

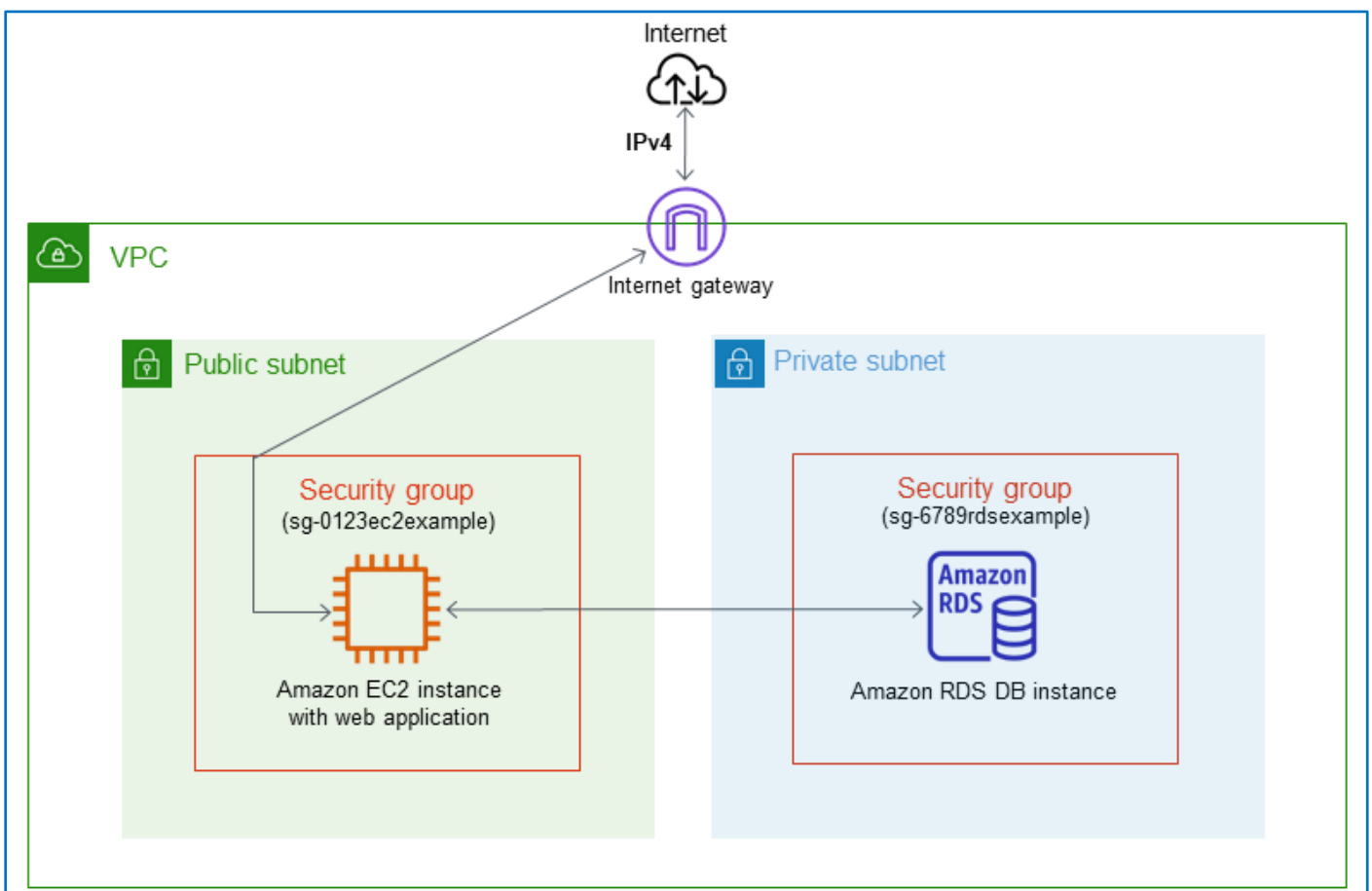
# Lab1. Using Amazon MySQL RDS

## 목적

이번 실습에서는 PHP가 있는 Apache 웹 서버를 설치하고 MySQL 데이터베이스를 생성한다. 이 웹 서버는 Amazon Linux를 사용하여 Amazon EC2 인스턴스에서 실행되며, MySQL 데이터베이스는 MySQL DB 인스턴스이다. Amazon EC2 인스턴스와 DB 인스턴스 모두 Amazon VPC 서비스를 기반으로 하는 Virtual Private Cloud(VPC)에서 실행할 것이다. 또한 Windows Server 2019 기반으로 MySQL Workbench 도구를 사용하여 설치된 MySQL 인스턴스에 접속하여 데이터베이스를 생성하고 테이블을 생성하며 필요한 데이터를 입력하는 것을 다룰 것이다.

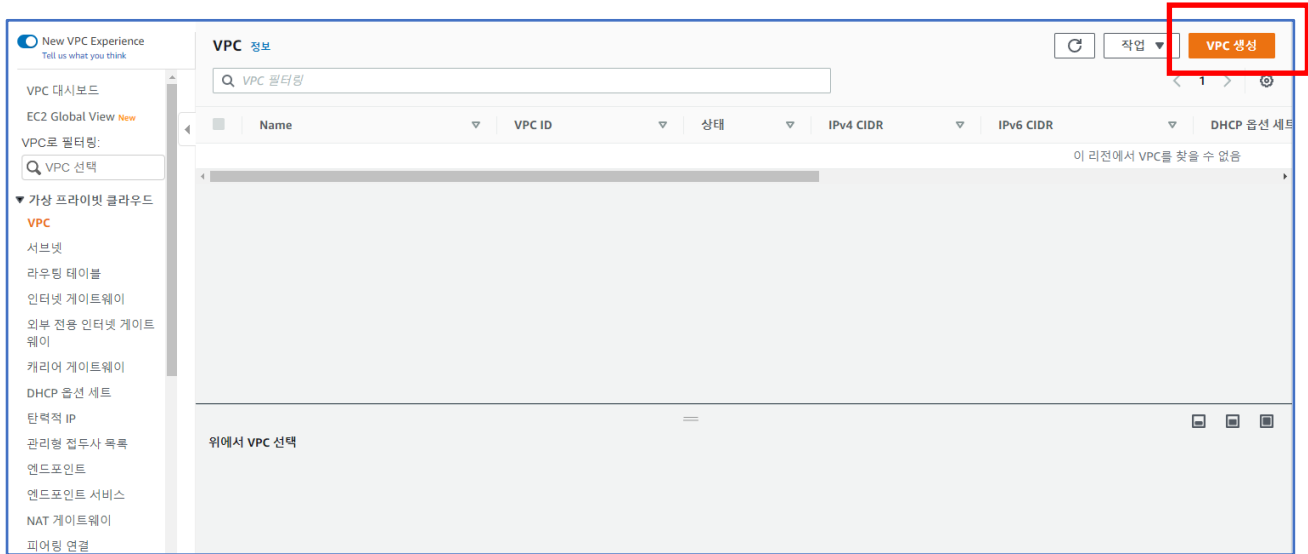
## 사전 준비물

AWS Free-Tier 계정



# Private Subnet과 Public Subnet을 포함하는 VPC 생성

1. [서비스] > [네트워킹 및 콘텐츠 전송] > [VPC]를 클릭하여 VPC 페이지로 이동한다. 우측 상단의 [VPC 생성]을 클릭한다.



2. [VPC 생성] 페이지에 들어왔다. 다음의 각 값을 설정한다. 나머지는 기본값을 사용한다.

- A. [생성할 리소스] : VPC만
- B. [이름 태그] : lab-vpc
- C. [IPv4 CIDR] : 10.0.0.0/16

VPC > VPC > VPC 생성

## VPC 생성 정보

VPC는 AWS 클라우드의 격리된 부분으로서, Amazon EC2 인스턴스와 같은 AWS 객체로 채워집니다.

### VPC 설정

**생성할 리소스 정보**  
VPC 리소스만 생성 또는 VPC, 서브넷 등을 생성

☒ VPC만 ☐ VPC, 서브넷 등

**이름 태그 - 선택 사항**  
'Name' 키와 사용자가 지정한 값을 포함하는 태그를 생성합니다.

lab-vpc

**IPv4 CIDR 블록 정보**

☒ IPv4 CIDR 수동 입력 ☐ IPAM 할당 IPv4 CIDR 블록

**IPv4 CIDR**

10.0.0.0/16

**IPv6 CIDR 블록 정보**

☒ IPv6 CIDR 블록 없음 ☐ IPAM 할당 IPv6 CIDR 블록 ☐ Amazon 제공 IPv6 CIDR 블록 ☐ 내가 소유한 IPv6 CIDR

**테넌시 정보**

기본값

3. 페이지를 아래로 스크롤다운하여 [태그] 섹션에서 [키]는 Name으로, [값]은 lab-vpc 로 되어 있음을 확인하고 [VPC 생성]을 클릭한다.

### 태그

태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키

값 - 선택 사항

Q Name X

Q lab-vpc X

제거

새 태그 추가

49을(를) 태그개 더 추가할 수 있습니다.

취소

VPC 생성

4. VPC와 연결할 인터넷 게이트웨이를 생성한다. 좌측 메뉴에서 [인터넷 게이트웨이]를 클릭한다. 인터넷 게이트웨이 페이지에서 우측 상단의 [인터넷 게이트웨이 생성]을 클릭한다.

New VPC Experience  
Tell us what you think

VPC 대시보드

EC2 Global View New

VPC로 필터링:  
Q VPC 선택

▼ 가상 프라이빗 클라우드

VPC

서브넷

인터넷 게이트웨이

이전 것을 인터넷 게이트웨이

웨이

캐리어 게이트웨이

DHCP 옵션 세트

탄력적 IP

관리형 접두사 목록

엔드포인트

엔드포인트 서비스

NAT 게이트웨이

피어링 연결

인터넷 게이트웨이 정보

Q 인터넷 게이트웨이 필터링

Name

인터넷 게이트웨이 ID

상태

VPC ID

소유자

이 리전에 인터넷 게이트웨이가 없음

위에서 인터넷 게이트웨이 선택

↺

작업

인터넷 게이트웨이 생성

5. [인터넷 게이트웨이 생성] 페이지에서 [이름 태그]에 lab-ig를 입력하고, [태그] 섹션에서 [키]를 Name으로, [값]을 lab-ig로 입력하고 [인터넷 게이트웨이 생성]을 클릭한다.

VPC > 인터넷 게이트웨이 > 인터넷 게이트웨이 생성

## 인터넷 게이트웨이 생성 정보

인터넷 게이트웨이는 VPC를 인터넷과 연결하는 가상 라우터입니다. 새 인터넷 게이트웨이를 생성하려면 아래에서 게이트웨이 이름을 지정해야 합니다.

### 인터넷 게이트웨이 설정

**이름 태그**  
'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

### 태그 - 선택 사항

태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

**키**

**값 - 선택 사항**

49을(를) 태그.개 더 추가할 수 있습니다.

