 구성 요소 형식

SpringCloudConfig

Java 구성 요소 이름

configserver

Git 리포지토리

기본값

유형

기본값

URI

https://github.com/spring-petclinic/spring-petclinic-microservices-config.git

인증

일반

<https://github.com/spring-petclinic/spring-petclinic-microservices-config>

서버 구성 요소 추가

Spring Config Server 추가하기



```
az containerapp env java-component config-server-for-spring create \
--resource-group first-msa \
--name configserver \
--environment first-msa-container-env \
--configuration spring.cloud.config.server.git.uri=https://github.com/spring-
petclinic/spring-petclinic-microservices-config.git
```

CLI를 통한 설정도 가능!

Config 파일

<https://github.com/spring-petclinic/spring-petclinic-microservices-config>

서버 구성 요소 추가

Eureka Server, Admin Server 구성 요소 추가하기

Java 구성 요소 구성

① 구성 ② 검토

Java 구성 요소 형식 *

유레카 서버 for Spring

[Spring용 Eureka Server에 대한 자세한 정보 보기](#)

Java 구성 요소 이름 *

eureka

바인딩

이 구성 요소는 여러 Container Apps에 바인딩할 수 있습니다. 앱이 이 구성 요소 유형에 이미 바인딩되어 있으면 선택할 수 없습니다.

앱 이름

삭제

환경에서 앱을 찾을 수 없음



Java 구성 요소 구성

① 구성 ② 검토

Java 구성 요소 형식 *

Spring 관리자

[Spring 관리자에 대한 자세한 정보 보기](#)

Java 구성 요소 이름 *

admin

바인딩

이 구성 요소는 여러 Container Apps에 바인딩할 수 있습니다. 앱이 이 구성 요소 유형에 이미 바인딩되어 있으면 선택할 수 없습니다.

앱 이름

삭제

환경에서 앱을 찾을 수 없음



서버 구성 요소 추가

Eureka Server 구성 요소 추가



```
az containerapp env java-component eureka-server-for-spring create \
--resource-group first-msa \
--name eureka \
--environment first-msa-container-env \
```

Admin Server 구성 요소 추가



```
az containerapp env java-component admin-for-spring create \
--resource-group first-msa \
--name admin \
--environment first-msa-container-env \
```

서버 구성 요소 추가

Java 구성 요소 확인

홈 > Container Apps 환경 > first-msa-container-env

first-msa-container-env | 서비스

Container Apps 환경

검색 새고침 구성 삭제 연결 끊기

개요
업데이트를 반영하는 데 최대 1분이 걸릴 수 있습니다. 페이지의 데이터가 오래된 경우 잠시 후에 새로 고치세요.

활동 로그
액세스 제어(IAM)
태그
문제 진단 및 해결

리소스 시각화 도우미

설정 앱 서비스 서비스 모니터링 자동화 도움말

검색 필터 추가

이름 ↑ ↓	유형 ↑ ↓	그룹 ↑ ↓	연결된 앱 ↑ ↓	프로비전 상태 ↑ ↓
<input type="checkbox"/> configserver	봄용 Config Server	Java 구성 요소	0	Succeeded
<input type="checkbox"/> eureka	유레카 서버 for Spring	Java 구성 요소	0	Succeeded
<input type="checkbox"/> admin	Spring 관리자	Java 구성 요소	0	Succeeded

マイクロ 서비스 앱 배포

고객, 수의사, 방문 앱 만들기

홈 > Container Apps >

컨테이너 앱 만들기 ...

기본 컨테이너 수신 태그 검토 + 만들기

프로젝트 세부 정보

구독

Visual Studio Enterprise 구독

리소스 그룹

first-msa

이름

customers-service

Container Apps 환경

지역

koreacentral

Container Apps 환경

first-msa-container-env

컨테이너

이름

customers-service

이미지 원본

일반

레지스트리 로그인 서버

ghcr.io

이미지 및 태그

azure-samples/javaaccelerator/spring-petclinic-customers-service

	이름	도커 컨테이너
고객 앱	customers-service	ghcr.io.azure-samples/javaaccelerator/spring-petclinic-customers-service
수의사 앱	vets-service	ghcr.io.azure-samples/javaaccelerator/spring-petclinic-vets-service
방문 앱	visits-service	ghcr.io.azure-samples/javaaccelerator/spring-petclinic-visits-service

프로젝트 | <https://github.com/spring-petclinic/spring-petclinic-microservices>

이미지 | <https://github.com/orgs/Azure-Samples/packages?tab=packages&q=spring-petclinic>

マイクロ 서비스 앱 배포

고객, 수의사, 방문 앱 만들기



```
az containerapp create \
--resource-group first-msa \
--name customers-service \
--environment first-msa-container-env \
--image ghcr.io/azure-samples/javaaccelerator/spring-petclinic-customers-service
```

	이름	도커 컨테이너
고객 앱	customers-service	ghcr.io/azure-samples/javaaccelerator/spring-petclinic-customers-service

마이크로 서비스 앱 배포

API Gateway 앱 만들기

The screenshot shows the Azure Container Apps interface for managing an API gateway application named 'api-gateway'. The left sidebar lists various settings like '개요', '활동 로그', '액세스 제어(IAM)', '태그', '문제 진단 및 해결', '리소스 시각화 도우미', '애플리케이션', '설정', 'Networking', and '도움말'. The 'Networking' section is expanded, and the '수신' (Receive) tab is selected. Under '수신', there's a note about enabling receiving traffic to the latest version. It shows the '수신 유형' (Receive Type) set to '어디서나 트래픽 허용' (Allow traffic from anywhere), with options for 'HTTP', 'TCP', '무시' (Ignore), '수락' (Accept), and '필요' (Required). The '대상 포트' (Target Port) is set to '8080'. Below this, the URL is displayed as <https://api-gateway.livelystone-a1f2dbe4.koreacentral.azurecontainerapps.io>. The 'IP 제한' (IP Restriction) section indicates that traffic is accepted from all IP addresses. The bottom of the screen shows the 'IP 보안 제한 모드' (IP Security Limit Mode) set to '모든 트래픽 허용(기본값)' (Allow all traffic (Default)).

- 8080 포트의 외부 접속 허용

マイクロ 서비스 앱 배포

API Gateway 앱 만들기



```
az containerapp create \
--resource-group first-msa \
--name api-gateway \
--environment first-msa-container-env \
--image ghcr.io/azure-samples/javaaccelerator/spring-petclinic-api-gateway
--ingress external \
--target-port 8080 \
--query properties.configuration.ingress.fqdn
```

CLI를 통한 제작도 가능!

컨테이너 앱 바인딩

- 시작 시 Config 서버에 각 앱의 구성 데이터(설정 정보) 삽입
- 서비스 검색을 위해 Eureka 서버에 앱 등록
- Admin 서버에서 앱을 모니터링할 수 있도록 설정

The screenshot shows the Azure Container Apps interface for managing a service named 'first-msa-container-env'. On the left, under the 'Services' tab, there is a list of bindings: 'configserver' (Config Server), 'eureka' (Eureka Server for Spring, selected), and 'admin' (Spring Manager). On the right, a modal window titled 'Java 구성 요소 구성' (Configure Java components) is open, showing a list of services that can be bound: 'vets-service', 'customers-service', 'visits-service', and 'api-gateway'. Each service has a dropdown menu and a delete icon.

Home > Container Apps > first-msa-container-env

first-msa-container-env | 서비스

Container Apps 환경

검색 새고침 구성 삭제 연결 끊기

업데이트를 반영하는 데 최대 1분이 걸릴 수 있습니다. 페이지의 데이터가 오래된 경우 잠시 후에 새로 고치시.

The public preview add-ons feature will be retired on 2025. 9. 29.. [자세한 정보 보기](#)

검색 필터 추가

이름 ↑ 유형 ↑

이름	유형
configserver	봄용 Config Server
eureka	유레카 서버 for Spring
admin	Spring 관리자

바인딩

이 구성 요소는 여러 Container Apps에 바인딩할 수 있습니다. 앱이 이 구성 요소 유형에 이미 바인딩되어 있으면 선택할 수 없습니다.

앱 이름 삭제

앱 이름	삭제
vets-service	삭제
customers-service	삭제
visits-service	삭제
api-gateway	삭제

선택

유지

컨테이너 앱 바인딩

- 시작 시 Config 서버에 각 앱의 구성 데이터(설정 정보) 삽입
- 서비스 검색을 위해 Eureka 서버에 앱 등록
- Admin 서버에서 앱을 모니터링할 수 있도록 설정

이름 ↑ ↓	유형 ↑ ↓	그룹 ↑ ↓	연결된 앱 ↑ ↓	프로비전 상태 ↑ ↓
<input type="checkbox"/> configserver	봄용 Config Server	Java 구성 요소	4	Succeeded
<input type="checkbox"/> eureka	유레카 서버 for Spring	Java 구성 요소	4	Succeeded
<input type="checkbox"/> admin	Spring 관리자	Java 구성 요소	4	Succeeded

visits/customers/vets service, api-gateway 총 4개의 앱 등록

컨테이너 앱 바인딩

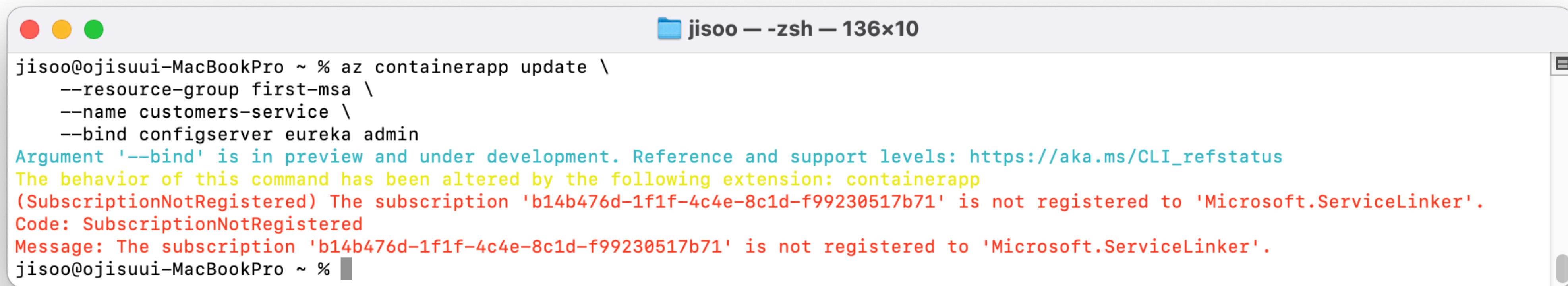
customers-service, vets-service, visits-service, api-gateway에 대해
다음과 같은 명령어를 실행하여 앱을 바인딩 할 수 있음



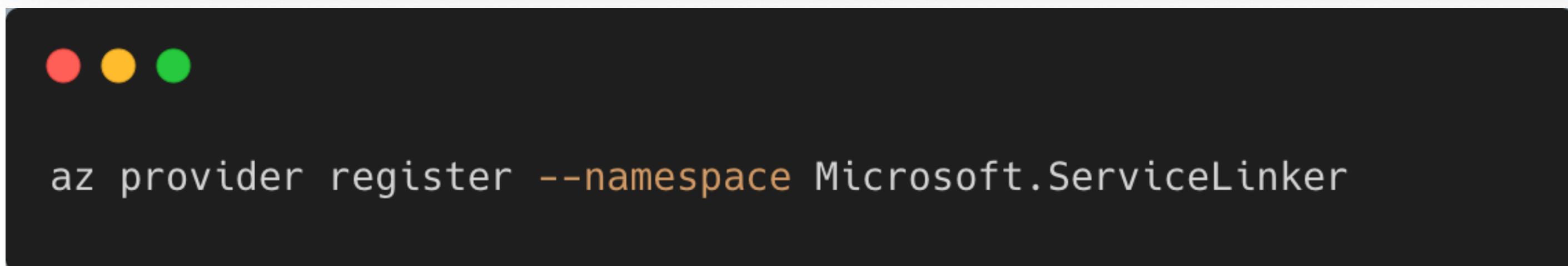
```
az containerapp update \
--resource-group first-msa \
--name customers-service \
--bind configserver eureka admin
```