6. 인터넷 게이트웨이가 성공적으로 생성되었다. 이제 생성된 인터넷 게이트웨이를 VPC에 연결하도록 한다. 우측 상단의 [VPC에 연결] 버튼을 클릭한다.

☒ igw-069f5a2a1966383ff - lab-ig의 인터넷 게이트웨이가 생성되었습니다. 이제 VPC에 연결하여 VPC가 인터넷과 통신하도록 할 수 있습니다.

VPC > 인터넷 게이트웨이 > igw-069f5a2a1966383ff

## igw-069f5a2a1966383ff / lab-ig 작업 ▼

### 세부 정보 정보

인터넷 게이트웨이 ID igw-069f5a2a1966383ff	상태 Detached	VPC ID -	소유자 789534828835
---------------------------------------	----------------	-------------	---------------------

### 태그

Key

Value

Name	lab-ig
------	--------

7. [VPC에 연결] 창에서 [사용 가능한 VPC] 목록에서 위에서 생성한 **lab-vpc**를 선택하고 [인터넷 게이트웨이 연결] 버튼을 클릭한다.

VPC > 인터넷 게이트웨이 > VPC에 연결(igw-069f5a2a1966383ff)

## VPC에 연결(igw-069f5a2a1966383ff) 정보

**VPC**  
인터넷 게이트웨이를 VPC에 연결하여 인터넷과의 통신을 활성화합니다. 아래에서 연결하려는 VPC를 지정하십시오.

사용 가능한 VPC  
인터넷 게이트웨이를 이 VPC에 연결합니다.

▶ AWS Command Line Interface 명령

취소 **인터넷 게이트웨이 연결**

8. 이제 인터넷 게이트웨이의 [상태] 값이 **Attached**로 변경되었음을 확인할 수 있다.

☑ 인터넷 게이트웨이 igw-069f5a2a1966383ff(가) vpc-0d7fa7b1654983496에 연결되었습니다.

VPC > 인터넷 게이트웨이 > igw-069f5a2a1966383ff

## igw-069f5a2a1966383ff / lab-ig

작업 ▼

세부 정보 정보

인터넷 게이트웨이 ID igw-069f5a2a1966383ff	<b>상태</b> 🟢 Attached	VPC ID vpc-0d7fa7b1654983496   lab-vpc	소유자 789534828835
---------------------------------------	-------------------------	---	---------------------

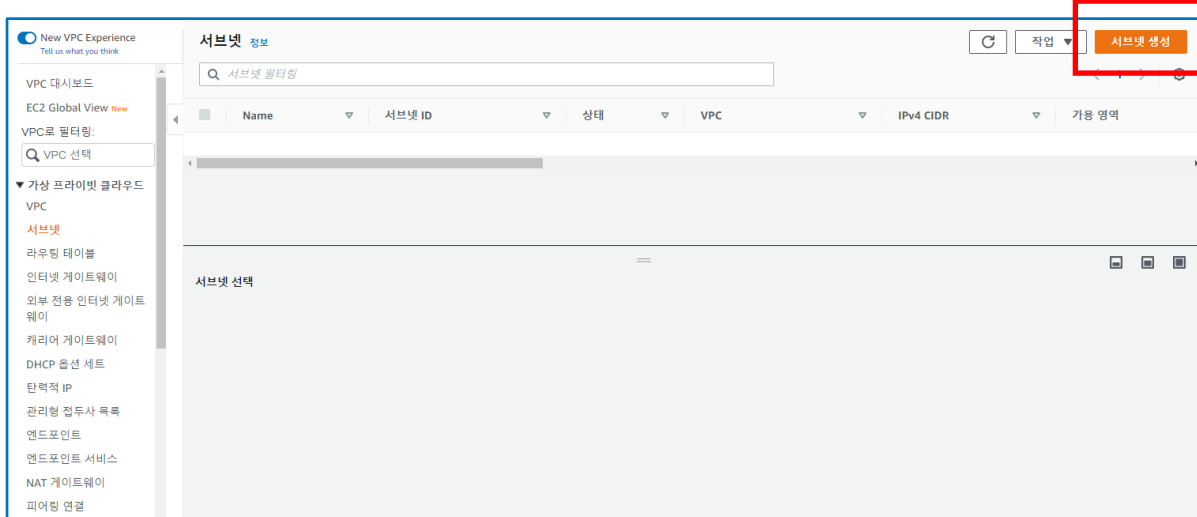
태그

태그 검색

태그 관리

Key	Value
Name	lab-ig

9. **서브넷**을 생성할 차례이다. 좌측 메뉴에서 **[서브넷]** 메뉴를 클릭하여 **서브넷** 페이지로 이동한 후, 우측 상단의 **[서브넷 생성]**을 클릭한다.



10. [서브넷 설정] 페이지이다. 다음의 각 값을 설정한 후, [서브넷 생성] 버튼을 클릭한다.

- A. [서브넷 이름] : lab-public-subnet
- B. [가용 영역] : 아시아 태평양(서울) / ap-northeast-2a
- C. [IPv4 CIDR 블록] : 10.0.10.0/24
- D. [키] : Name
- E. [값] : lab-public-subnet

### 서브넷 설정

서브넷의 CIDR 블록 및 가용 영역을 지정합니다.

#### 1/1개 서브넷

서브넷 이름  
'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

이름은 최대 256자까지 입력할 수 있습니다.

가용 영역 정보  
서브넷이 상주할 영역을 선택합니다. 선택하지 않으면 Amazon이 자동으로 선택합니다.

IPv4 CIDR 블록 정보

▼ 태그 - 선택 사항

키	값 - 선택 사항
<input type="text" value="Name"/>	<input type="text" value="lab-public-subnet"/>

새 태그 추가  
49을(를) 태그 개 더 추가할 수 있습니다.

제거

새 서브넷 추가

취소

서브넷 생성

11. 2번째 서브넷을 생성하는데, 이번에는 프라이빗 서브넷이다. 다음의 각 값을 입력 후, [서브넷 생성]을 클릭한다.

- A. [서브넷 이름] : lab-private1-subnet
- B. [가용 영역] : 아시아 태평양(서울) / ap-northeast-2c
- C. [IPv4 CIDR 블록] : 10.0.20.0/24
- D. [키] : Name
- E. [값] : lab-private1-subnet

### 서브넷 설정

서브넷의 CIDR 블록 및 가용 영역을 지정합니다.

1/1개 서브넷

서브넷 이름

'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

이름은 최대 256자까지 입력할 수 있습니다.

가용 영역 정보

서브넷이 상주할 영역을 선택합니다. 선택하지 않으면 Amazon이 자동으로 선택합니다.

IPv4 CIDR 블록 정보

▼ 태그 - 선택 사항

키	값 - 선택 사항
<input type="text" value="Name"/>	<input type="text" value="lab-private1-subnet"/>

49글(을) 태그.개 더 추가할 수 있습니다.



12. 3번째 서브넷도 프라이빗 서브넷이다. 다음의 각 값을 입력 후, [서브넷 생성]을 클릭한다.

- A. [서브넷 이름] : lab-private2-subnet
- B. [가용 영역] : 아시아 태평양(서울) / ap-northeast-2d
- C. [IPv4 CIDR 블록] : 10.0.30.0/24
- D. [키] : Name
- E. [값] : lab-private2-subnet

### 서브넷 설정

서브넷의 CIDR 블록 및 가용 영역을 지정합니다.

1/1개 서브넷

서브넷 이름

'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

이름은 최대 256자까지 입력할 수 있습니다.

가용 영역 정보

서브넷이 상주할 영역을 선택합니다. 선택하지 않으면 Amazon이 자동으로 선택합니다.

아시아 태평양 (서울) / ap-northeast-2d

IPv4 CIDR 블록 정보

▼ 태그 - 선택 사항

키

값 - 선택 사항

제거

새 태그 추가

49글(를) 태그.개 더 추가할 수 있습니다.

제거

새 서브넷 추가

취소

서브넷 생성

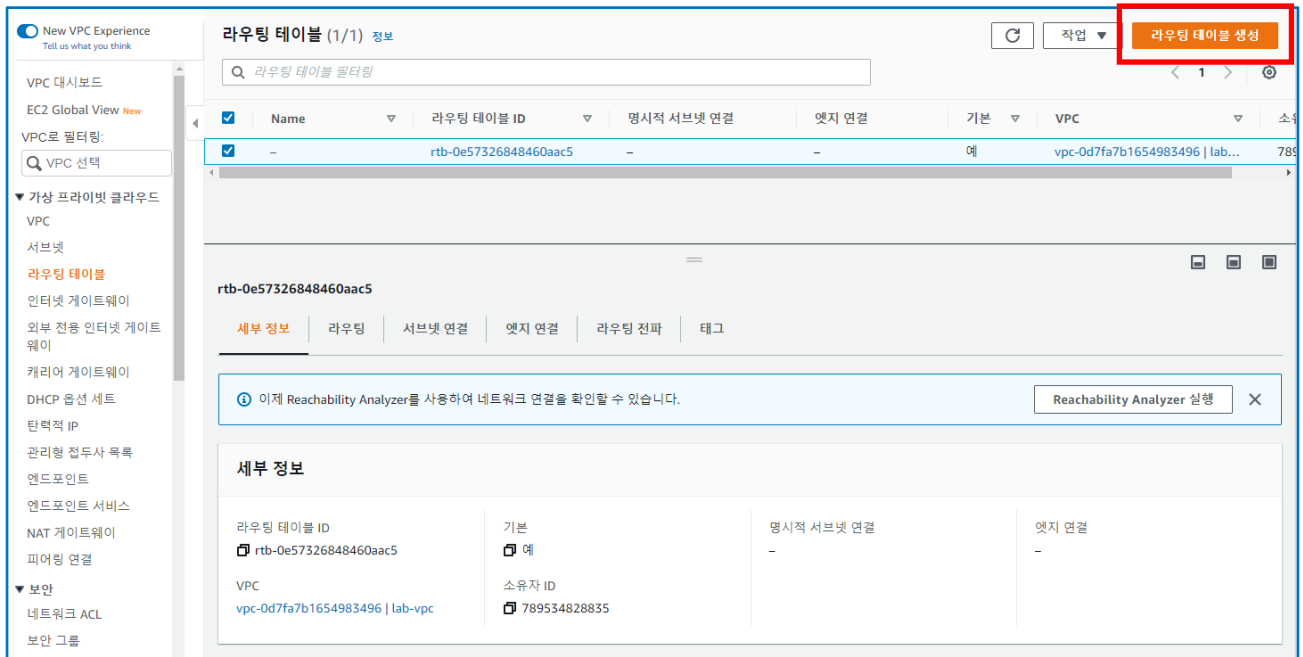
13. 이렇게 1개의 퍼블릭 서브넷과 2개의 프라이빗 서브넷을 생성했다.