바인딩 에러

CLI의 --bind 명령어 실행시 다음과 같은 에러가 난다면,
ServiceLinker 권한을 부여해주면 됨



```
jisoo@ojisuui-MacBookPro ~ % az containerapp update \
  --resource-group first-msa \
  --name customers-service \
  --bind configserver eureka admin
Argument '--bind' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
The behavior of this command has been altered by the following extension: containerapp
(SubscriptionNotRegistered) The subscription 'b14b476d-1f1f-4c4e-8c1d-f99230517b71' is not registered to 'Microsoft.ServiceLinker'.
Code: SubscriptionNotRegistered
Message: The subscription 'b14b476d-1f1f-4c4e-8c1d-f99230517b71' is not registered to 'Microsoft.ServiceLinker'.
jisoo@ojisuui-MacBookPro ~ %
```



```
az provider register --namespace Microsoft.ServiceLinker
```

프론트엔드 앱 확인

API Gateway URL을 통해 앱 확인 가능

홈 > Container Apps >

api-gateway

컨테이너 앱

검색

□ 종지 ⌂ 새로 고침 🗑 삭제 📩 피드백을 보내 주세요

개요

활동 로그

액세스 제어(IAM)

태그

문제 진단 및 해결

리소스 시작화 도우미

> 애플리케이션

> 설정

> Networking

> Security

> 모니터링

필수

리소스 그룹 ([이동](#)) : [first-msa](#)

상태 : 실행하는 중

위치 ([이동](#)) : Korea Central

구독 ([이동](#)) : [Visual Studio Enterprise 구독](#)

구독 ID : b14b476d-1f1f-4c4e-8c1d-f99230517b71

애플리케이션 URL : <https://api-gateway.victoriousmeadow-33d67441.koreace...> 📁

Container Apps 환경 : [first-msa-container-env](#)

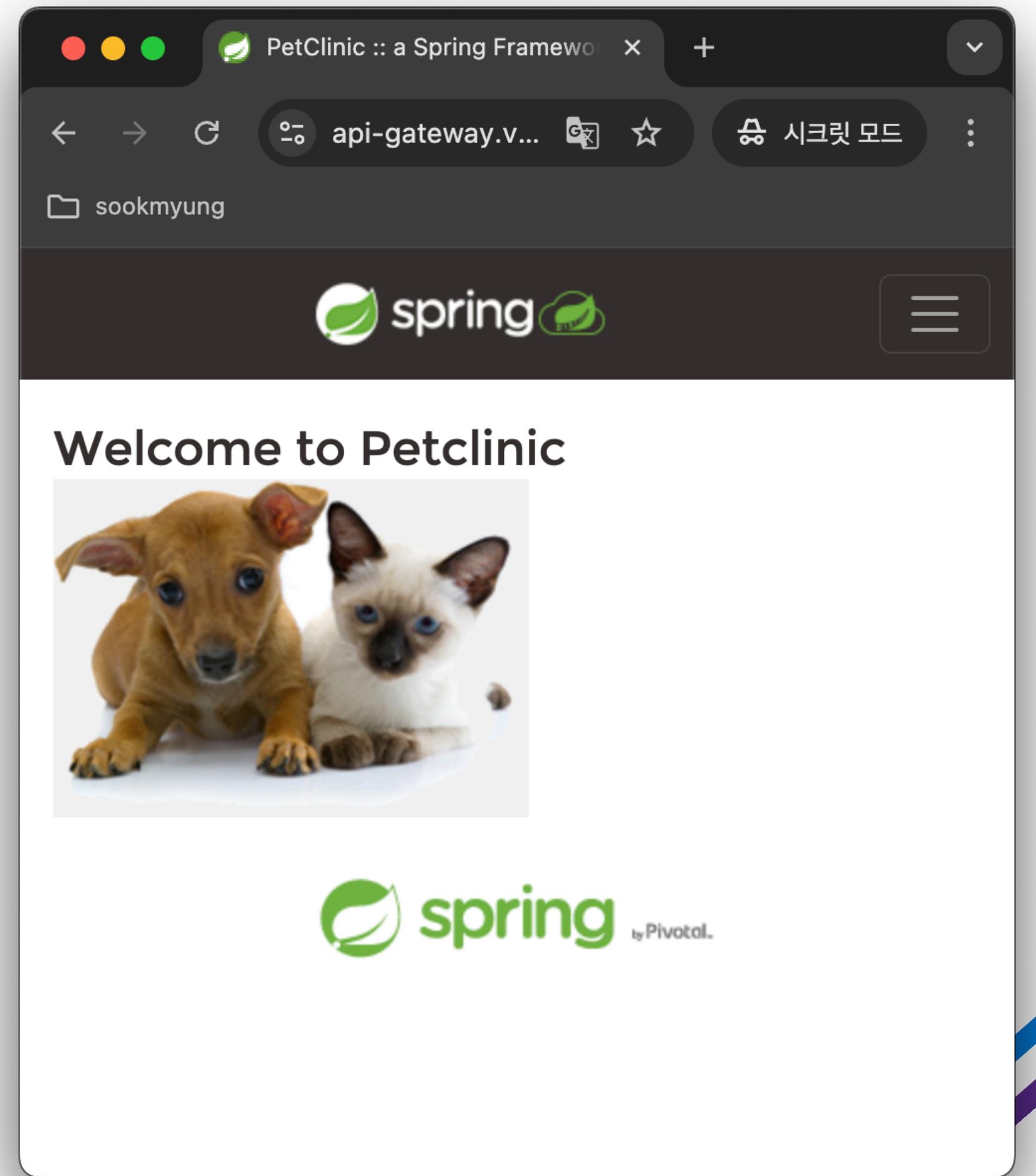
환경 유형 : 워크로드 프로필

Log Analytics : [workspace-firstmsaqPSj](#)