서브넷 1개를 성공적으로 생성하였습니다. subnet-00b712f1a57904766

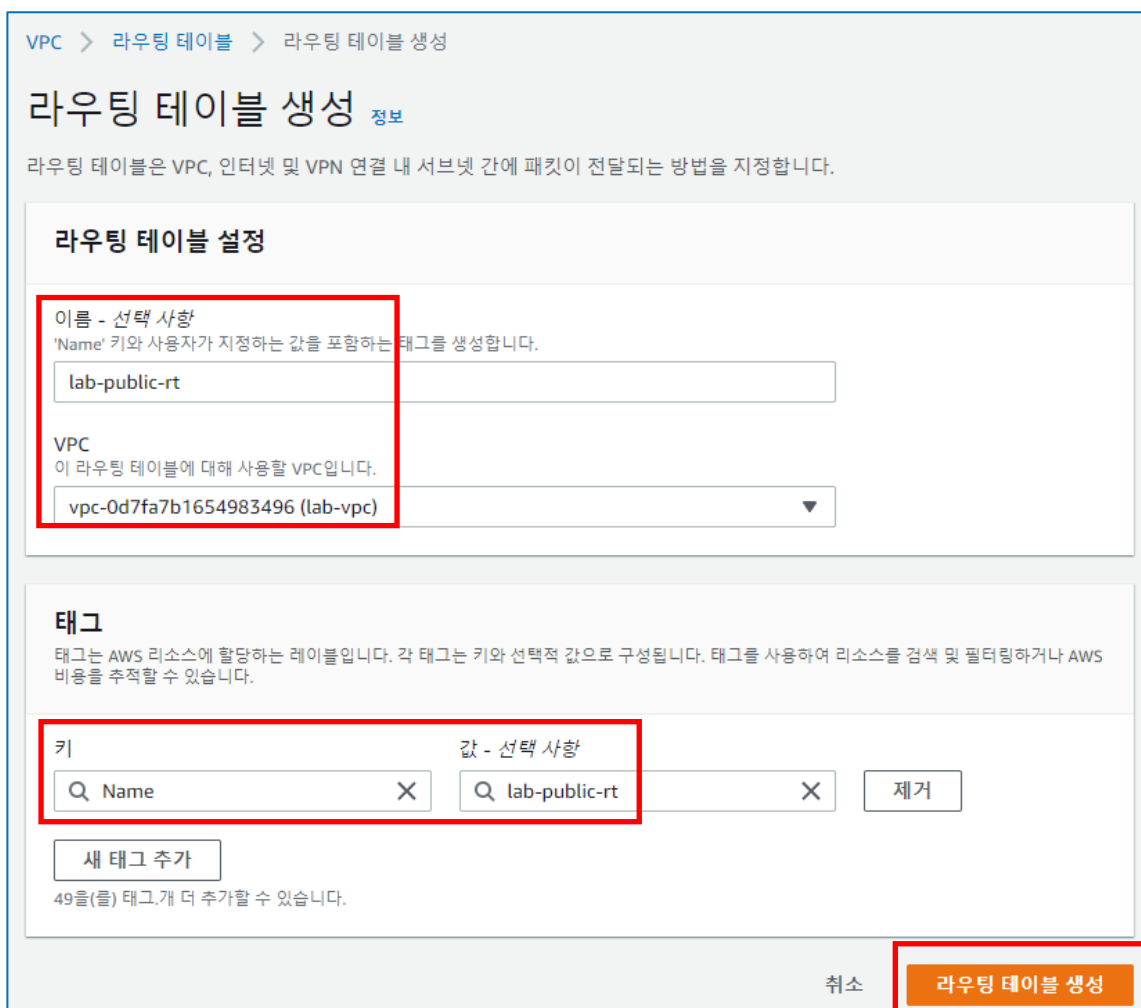
서브넷 (3) 정보

<input type="checkbox"/>	Name	서브넷 ID	상태	VPC	IPv4 CIDR	가용 영역
<input type="checkbox"/>	lab-private2-subnet	subnet-00b712f1a57904766	Available	vpc-0d7fa7b1654983496   lab...	10.0.30.0/24	ap-northeast-2d
<input type="checkbox"/>	lab-private1-subnet	subnet-0b86674deed6ee5b9	Available	vpc-0d7fa7b1654983496   lab...	10.0.20.0/24	ap-northeast-2c
<input type="checkbox"/>	lab-public-subnet	subnet-0fcb894b841c2850f	Available	vpc-0d7fa7b1654983496   lab...	10.0.10.0/24	ap-northeast-2a

14. 라우팅 테이블 생성 차례이다. 좌측 메뉴 중 [라우팅 테이블]을 선택해서 라우팅 테이블페이지로 이동한 후, 우측 상단의 [라우팅 테이블 생성]을 클릭한다.



15. [라우팅 테이블 생성] 페이지이다. [이름]은 lab-public-rt로, 그리고 [VPC]는 lab-vpc를 선택한다. [태그] 섹션에서는 자동으로 입력된 값을 확인하고 [라우팅 테이블 생성]을 클릭한다.



16. **lab-public-rt** 라우팅 테이블이 생성되면 **[라우팅]** 섹션에서 **[라우팅 편집]**을 클릭하여 인터넷 게이트웨이까지의 **라우팅**을 설정하도록 한다.

VPC > 라우팅 테이블 > rtb-03a16e0615162ad0f

### rtb-03a16e0615162ad0f / lab-public-rt

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. [Reachability Analyzer 실행](#)

#### 세부 정보 정보

라우팅 테이블 ID rtb-03a16e0615162ad0f	기본 아니요	명시적 서브넷 연결 -	옛지 연결 -
VPC vpc-0d7fa7b1654983496   lab-vpc	소유자 ID 789534828835		

**라우팅** | 서브넷 연결 | 옛지 연결 | 라우팅 전파 | 태그

#### 라우팅 (1)

라우팅 필터링:  모두

대상	대상	상태	전파됨
10.0.0.0/16	local	활성	아니요

17. **[라우팅 편집]** 페이지에서, **[라우팅 추가]**를 클릭하여 다음의 값을 설정한 후, **[변경 사항 저장]**을 클릭한다.

A. **[대상]** : 0.0.0.0/0

B. **[대상]** : 인터넷 게이트웨이 > lab-ig

VPC > 라우팅 테이블 > rtb-03a16e0615162ad0f > 라우팅 편집

### 라우팅 편집

대상	대상	상태	전파됨
10.0.0.0/16	local	활성	아니요
0.0.0.0/0	igw-069f5a2a1966383ff	-	아니요

[라우팅 추가](#)

취소 [미리 보기](#) [변경 사항 저장](#)

18. 이번에는 **lab-public-rt 라우팅 테이블** 페이지에서 [서브넷 연결] 탭을 클릭하여 이 라우팅 테이블이 **lab-public-subnet** 과 연결되도록 한다. [서브넷 연결 편집]에서 **lab-public-subnet**을 체크하여 선택한다. 그리고 [연결 저장]을 클릭한다.

VPC > 라우팅 테이블 > rtb-03a16e0615162ad0f > 서브넷 연결 편집

### 서브넷 연결 편집

이 라우팅 테이블과 연결된 서브넷을 변경합니다.

이용 가능한 서브넷 (1/3)

Q 서브넷 연결 필터링

	이름	서브넷 ID	IPv4 CIDR	IPv6 CIDR	라우팅 테이블 ID
<input type="checkbox"/>	lab-private2-subnet	subnet-00b712f1a57904766	10.0.30.0/24	-	기본 (rtb-0e57326848460aac5)
<input type="checkbox"/>	lab-private1-subnet	subnet-0b86674deed6ee5b9	10.0.20.0/24	-	기본 (rtb-0e57326848460aac5)
<input checked="" type="checkbox"/>	lab-public-subnet	subnet-0fcb894b841c2850f	10.0.10.0/24	-	기본 (rtb-0e57326848460aac5)

선택한 서브넷

subnet-0fcb894b841c2850f / lab-public-subnet X

취소 **연결 저장**

19. 2번째 라우팅 테이블을 생성한다. 다음의 각 값을 입력 후, [라우팅 테이블 생성]을 클릭한다.

- A. [이름] : lab-private-rt
- B. [VPC] : lab-vpc
- C. [태그] : [키]/Name, [값]/lab-private-rt

VPC > 라우팅 테이블 > 라우팅 테이블 생성

### 라우팅 테이블 생성 정보

라우팅 테이블은 VPC, 인터넷 및 VPN 연결 내 서브넷 간에 패킷이 전달되는 방법을 지정합니다.

#### 라우팅 테이블 설정

이름 - 선택 사항  
'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

lab-private-rt

VPC  
이 라우팅 테이블에 대해 사용할 VPC입니다.

vpc-0d7fa7b1654983496 (lab-vpc)

#### 태그

태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키: Q Name X

값 - 선택 사항: Q lab-private-rt X

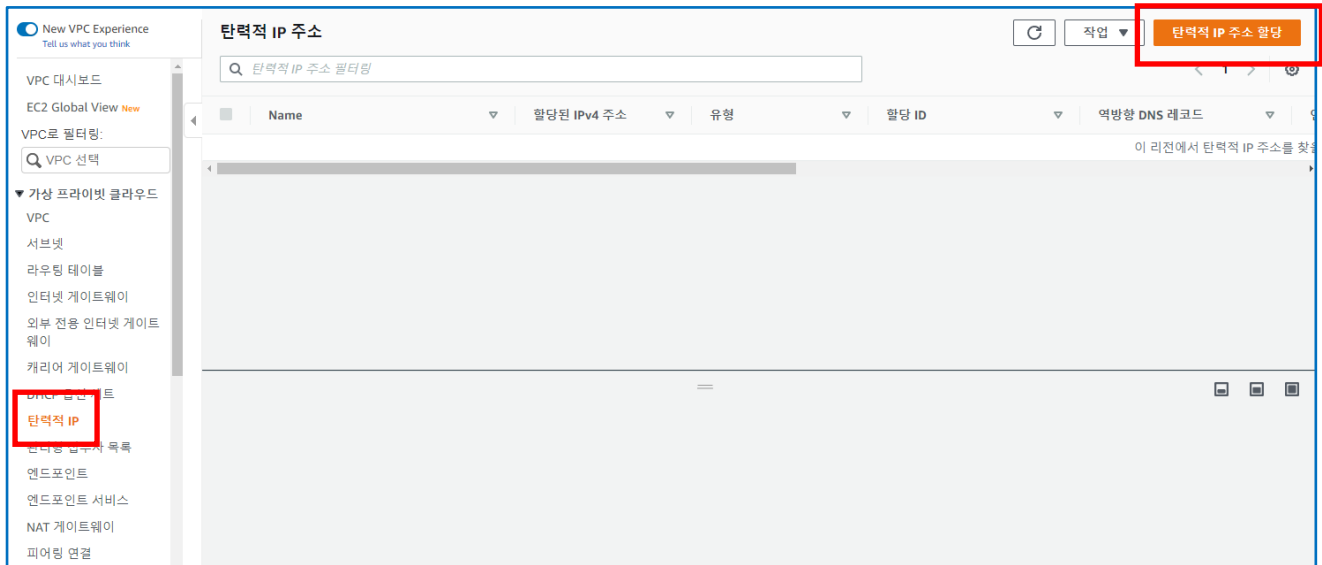
제거

새 태그 추가

49글(줄) 태그 개 더 추가할 수 있습니다.