개발 단계 : 제네릭 (관리)

.NET 구성 대시보드 : Not yet active ([set up](#))

JSON 보기



프론트엔드 앱 확인

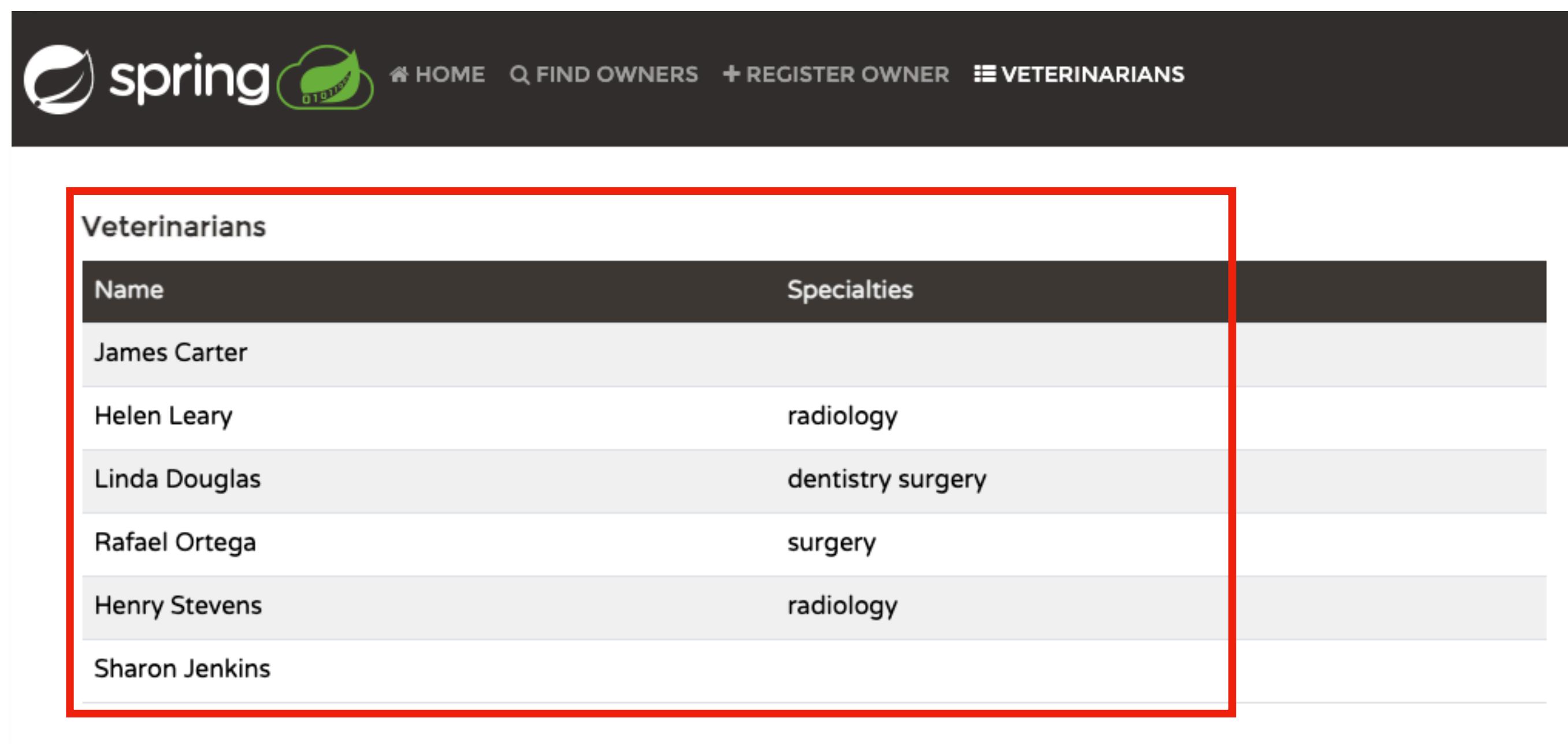
바인딩 확인

```
/*
 * @author Juergen Hoeller
 * @author Mark Fisher
 * @author Ken Krebs
 * @author Arjen Poutsma
 * @author Maciej Szarlinski
 */
@RequestMapping("/vets")
@RestController
class VetResource {

    private final VetRepository vetRepository;

    VetResource(VetRepository vetRepository) {
        this.vetRepository = vetRepository;
    }

    @GetMapping
    @Cacheable("vets")
    public List<Vet> showResourcesVetList() {
        return vetRepository.findAll();
    }
}
```



Name	Specialties
James Carter	
Helen Leary	radiology
Linda Douglas	dentistry surgery
Rafael Ortega	surgery
Henry Stevens	radiology
Sharon Jenkins	

Vets 앱 > 모든 수의사 찾기 가능

프론트엔드 앱 확인

바인딩 확인

```
@RestController  
@Timed("petclinic.visit")  
class VisitResource {  
  
    private static final Logger log = LoggerFactory.getLogger(VisitResource.class);  
  
    private final VisitRepository visitRepository;  
  
    VisitResource(VisitRepository visitRepository) {  
        this.visitRepository = visitRepository;  
    }  
  
    @PostMapping("owners/*/pets/{petId}/visits")  
    @ResponseStatus(HttpStatus.CREATED)  
    public Visit create(  
        @Valid @RequestBody Visit visit,  
        @PathVariable("petId") @Min(1) int petId) {  
  
        visit.setPetId(petId);  
        log.info("Saving visit {}", visit);  
        return visitRepository.save(visit);  
    }  
  
    @GetMapping("owners/*/pets/{petId}/visits")  
    public List<Visit> read(@PathVariable("petId") @Min(1) int petId) {  
        return visitRepository.findById(petId);  
    }  
}
```

The screenshot shows a web application interface for managing pet visits. The top section, "Owner Information", displays details for George Franklin: Name (George Franklin), Address (110 W. Liberty St.), City (Madison), and Telephone (6085551023). Below this are two buttons: "Edit Owner" and "Add New Pet". The bottom section, "Pets and Visits", lists a pet named Leo with birth date 2010 Sep 07 and type cat. To the right of Leo's information is a table for visits:

Visit Date	Description
2025 Aug 29	방문 기록 추가 1

Buttons for "Edit Pet" and "Add Visit" are located at the bottom of the visit table.

Visits 앱 > 방문 확인, 추가 가능

프론트엔드 앱 확인

바인딩 확인

```
@PostMapping("/owners/{ownerId}/pets")
@ResponseStatus(HttpStatus.CREATED)
public Pet processCreationForm(
    @RequestBody PetRequest petRequest,
    @PathVariable("ownerId") @Min(1) int ownerId) {

    Owner owner = ownerRepository.findById(ownerId)
        .orElseThrow(() -> new ResourceNotFoundException("Owner " + ownerId + " not found"));

    final Pet pet = new Pet();
    owner.addPet(pet);
    return save(pet, petRequest);
}

@PutMapping("/owners/*/pets/{petId}")
@ResponseStatus(HttpStatus.NO_CONTENT)
public void processUpdateForm(@RequestBody PetRequest petRequest) {
    int petId = petRequest.id();
    Pet pet = findPetById(petId);
    save(pet, petRequest);
}
```

The screenshot shows the Spring PetClinic application interface. At the top, there's a navigation bar with the Spring logo. Below it, the 'Owner Information' section displays details for George Franklin: Name (George Franklin), Address (110 W. Liberty St.), City (Madison), and Telephone (6085551023). Two buttons are present: 'Edit Owner' and 'Add New Pet'. The 'Add New Pet' button is highlighted with a red box. In the 'Pets and Visits' section, a pet named 'Leo2' is listed with a birth date of 2010 Sep 07 and a type of 'cat'. An 'Edit Pet' button is also highlighted with a red box. To the right, a modal dialog is open for adding a new pet. It has fields for 'Pet', 'Owner' (set to George Franklin), 'Name' (Leo2), 'Birth date' (2010.09.07), and 'Type' (cat). A 'Submit' button is at the bottom of the modal.

Customers 앱 > 반려동물 조회/추가/수정

프로젝트 | <https://github.com/spring-petclinic/spring-petclinic-microservices/blob/main/spring-petclinic-customers-service/src/main/java/org/springframework/samples/petclinic/customers/web/PetResource.java>

프론트엔드 앱 확인

바인딩 확인

```
/*
 * Create Owner
 */
@PostMapping
@ResponseStatus(HttpStatus.CREATED)
public Owner createOwner(@Valid @RequestBody OwnerRequest ownerRequest) {
    Owner owner = ownerEntityMapper.map(new Owner(), ownerRequest);
    return ownerRepository.save(owner);
}

/**
 * Read single Owner
 */
@GetMapping(value = "/{ownerId}")
public Optional<Owner> findOwner(@PathVariable("ownerId") @Min(1) int ownerId) {
    return ownerRepository.findById(ownerId);
}

/**
 * Read List of Owners
 */
@GetMapping
public List<Owner> findAll() {
    return ownerRepository.findAll();
}
```

The screenshot shows two views of the Spring Petclinic application. The top view is a 'REGISTER OWNER' form with fields for First name, Last name, Address, City, and Telephone, each with a placeholder value. A 'Submit' button is at the bottom. The bottom view is a 'OWNERS' list page with a search bar and a table of owner records:

Name	Address	City	Telephone	Pets
George Franklin	110 W. Liberty St.	Madison	6085551023	Leo2
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085551749	Basil
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George

Customers 앱 > 주인 조회/추가/수정

프로젝트 | <https://github.com/spring-petclinic/spring-petclinic-microservices/blob/main/spring-petclinic-customers-service/src/main/java/org/springframework/samples/petclinic/customers/web/OwnerResource.java>

Eureka 대시보드 확인

Container Apps 환경에 IAM 역할 추가

역할 구성원 조건 검토 + 할당

역할 정의는 권한 컬렉션입니다. 기본 제공 역할을 사용하거나 사용자 지정 역할을 만들 수 있습니다. [자세한 정보](#)

작업 기능 역할 권한 있는 관리자 역할

다른 사용자에게 역할을 할당하는 기능과 같은 권한 있는 관리자 액세스를 부여합니다.

⚠️ 액세스 권한이 적은 작업 기능 역할을 대신 사용할 수 있나요?

역할 이름, 설명, 사용 권한 또는 ID로 검색 형식 : 모두 범주 : 모두

이름 ↑↓ 설명 ↑↓

소유자	Azure RBAC에서 역할을 할당하는 기능을 포함하여 모든 리소스를 관리할 수 있는 모든 권한을 부여합니다.
-----	---

소유자 또는 기여자 역할 추가하기

역할 구성원 조건 검토 + 할당

선택한 역할 소유자

다음에 대한 액세스 할당: 사용자, 그룹 또는 서비스 주체 관리 ID

구성원 + 구성원 선택

이름	개체 ID	유형
Jisoo Oh	..	사용자

Eureka 대시보드 확인

'소유자' 역할 확인 가능

홈 > first-msa-container-env

first-msa-container-env | 액세스 제어(IAM)