취소 **라우팅 테이블 생성**

20. 프라이빗 서브넷은 NAT 게이트웨이와 연결되어야 한다. NAT 게이트웨이는 탄력적 IP 설정이 되어야 한다. 먼저, 좌측 메뉴 중 [탄력적 IP]를 클릭하여 [탄력적 IP 주소] 페이지로 이동한 후, 우측 상단의 [탄력적 IP 주소 할당] 버튼을 클릭한다.



21. [탄력적 IP 주소 설정] 페이지에서, 기본값 그대로 사용하기로 하고 [태그] 섹션에서 [키]는 Name으로, [값]은 lab-nat-eip를 입력한 후, [할당] 버튼을 클릭한다.

탄력적 IP 주소 설정 정보

네트워크 경계 그룹 정보

ap-northeast-2

퍼블릭 IPv4 주소 풀

☒ Amazon의 IPv4 주소 풀

☐ AWS 계정으로 가져오는 퍼블릭 IPv4 주소 (풀을 찾을 수 없으므로 옵션이 비활성화됨) 자세히 알아보기

☐ IPv4 주소의 고객 소유 풀 (고객 소유 풀을 찾을 수 없기 때문에 옵션이 비활성화됨) 자세히 알아보기

글로벌 정적 IP 주소

AWS Global Accelerator는 AWS 엣지 로케이션의 애니캐스트를 사용하여 전 세계에 발표된 글로벌 정적 IP 주소를 제공할 수 있습니다. 이를 통해 Amazon 글로벌 네트워크를 사용하여 사용자 트래픽의 가용성과 지연 시간을 개선할 수 있습니다. 자세히 알아보기

액셀러레이터 생성

태그 - 선택 사항

태그는 사용자가 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키

값 - 선택 사항

Name

lab-nat-eip

제거

새로운 태그 추가

최대 49개의 태그를 더 추가할 수 있습니다.

취소

할당

22. 좌측 메뉴 중 **[NAT 게이트웨이]**를 클릭하여 **[NAT 게이트웨이]** 페이지로 이동한다. 우측 상단의 **[NAT 게이트웨이 생성]** 버튼을 클릭하여 다음 그림과 같이 **[NAT 게이트웨이 생성]** 페이지로 이동했다. 다음의 각 값을 설정한다.

- A. **[이름]** : lab-nat
- B. **[서브넷]** : lab-public-subnet
- C. **[연결 유형]** : 퍼블릭
- D. **[탄력적 IP 할당 ID]** : lab-nat-eip

VPC > NAT 게이트웨이 > NAT 게이트웨이 생성

## NAT 게이트웨이 생성 정보

프라이빗 서브넷의 인스턴스가 다른 VPC, 온프레미스 네트워크 또는 인터넷의 서비스에 연결하는 데 사용할 수 있는 가용성이 뛰어난 관리형 NAT(Network Address Translation) 서비스입니다.

### NAT 게이트웨이 설정

**이름 - 선택 사항**  
'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

lab-nat

이름은 최대 256자까지 입력할 수 있습니다.

**서브넷**  
NAT 게이트웨이를 생성할 서브넷을 선택합니다.

subnet-0fcb894b841c2850f (lab-public-subnet)

**연결 유형**  
NAT 게이트웨이에 대한 연결 유형을 선택합니다.

☒ 퍼블릭  
☐ 프라이빗

**탄력적 IP 할당 ID 정보**  
NAT 게이트웨이에 탄력적 IP 주소를 할당합니다.

eipalloc-09b6b42dc8bd0856a (lab-nat-eip)

탄력적 IP 할당

23. 페이지를 스크롤다운하여 **[태그]** 섹션에서, **[키]**를 **Name**으로, **[값]**을 **lab-nat**로 설정한 후, **[NAT 게이트웨이 생성]**을 클릭한다.

**태그**  
태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키: Name      값 - 선택 사항: lab-nat

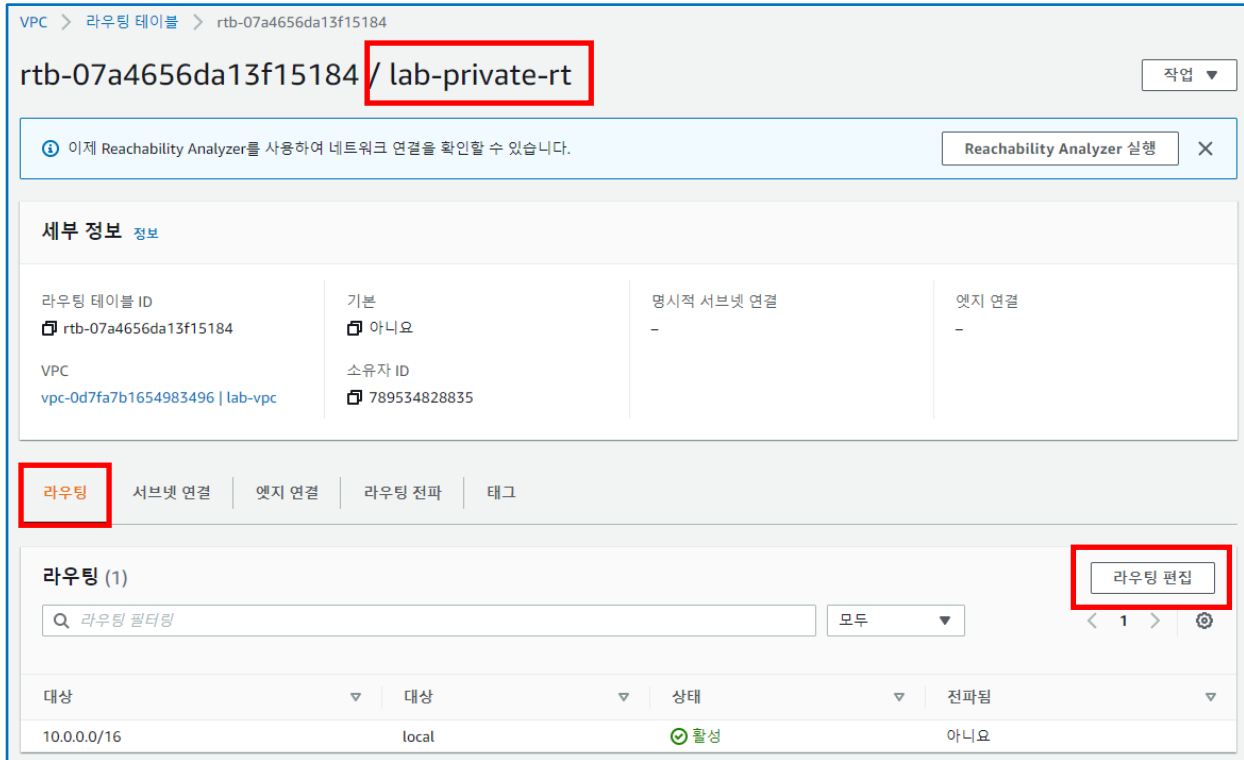
제거

새 태그 추가

49을(를) 태그 개 더 추가할 수 있습니다.

취소      **NAT 게이트웨이 생성**

24. 다시 라우팅 테이블 페이지로 이동하여 **lab-private-rt** 라우팅 테이블을 선택하고, [라우팅] 탭에서 [라우팅 편집]을 클릭한다.



VPC > 라우팅 테이블 > rtb-07a4656da13f15184

rtb-07a4656da13f15184 / lab-private-rt

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. Reachability Analyzer 실행

**세부 정보**

라우팅 테이블 ID rtb-07a4656da13f15184	기본 아니요	명시적 서브넷 연결 -	엣지 연결 -
VPC vpc-0d7fa7b1654983496   lab-vpc	소유자 ID 789534828835		

**라우팅** | 서브넷 연결 | 엣지 연결 | 라우팅 전파 | 태그

**라우팅 (1)**

라우팅 필터링: Q 라우팅 필터링 | 모두

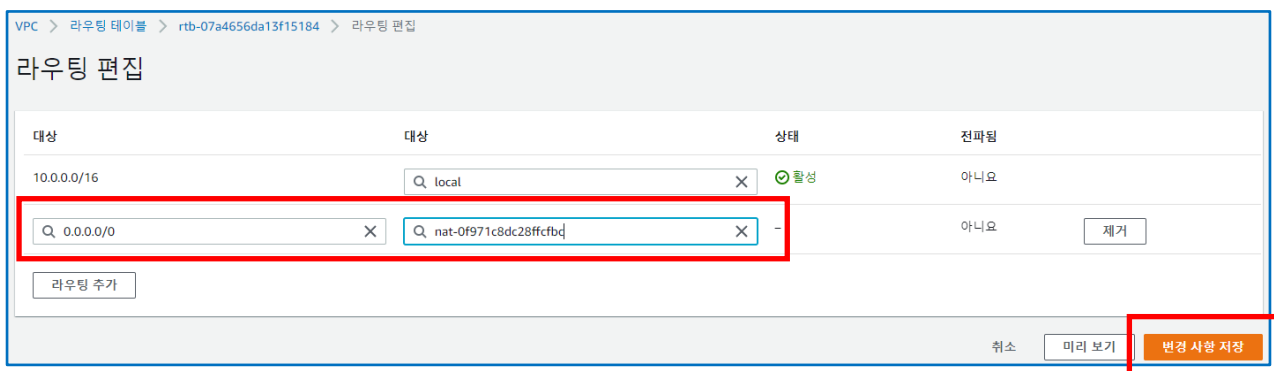
대상	대상	상태	전파됨
10.0.0.0/16	local	활성	아니요

라우팅 편집

25. [라우팅 추가] 버튼을 클릭하여 다음과 같이 값을 입력한 후, [변경 사항 저장]을 클릭한다.

A. [대상] : 0.0.0.0/0

B. [대상] : NAT 게이트웨이 > lab-nat



VPC > 라우팅 테이블 > rtb-07a4656da13f15184 > 라우팅 편집

**라우팅 편집**

대상	대상	상태	전파됨
10.0.0.0/16	local	활성	아니요
0.0.0.0/0	nat-0f971c8dc28ffcfd	-	아니요

라우팅 추가

취소 | 미리 보기 | **변경 사항 저장**

26. 좌측 메뉴 중 [서브넷] 메뉴를 클릭하여 [서브넷] 페이지로 이동한다. 서브넷 목록 중 **lab-private1-subnet**을 체크하여 선택한다. 그리고 페이지 아래의 여러 탭 중 [라우팅 테이블] 탭을 선택한다. [라우팅 테이블 연결 편집] 버튼을 클릭한다.

서브넷 (1/3) 정보

서브넷 필터링

1

서브넷 생성

Name	서브넷 ID	상태	VPC	IPv4 CIDR	가용 영역
lab-private2-subnet	subnet-00b312f1a57004766	Available	vpc-0d7fa7b1654983496   lab...	10.0.30.0/24	ap-northeast-2d
<input checked="" type="checkbox"/> lab-private1-subnet	subnet-0b86674deed6ee5b9	Available	vpc-0d7fa7b1654983496   lab...	10.0.20.0/24	ap-northeast-2c
lab-public-subnet	subnet-0fcb894b841c2850f	Available	vpc-0d7fa7b1654983496   lab...	10.0.10.0/24	ap-northeast-2a

subnet-0b86674deed6ee5b9 / lab-private1-subnet

세부 정보 | 플로우 로그 | 라우팅 테이블 | 네트워크 ACL | CIDR 예약 | 공유 중 | 태그

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다.