Container Apps 환경

검색

+ 추가 ▼ ⏪ 역할 할당 다운로드 ⏮ 열 편집

개요

활동 로그

액세스 제어(IAM) 선택됨

태그

문제 진단 및 해결

리소스 시각화 도우미

> 설정

> 앱

액세스 권한 확인

역할 할당

역할

거부 할당

내 액세스

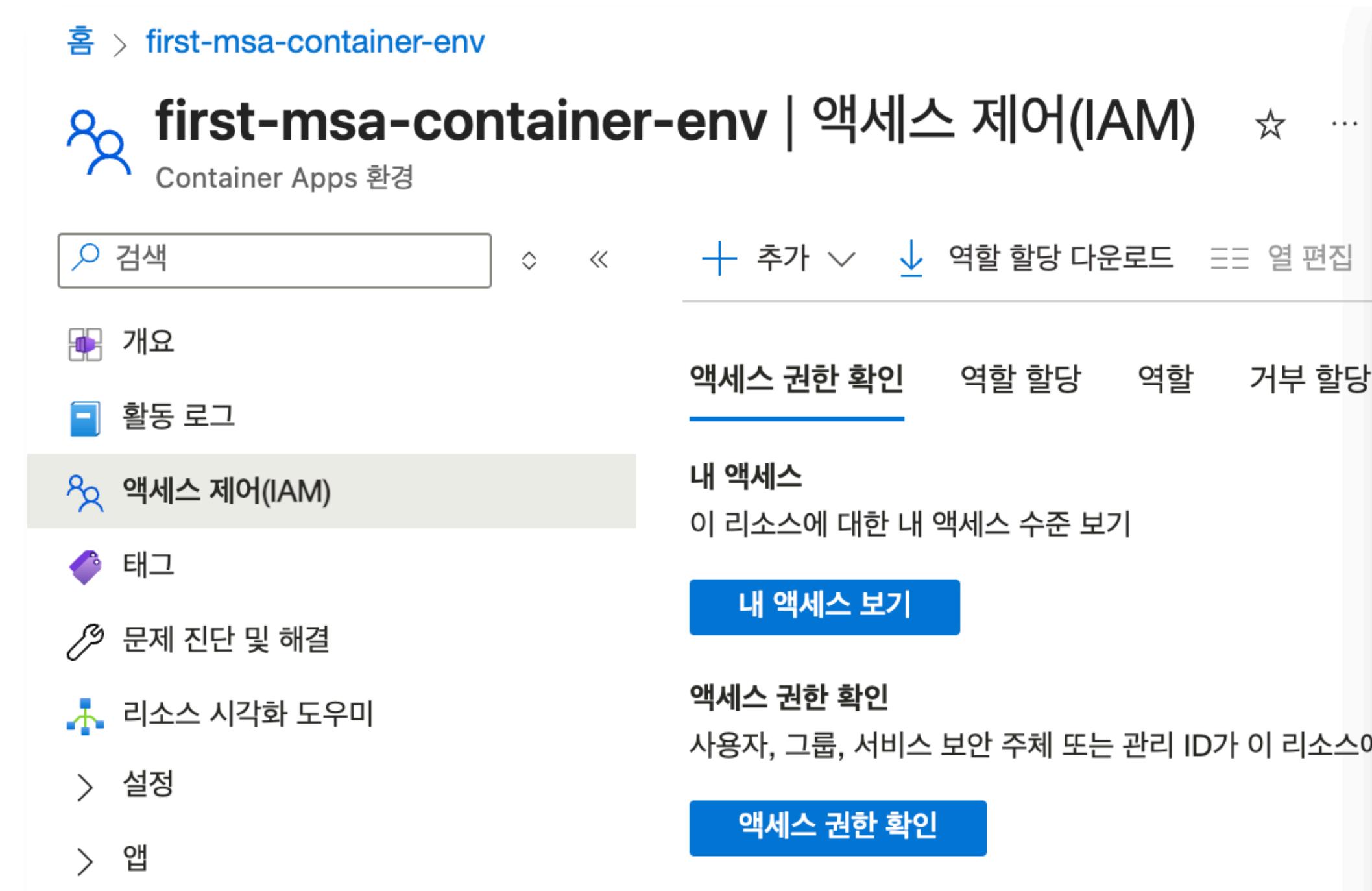
이 리소스에 대한 내 액세스 수준 보기

내 액세스 보기

액세스 권한 확인

사용자, 그룹, 서비스 보안 주체 또는 관리 ID가 이 리소스에

액세스 권한 확인



Jisoo Oh 할당 - first-msa-container-env

현재 역할 할당 적격 할당

이 범위에 상속되거나 이 범위에 있는 선택한 사용자, 그룹, 서비스 주체 또는 관리되는 ID에 대한 할당입니다.

할당 이름이나 설명으로 검색

역할 할당(1) ⓘ

역할	설명	범위	그룹 할당	조건
소유자	Azure RBAC에서 역할을 할당하는 ... 이 리소스	--	--	보기/편집

거부 할당(0) ⓘ

클래스 관리자(1) ⓘ

역할	설명
클래스 관리자	설명

Eureka 대시보드 확인

The screenshot shows the Eureka dashboard interface. At the top, there's a header bar with the Eureka logo, a search bar, and navigation links. Below the header, the main content area is divided into sections: 'System Status' and 'Instances currently registered with Eureka'.

System Status

Environment	test	Current time	2025-08-18T09:28:23 +0000
Data center	default	Uptime	01:27
		Lease expiration enabled	true
		Renews threshold	0
		Renews (last min)	12

DS Replicas

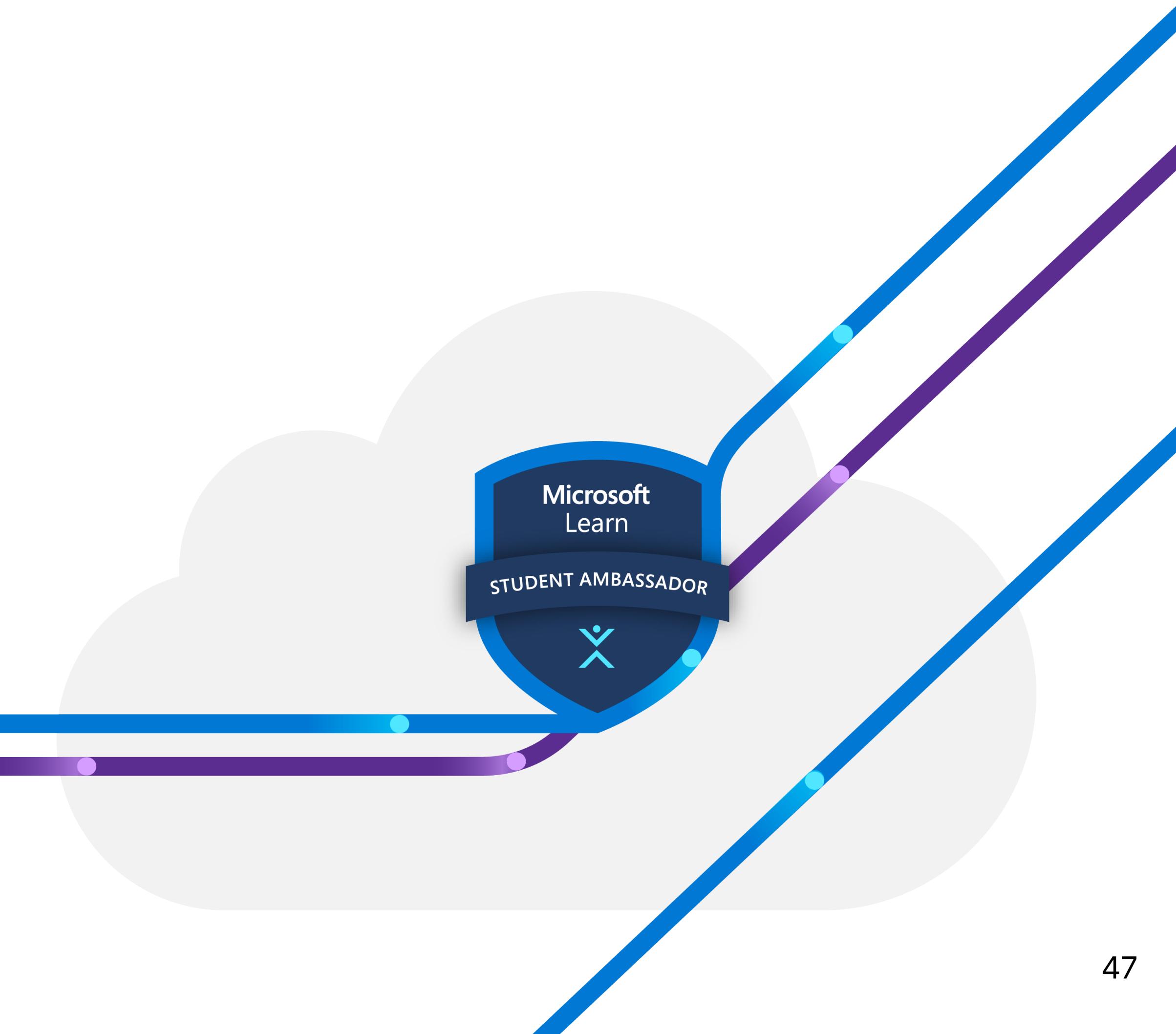
localhost

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
CUSTOMERS-SERVICE	n/a (1)	(1)	UP (1) - customers-service:6c27d17c-5414-4372-8999-dfbab693ad10
VETS-SERVICE	n/a (1)	(1)	UP (1) - vets-service--0000004-84fdb4965d-2zmpt:vets-service:0
VISITS-SERVICE	n/a (1)	(1)	UP (1) - visits-service:76e9f884-897c-4f33-b677-da05b0434735

03.

마무리



마무리

이번 세션을 통해 알아본 것들은?