Reachability Analyzer 실행

라우팅 테이블: rtb-0e57326848460aac5

라우팅 테이블 연결 편집

라우팅 (1)

라우팅 필터링

1

대상	대상
10.0.0.0/16	local



27. [라우팅 테이블 연결 편집] 페이지이다. [라우팅 테이블 ID] 목록에서 **lab-private-rt**를 선택한다. 이렇게 하면 **lab-private1-subnet**은 **lab-private-rt** 라우팅 테이블과 연결되게 된다. [저장]을 클릭한다.

VPC > 서브넷 > subnet-0b86674deed6ee5b9 > 라우팅 테이블 연결 편집

## 라우팅 테이블 연결 편집 정보

### 서브넷 라우팅 테이블 설정

서브넷 ID  
subnet-0b86674deed6ee5b9

라우팅 테이블 ID  
rtb-07a4656da13f15184 (lab-private-rt) ▼ ↻

📘 이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. Reachability Analyzer 실행 ✕

### 라우팅 (2)

🔍 라우팅 필터링 < 1 > ⚙️

대상	대상
10.0.0.0/16	local
0.0.0.0/0	nat-0f971c8dc28ffcfbc

취소 저장

28. 이번에는 **lab-private2-subnet**을 **lab-private-rt** 라우팅 테이블과 연결할 차례이다. 즉, 프라이빗 서브넷은 모두 **lab-private-rt** 같은 라우팅 테이블과 연결하는 것이다. [라우팅 테이블] 탭의 [라우팅 테이블 연결 편집]을 클릭한다.

서브넷 (1/3) 정보

Name	서브넷 ID	상태	VPC	IPv4 CIDR	가용 영역
<input checked="" type="checkbox"/> lab-private2-subnet	subnet-00b712f1a57904766	Available	vpc-0d7fa7b1654983496   lab...	10.0.30.0/24	ap-northeast-2d
<input type="checkbox"/> lab-private1-subnet	subnet-0b86674deed6ee5b9	Available	vpc-0d7fa7b1654983496   lab...	10.0.20.0/24	ap-northeast-2c
<input type="checkbox"/> lab-public-subnet	subnet-0fcb894b841c2850f	Available	vpc-0d7fa7b1654983496   lab...	10.0.10.0/24	ap-northeast-2a

subnet-00b712f1a57904766 / lab-private2-subnet

세부 정보 | 플로우 로그 | **라우팅 테이블** | 네트워크 ACL | CIDR 예약 | 공유 중 | 태그

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. Reachability Analyzer 실행

라우팅 테이블: **rtb-0e57326848460aac5** 라우팅 테이블 연결 편집

라우팅 (1)

대상	대상
10.0.0.0/16	local

29. **lab-private2-subnet** 역시 **lab-private-rt** 라우팅 테이블과 연결했다. [저장] 버튼을 클릭한다.

VPC > 서브넷 > subnet-00b712f1a57904766 > 라우팅 테이블 연결 편집

## 라우팅 테이블 연결 편집 정보

서브넷 라우팅 테이블 설정

서브넷 ID

라우팅 테이블 ID

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다. Reachability Analyzer 실행

라우팅 (2)

대상	대상
10.0.0.0/16	local
0.0.0.0/0	nat-0f971c8dc28ffcfbc

취소 저장

30. 반대로 라우팅 테이블쪽에서 보면 lab-private-rt 라우팅 테이블은 lab-private1-subnet과 lab-private2-subnet 2개의 서브넷과 연결되어 있음을 확인할 수 있다.

라우팅 테이블 (1/1) 정보

라우팅 테이블 ID: rtb-07a4656da13f15184

Name	라우팅 테이블 ID	명시적 서브넷 연결	엣지 연결	기본	VPC
lab-private-rt	rtb-07a4656da13f15184	2 서브넷	-	아니요	vpc-0d7fa7b1654983496   lab...

rtb-07a4656da13f15184 / lab-private-rt

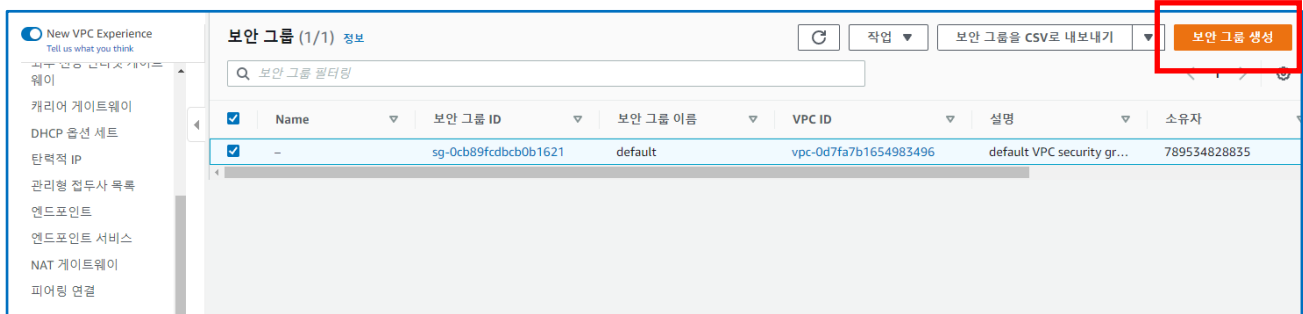
세부 정보 | 라우팅 | 서브넷 연결 | 엣지 연결 | 라우팅 전파 | 태그

명시적 서브넷 연결 (2)

서브넷 ID	IPv4 CIDR	IPv6 CIDR
subnet-00b712f1a57904766 / lab-private2-subnet	10.0.30.0/24	-
subnet-0b86674deed6ee5b9 / lab-private1-subnet	10.0.20.0/24	-

## 웹 서버와 DB 인스턴스를 위한 VPC 보안 그룹 생성

- 이번에는 보안 그룹을 생성할 차례이다. 좌측 메뉴 중 [보안 그룹]을 클릭하여 보안 그룹 페이지로 이동한다. 그리고 우측 상단의 [보안 그룹 생성]을 클릭한다.



2. [보안 그룹 생성] 페이지이다. 2개의 보안 그룹을 생성할 것이다. 먼저 웹 서버가 사용할 보안 그룹이다. 다음의 각 값을 설정한다.

A. [보안 그룹 이름] : lab-web-sg

B. [설명] : Web Security Group for Lab1

C. [VPC] : lab-vpc

[인바운드 규칙] 섹션에서 [규칙 추가]를 클릭한다.

D. [유형] : SSH

E. [소스] : Anywhere 0.0.0.0/0

F. [유형] : HTTP

G. [소스] : Anywhere 0.0.0.0/0

VPC > 보안 그룹 > 보안 그룹 생성

### 보안 그룹 생성 정보

보안 그룹은 인바운드 및 아웃바운드 트래픽을 관리하는 인스턴스의 가상 방화벽 역할을 합니다. 새 보안 그룹을 생성하려면 아래의 필드를 작성하십시오.

**기본 세부 정보**

**보안 그룹 이름** 정보  
lab-web-sg  
생성 후에는 이름을 편집할 수 없습니다.

**설명** 정보  
Web Security Group for Lab1

**VPC** 정보  
vpc-0d7fa7b1654983496 X

**인바운드 규칙** 정보

유형	프로토콜	포트 범위	소스	설명 - 선택 사항
SSH	TCP	22	Anywhere... 0.0.0.0/0 X	<input type="text"/> 삭제
HTTP	TCP	80	Anywhere... 0.0.0.0/0 X	<input type="text"/> 삭제

3. 페이지를 스크롤다운하여 [태그] 섹션에서, [키]는 Name으로, [값]은 lab-web-sg로 입력하고, [보안 그룹 생성]을 클릭한다.

### 태그 선택 사항

태그는 사용자가 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

**키**  
Name X

**값 - 선택 사항**  
lab-web-sg X

최대 49개의 태그를 더 추가할 수 있습니다.

4. **lab-web-sg** 보안 그룹이 성공적으로 생성되었다. 2번째 보안 그룹 생성을 위하여 **[보안 그룹 생성]**을 클릭한다.

보안 그룹 (1/2) 정보							보안 그룹 생성
Q 보안 그룹 필터링							< 1 > ⚙
<input type="checkbox"/>	Name	보안 그룹 ID	보안 그룹 이름	VPC ID	설명	소유자	
<input checked="" type="checkbox"/>	-	sg-0cb89fcdcb0b1621	default	vpc-0d7fa7b1654983496	default VPC security gr...	789534828835	
<input type="checkbox"/>	lab-web-sg	sg-0ddf74be0d3a71156	lab-web-sg	vpc-0d7fa7b1654983496	Web Security Group fo...	789534828835	

5. 2번째 보안 그룹은 데이터베이스 보안 그룹이다. **[보안 그룹 생성]** 페이지에서 다음의 각 값을 입력한다.

A. **[보안 그룹 이름]** : lab-db-sg

B. **[설명]** : DB Security Group for Lab1

C. **[VPC]** : lab-vpc

**[인바운드 규칙]** 섹션에서 **[규칙 추가]**를 클릭한다.

D. **[유형]** : MYSQL/Aurora

E. **[소스]** : Anywhere 0.0.0.0/0

VPC > 보안 그룹 > 보안 그룹 생성

### 보안 그룹 생성 정보

보안 그룹은 인바운드 및 아웃바운드 트래픽을 관리하는 인스턴스의 가상 방화벽 역할을 합니다. 새 보안 그룹을 생성하려면 아래의 필드를 작성하십시오.

#### 기본 세부 정보

**보안 그룹 이름 정보**  
lab-db-sg  
생성 후에는 이름을 편집할 수 없습니다.

**설명 정보**  
DB Security Group for Lab1

**VPC 정보**  
vpc-0d7fa7b1654983496

#### 인바운드 규칙 정보

유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
MYSQL/Aurora	TCP	3306	Anywh... 0.0.0.0/0	

규칙 추가

6. 페이지를 스크롤다운하여 [태그] 섹션에서, [키]는 Name으로, [값]은 lab-db-sg로 입력하고, [보안 그룹 생성]을 클릭한다.

**태그 선택 사항**  
태그는 사용자가 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키:  X    값 - 선택 사항:  X   

최대 49개의 태그를 더 추가할 수 있습니다.

7. 이렇게 2개의 보안 그룹을 생성하였다.

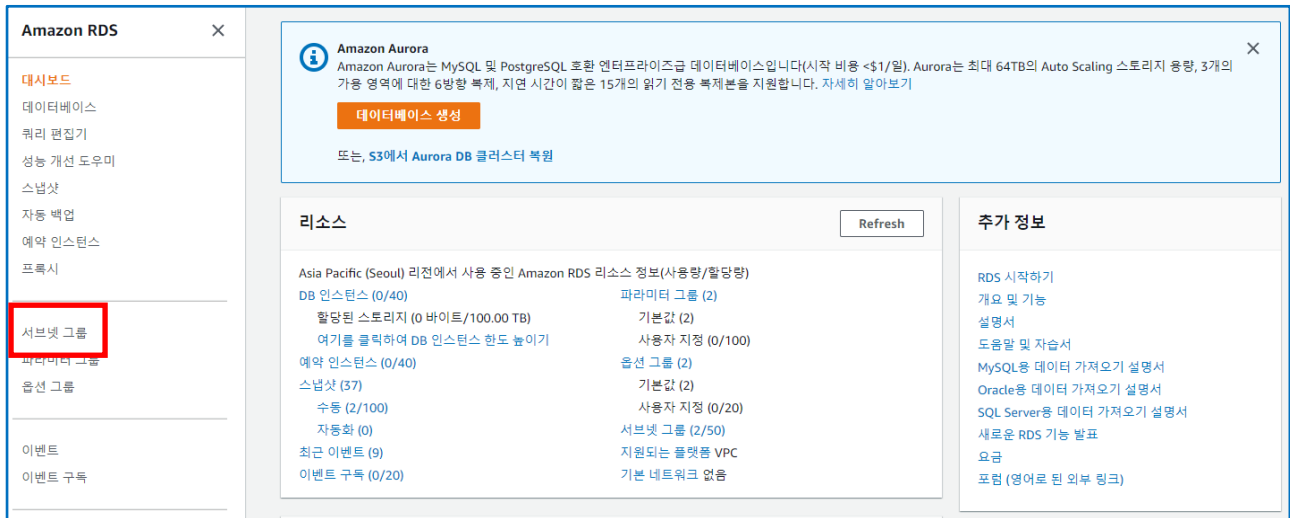
**보안 그룹 (1/3) 정보**               

   < 1 >    ⚙

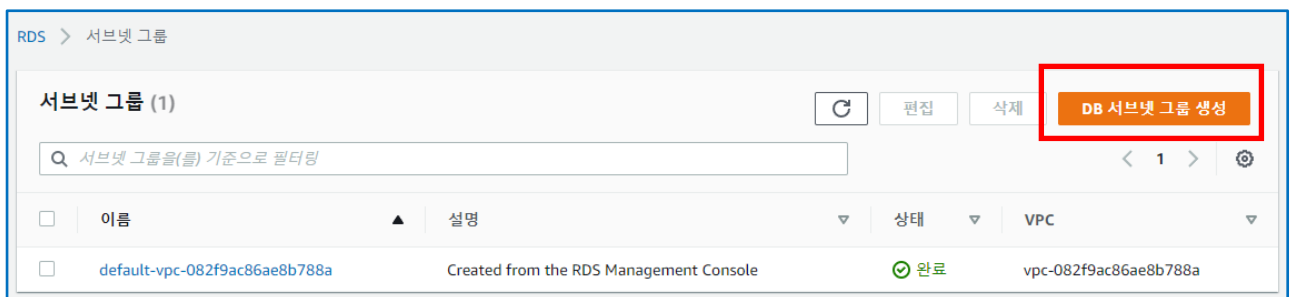
<input type="checkbox"/>	Name	보안 그룹 ID	보안 그룹 이름	VPC ID	설명	소유자
<input type="checkbox"/>	lab-db-sg	sg-083181e1d3d367c8c	lab-db-sg	vpc-0d7fa7b1654983496	DB Security Group for ...	789534828835
<input checked="" type="checkbox"/>	-	sg-0cb89fcdcb0b1621	default	vpc-0d7fa7b1654983496	default VPC security gr...	789534828835
<input type="checkbox"/>	lab-web-sg	sg-0ddf74be0d3a71156	lab-web-sg	vpc-0d7fa7b1654983496	Web Security Group fo...	789534828835

# DB 서브넷 그룹 만들기

1. [서비스] > [데이터베이스] > [RDS]를 클릭하여 Amazon RDS 페이지로 이동한다. 좌측 메뉴 중, [서브넷 그룹] 메뉴를 클릭한다. 릭한다.



2. [서브넷 그룹] 페이지이다. 페이지 우측 상단의 [DB 서브넷 그룹 생성] 버튼을 클릭한다.





3. [DB 서브넷 그룹 생성] 페이지이다. 다음의 각 값을 설정하고, [생성] 버튼을 클릭한다.

A. [이름] : lab-db-subnet-group

B. [설명] : DB Subnet Group

C. [VPC] : lab-vpc

RDS > 서브넷 그룹 > DB 서브넷 그룹 생성

## DB 서브넷 그룹 생성

새 서브넷 그룹을 생성하려면 이름과 설명을 입력하고 기존 VPC를 선택합니다. 그러면 해당 VPC와 관련된 서브넷을 추가할 수 있습니다.

### 서브넷 그룹 세부 정보

이름  
서브넷 그룹이 생성된 후에는 이름을 수정할 수 없습니다.

lab-db-subnet-group

1~255자로 구성되어야 합니다. 영숫자, 공백, 하이픈, 밑줄 및 마침표를 사용할 수 있습니다.

설명  
DB Subnet Group

VPC  
DB 서브넷 그룹에 사용할 서브넷에 해당하는 VPC 식별자를 선택합니다. 서브넷 그룹이 생성된 후에는 다른 VPC 식별자를 선택할 수 없습니다.

lab-vpc (vpc-0d7fa7b1654983496)

D. [가용 영역] : ap-northeast-2c, ap-northeast-2d

E. [서브넷] : lab-private1-subnet, lab-private2-subnet

### 서브넷 추가

가용 영역  
추가할 서브넷이 포함된 가용 영역을 선택합니다.

가용 영역 선택

ap-northeast-2c X ap-northeast-2d X

서브넷  
추가할 서브넷을 선택합니다. 목록에는 선택한 가용 영역의 서브넷이 포함됩니다.

서브넷 선택

subnet-00b712f1a57904766 (10.0.30.0/24) X

subnet-0b86674deed6ee5b9 (10.0.20.0/24) X

#### 서브넷이 선택됨 (2)

가용 영역	서브넷 ID	CIDR 블록
ap-northeast-2d	subnet-00b712f1a57904766	10.0.30.0/24
ap-northeast-2c	subnet-0b86674deed6ee5b9	10.0.20.0/24

취소 생성

4. [서브넷 그룹]이 생성되었다.

RDS > 서브넷 그룹

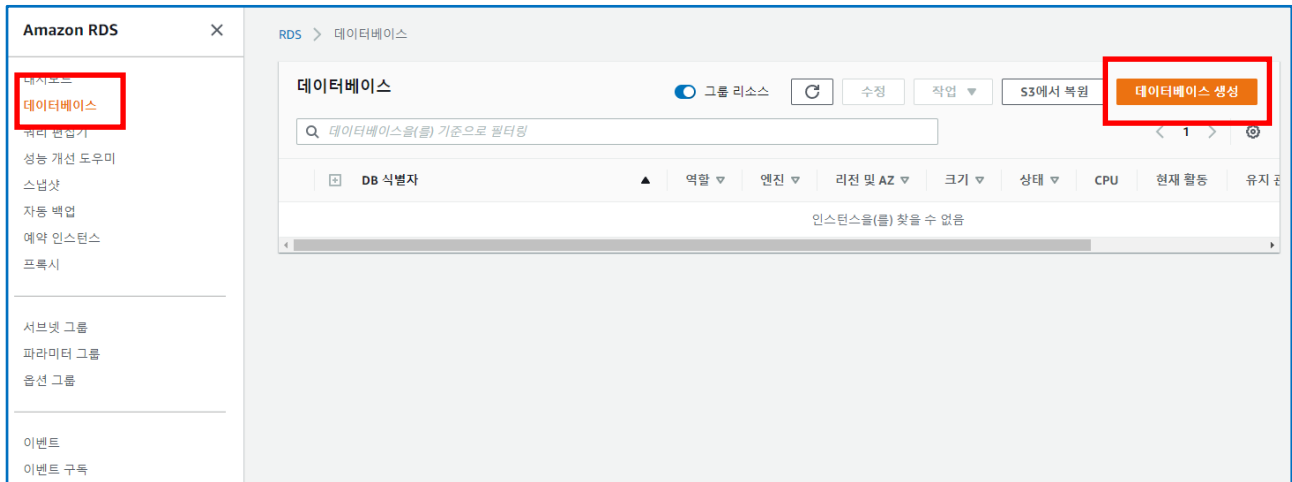
서브넷 그룹 (2) 🔄 편집 삭제 DB 서브넷 그룹 생성

🔍 서브넷 그룹을(를) 기준으로 필터링 < 1 > ⚙️

<input type="checkbox"/>	이름 ▲	설명 ▼	상태 ▼	VPC ▼
<input type="checkbox"/>	<a href="#">default-vpc-082f9ac86ae8b788a</a>	Created from the RDS Management Console	✅ 완료	vpc-082f9ac86ae8b788a
<input type="checkbox"/>	<a href="#">lab-db-subnet-group</a>	DB Subnet Group	✅ 완료	vpc-0d7fa7b1654983496

# DB 인스턴스 생성

1. 계속해서 **Amazon RDS** 페이지에서 좌측 메뉴 중 **[데이터베이스]**를 클릭하여 **데이터베이스** 페이지로 이동한다. 우측 상단의 **[데이터베이스 생성]** 버튼을 클릭한다.