Container, MSA에 대한 간단한 개념



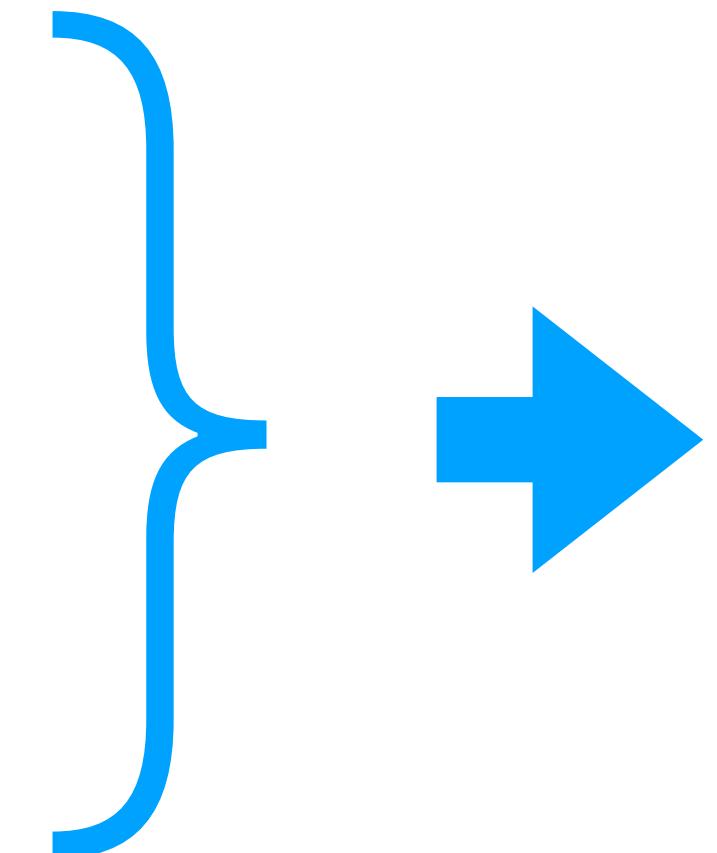
Config, Eureka, Admin 구성 요소 제작



api-gateway, customers/vets/visits 서비스 배포

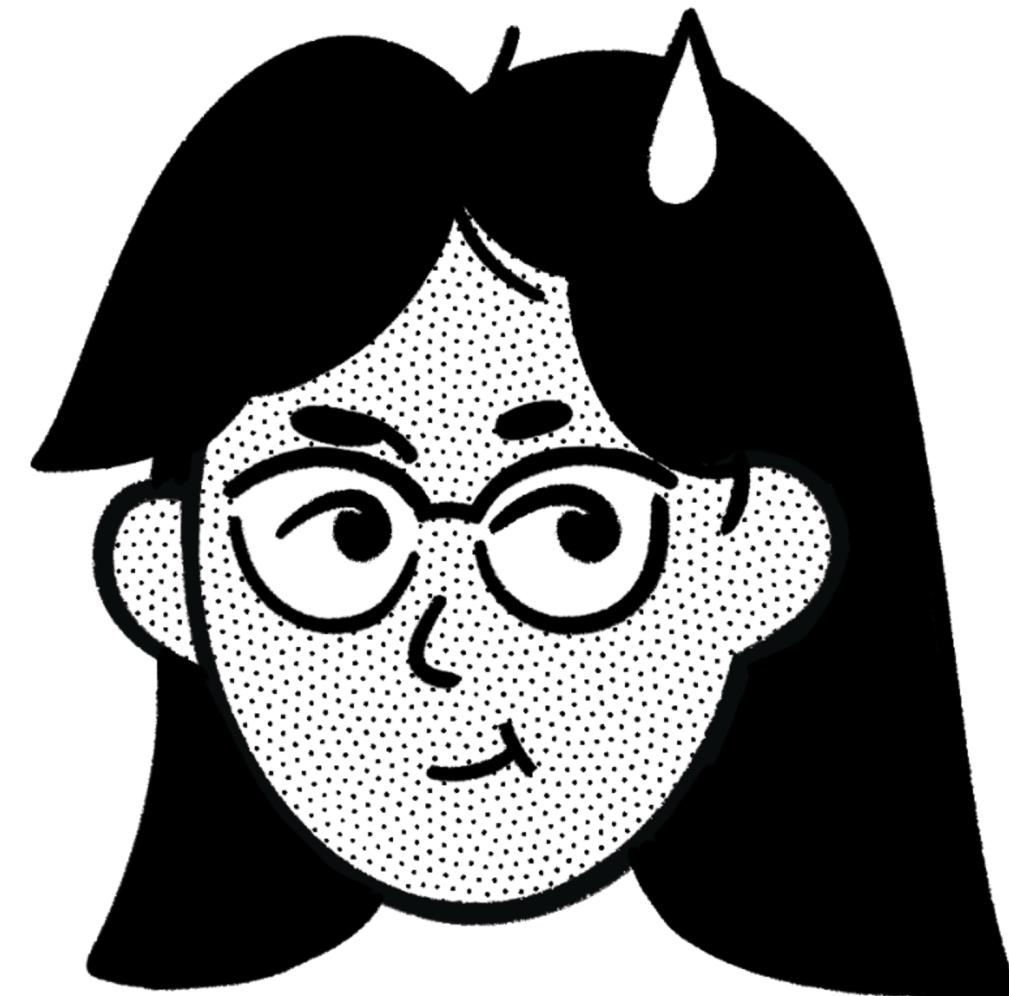


각각의 구성요소에 서비스 바인딩

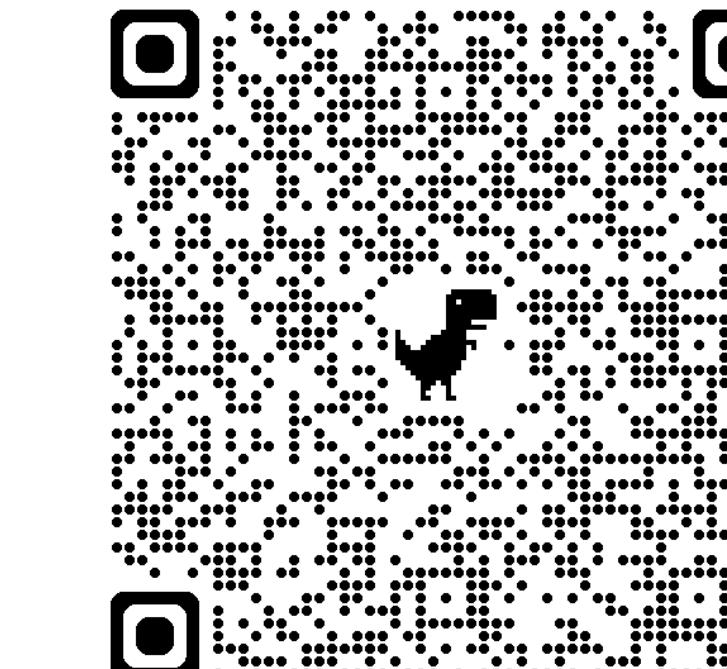
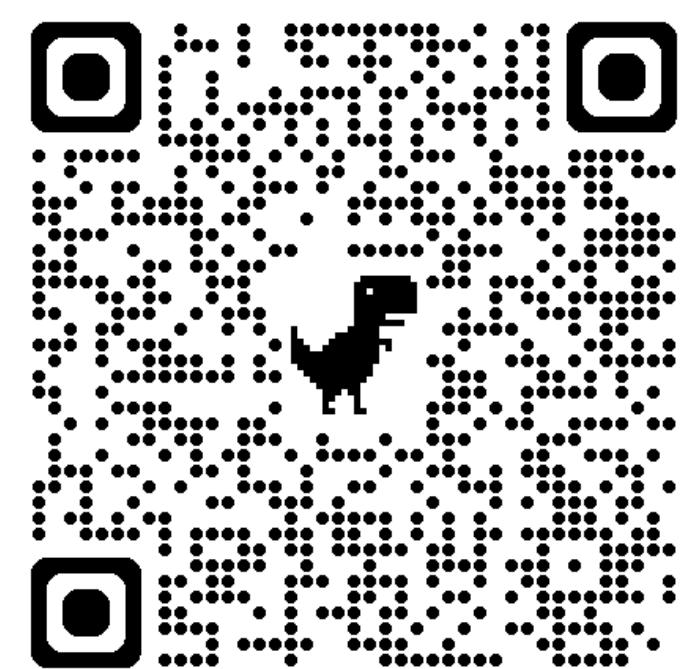


Azure Container Apps 찍어먹기~

Q&A



이메일 | lucy27@sookmyung.ac.kr



MS Learn

LinkedIn

감사합니다!

MLSA Beta 오지수

250831 IT 러닝 캠프