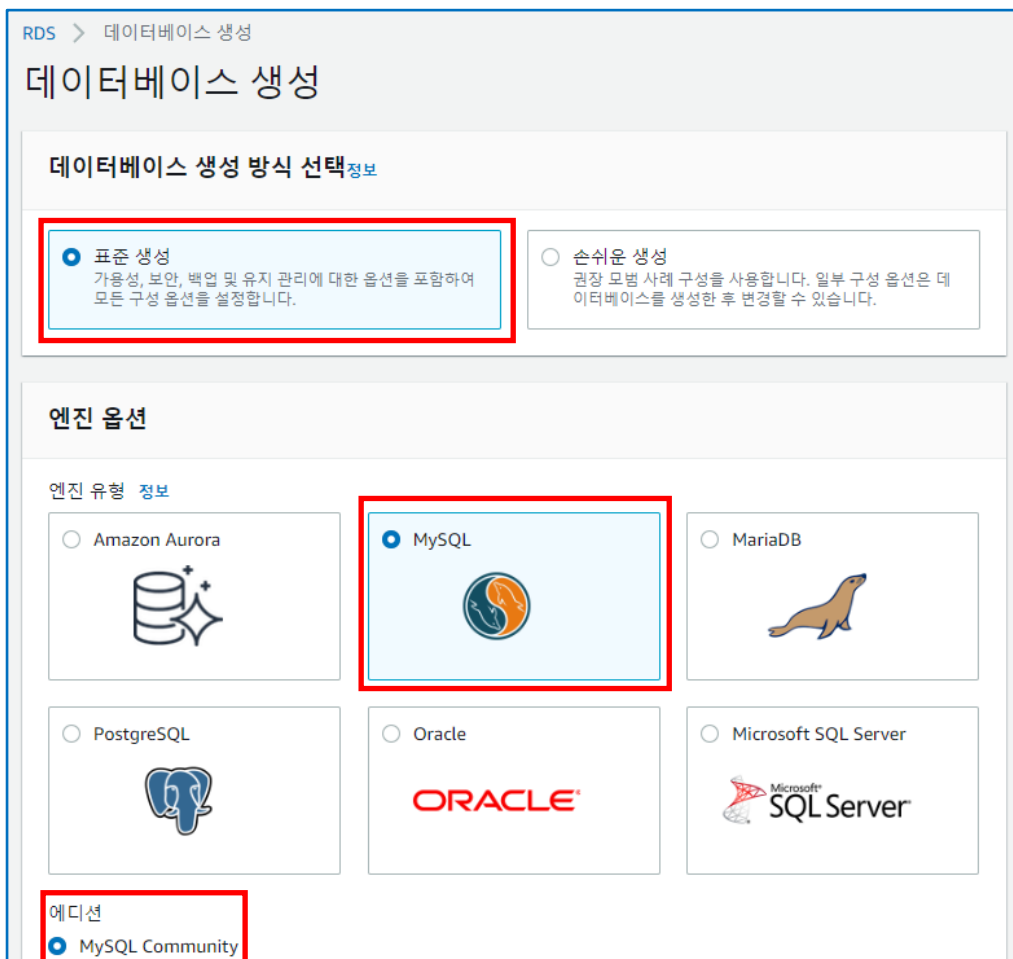


2. **[데이터베이스 생성]** 페이지이다. 다음의 각 값을 설정한다.

A. **[데이터베이스 생성 방식 선택]** : 표준 생성

B. **[엔진 유형]** : MySQL

C. **[에디션]** : MySQL Community



D. [버전] : MySQL 8.0.28

E. [템플릿] : 프리 티어

버전

MySQL 8.0.28 ▼

**템플릿**

해당 사용 사례를 충족하는 샘플 템플릿을 선택하세요.

☐ 프로덕션

고가용성 및 빠르고 일관된 성능을 위해 기본값을 사용하세요.

☐ 개발/테스트

이 인스턴스는 프로덕션 환경 외부에서 개발 용도로 마련되었습니다.

☒ 프리 티어

RDS 프리 티어를 사용하여 새로운 애플리케이션을 개발하거나, 기존 애플리케이션을 테스트하거나 Amazon RDS에서 실무 경험을 쌓을 수 있습니다. [정보](#)

F. [DB 인스턴스 식별자] : lab-db-instance

G. [마스터 사용자 이름] : admin

H. [마스터 암호] : suwonmymysql

I. [암호 확인] : suwonmymysql

**설정**

**DB 인스턴스 식별자** [정보](#)

DB 인스턴스 이름을 입력하세요. 이름은 현재 AWS 리전에서 AWS 계정이 소유하는 모든 DB 인스턴스에 대해 고유해야 합니다.

lab-db-instance

DB 인스턴스 식별자는 대소문자를 구분하지 않지만 'mydbinstance'와 같이 모두 소문자로 저장됩니다. 제약: 1자~60자의 영숫자 또는 하이픈으로 구성되어야 합니다. 첫 번째 문자는 글자이어야 합니다. 하이픈 2개가 연속될 수 없습니다. 끝에 하이픈이 올 수 없습니다.

▼ 자격 증명 설정

**마스터 사용자 이름** [정보](#)

DB 인스턴스의 마스터 사용자에게 로그인 ID를 입력하세요.

admin

1~16자의 영숫자. 첫 번째 문자는 글자여야 합니다.

☐ 암호 자동 생성

Amazon RDS에서 사용자를 대신하여 암호를 생성하거나 사용자가 직접 암호를 지정할 수 있습니다.

**마스터 암호** [정보](#)

.....

제약 조건: 8자 이상의 인쇄 가능한 ASCII 문자. 다음은 포함할 수 없습니다. /(슬래시), '(작은따옴표), "(큰따옴표) 및 @(앳 기호).

**암호 확인** [정보](#)

.....

J. [DB 인스턴스 클래스] : 버스터블 클래스(t 클래스 포함) / db.t2.micro

K. [이전 세대 클래스 포함] : 선택

### 인스턴스 구성

아래의 DB 인스턴스 구성 옵션은 위에서 선택한 엔진에서 지원하는 옵션으로 제한됩니다.

#### DB 인스턴스 클래스 정보

- ☐ 스탠다드 클래스(m 클래스 포함)
- ☐ 메모리 최적화 클래스(r 및 x 클래스 포함)
- ☒ 버스터블 클래스(t 클래스 포함)

db.t2.micro  
1 vCPUs 1 GiB RAM Not EBS Optimized

☒ 이전 세대 클래스 포함

L. [Virtual Private Cloud(VPC)] : lab-vpc

M. [서브넷 그룹] : lab-db-subnet-group

N. [퍼블릭 액세스] : 아니요

### 연결

#### Virtual Private Cloud(VPC) 정보

이 DB 인스턴스의 가상 네트워킹 환경을 정의하는 VPC.

lab-vpc (vpc-0d7fa7b1654983496)

해당 DB 서브넷 그룹이 있는 VPC만 나열됩니다.

❗ 데이터베이스를 생성한 후에는 VPC를 변경할 수 없습니다.

#### 서브넷 그룹 정보

선택한 VPC에서 DB 인스턴스가 어떤 서브넷과 IP 범위를 사용할 수 있는지를 정의하는 DB 서브넷 그룹.

lab-db-subnet-group

#### 퍼블릭 액세스 정보

- ☐ 예  
VPC 외부의 Amazon EC2 인스턴스 및 디바이스는 데이터베이스에 연결할 수 있습니다. 데이터베이스에 연결할 수 있는 VPC 내부의 EC2 인스턴스 및 디바이스를 지정하는 하나 이상의 VPC 보안 그룹을 선택하세요.
- ☒ 아니요  
RDS는 데이터베이스에 퍼블릭 IP 주소를 할당하지 않습니다. VPC 내부의 Amazon EC2 인스턴스 및 디바이스만 데이터베이스에 연결할 수 있습니다.

O. [VPC 보안 그룹] : 기존 항목 선택

P. [기존 VPC 보안 그룹] : lab-db-sg

Q. [가용 영역] : 기본 설정 없음

[추가 구성]을 클릭하여 확장 후,

R. [데이터베이스 포트] : 3306

VPC 보안 그룹  
데이터베이스에 대한 액세스를 허용할 VPC 보안 그룹을 선택합니다. 보안 그룹 규칙이 적절한 수신 트래픽을 허용하는지 확인합니다.

☒ 기존 항목 선택  
기존 VPC 보안 그룹 선택

☐ 새로 생성  
새 VPC 보안 그룹 생성

기존 VPC 보안 그룹  
VPC 보안 그룹 선택 ▼

lab-db-sg X

가용 영역 정보  
기본 설정 없음 ▼

▼ 추가 구성

데이터베이스 포트 정보  
데이터베이스가 애플리케이션 연결에 사용할 TCP/IP 포트입니다.

3306

3. 페이지를 스크롤다운하여 [데이터베이스 생성] 버튼을 클릭한다 .

▶ 추가 구성  
데이터베이스 옵션, 백업 활성화됨, 역추적 비활성화됨, 향상된 모니터링 비활성화됨, 유지 관리, CloudWatch Logs, 삭제 보호 비활성화됨.

월별 추정 요금

Amazon RDS 프리 티어는 12개월 동안 사용할 수 있습니다. 매월 프리 티어를 통해 아래 나열된 Amazon RDS 리소스를 무료로 사용할 수 있습니다.

- 단일 AZ db.t2.micro, db.t3.micro 또는 db.t4g.micro 인스턴스에서 Amazon RDS를 750시간 사용.
- 20GB의 범용 스토리지(SSD).
- 20GB의 자동 백업 스토리지 및 사용자가 시작한 모든 DB 스냅샷.

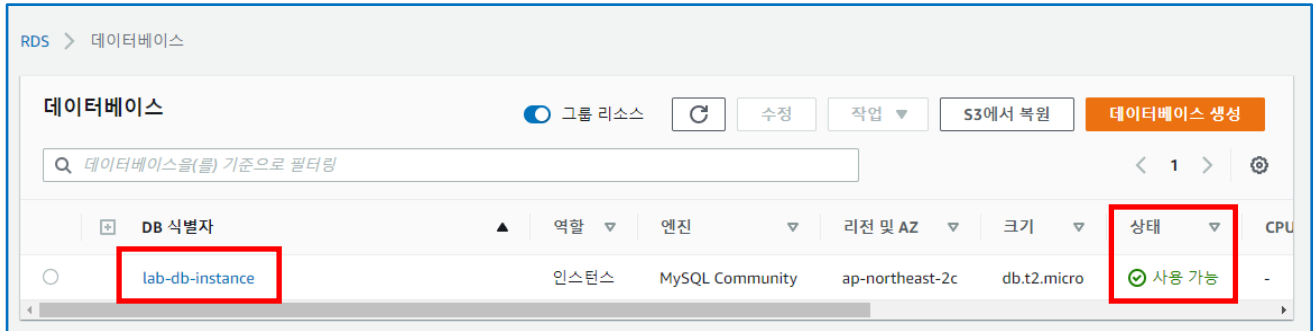
[AWS 무료 티어에 대해 자세히 알아보세요.](#)

무료 사용이 만료되었거나 애플리케이션에서 프리 티어 사용량을 초과한 경우 [Amazon RDS 요금 페이지](#)에서 설명한 대로, 표준 종량 서비스 요금이 적용됩니다.

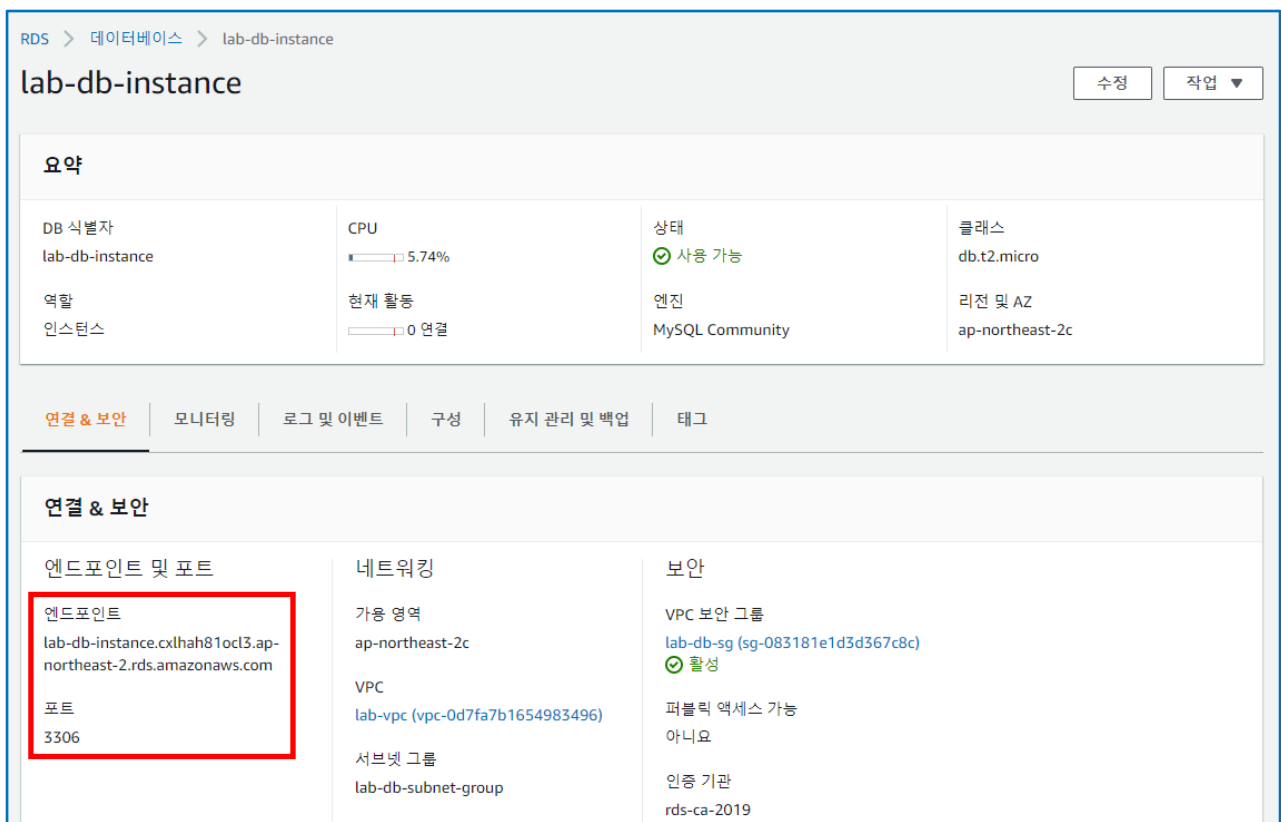
❗ 귀하는 AWS 서비스와 함께 사용하는 타사 제품 또는 서비스 일체에 대해 필요한 모든 권리를 보유할 책임이 있습니다.

취소 **데이터베이스 생성**

4. 몇 분의 시간이 흐른 뒤, **MySQL 데이터베이스 인스턴스**가 성공적으로 생성되었다. **[상태]**가 사용 가능이 될 때까지 기다려야 한다. 방금 생성한 **lab-db-instance**를 클릭한다.

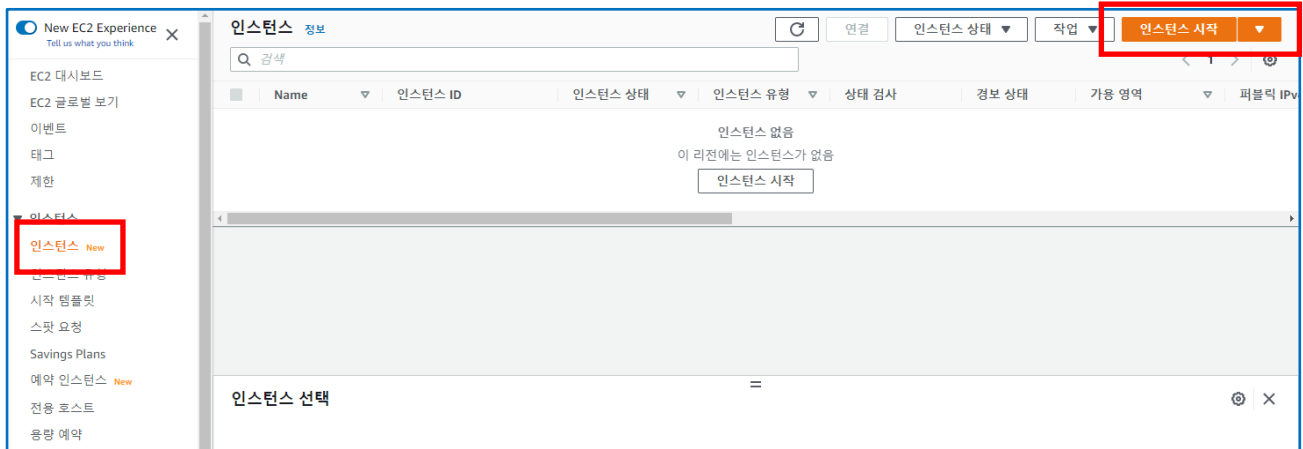


5. **lab-db-instance** 요약 페이지이다. **[연결 & 보안]** 섹션의 **[엔드포인트]** 및 **[포트]**를 확인한다.



## EC2 인스턴스 생성

1. [서비스] > [컴퓨팅] > [EC2]를 클릭하여 EC2 인스턴스 페이지로 이동한다. 페이지 우측 상단의 [인스턴스 시작]을 클릭한다.



2. 다음과 같이 **WebServer** 인스턴스를 생성한다.

A. [단계1 : AMI 선택] : Amazon Linux 2 AMI(HVM) – Kernel 5.10, SSD Volume Type, 64비트

B. [단계2:인스턴스 유형 선택] : t2.micro

C. [단계3:인스턴스 세부 정보 구성]

[네트워크] : lab-vpc

[서브넷] : lab-public-subnet

[퍼블릭 IP 자동 할당] : 활성화

네트워크	vpc-0d7fa7b1654983496   lab-vpc
기본 VPC가 없습니다. 새 기본 VPC 생성.	
서브넷	subnet-0fcb894b841c2850f   lab-public-subnet   ap-n
250개 IP 주소 사용 가능	
퍼블릭 IP 자동 할당	활성화
호스트 이름 유형	서브넷 사용 설정 (IP 이름)

D. [단계4:스토리지 추가] : 8GiB

E. [단계5:태그 추가] : Name/al-webserver-ec2

F. [단계6:보안 그룹 구성] : 기존 보안 그룹 선택 / lab-web-sg

G. [키 페어] : 새 키 페어 생성 > al-webserver-ec2-key.pem > 키 페어 다운로드



3. 다음과 같이 **Windows Server** 인스턴스를 생성한다.

A. [단계1 : AMI 선택] : **Microsoft Windows Server 2019 Base, 64비트**

B. [단계2:인스턴스 유형 선택] : **t2.micro**

C. [단계3:인스턴스 세부 정보 구성]

[네트워크] : **lab-vpc**

[서브넷] : **lab-public-subnet**

[퍼블릭 IP 자동 할당] : **활성화**

네트워크 ⓘ	vpc-0d7fa7b1654983496   lab-vpc 기본 VPC가 없습니다. 새 기본 VPC 생성.
서브넷 ⓘ	subnet-0fcb894b841c2850f   lab-public-subnet   ap-n 250개 IP 주소 사용 가능
퍼블릭 IP 자동 할당 ⓘ	활성화
호스트 이름 유형 ⓘ	서브넷 사용 설정 (IP 이름)

D. [단계4:스토리지 추가] : **30GiB**

E. [단계5:태그 추가] : **Name/windows-ec2**

F. [단계6:보안 그룹 구성] : **새 보안 그룹 생성 > lab-windows-sg**

G. [유형] : **RDP**

H. [프로토콜] : **TCP**

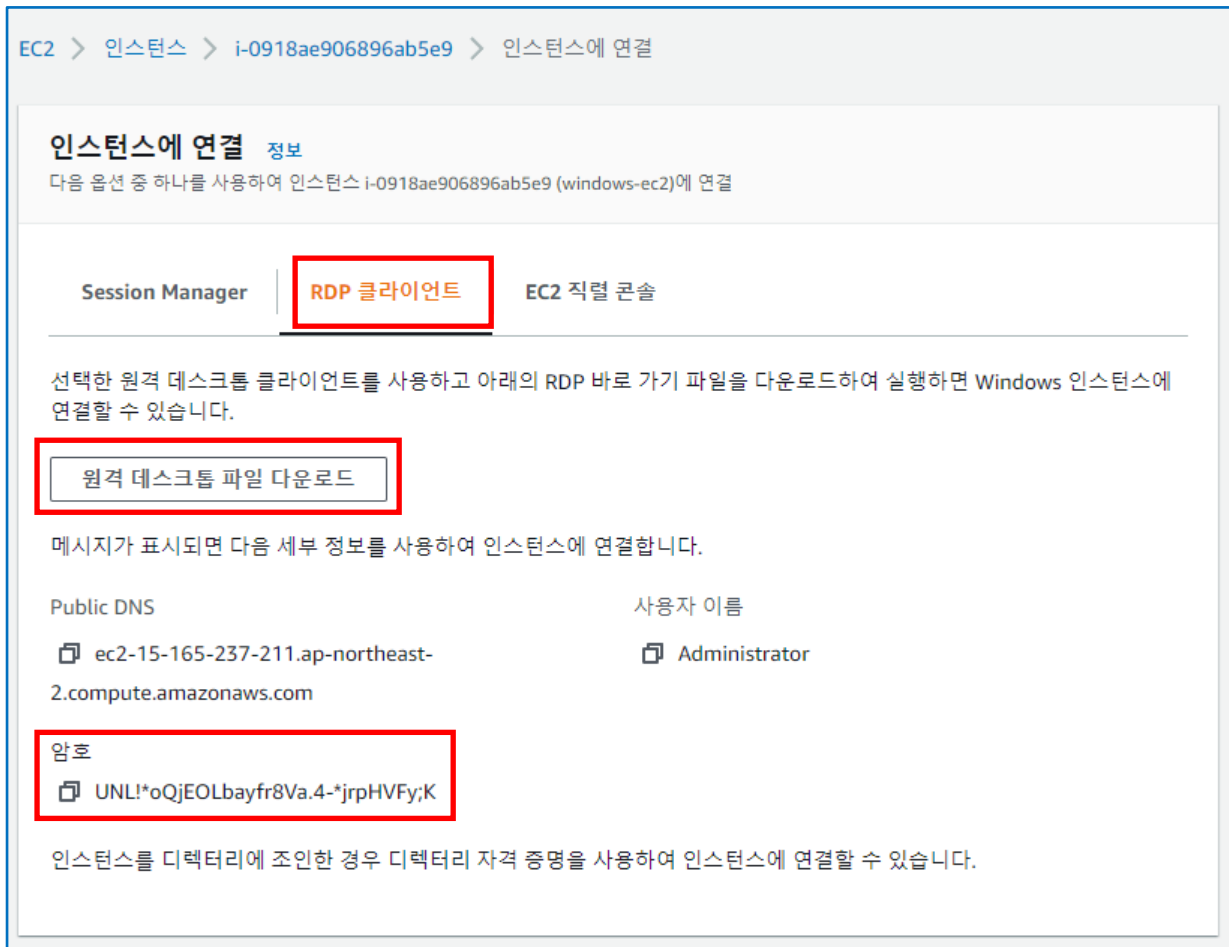
I. [포트 범위] : **3389**

J. [소스] : **사용자 지정, 0.0.0.0/0**

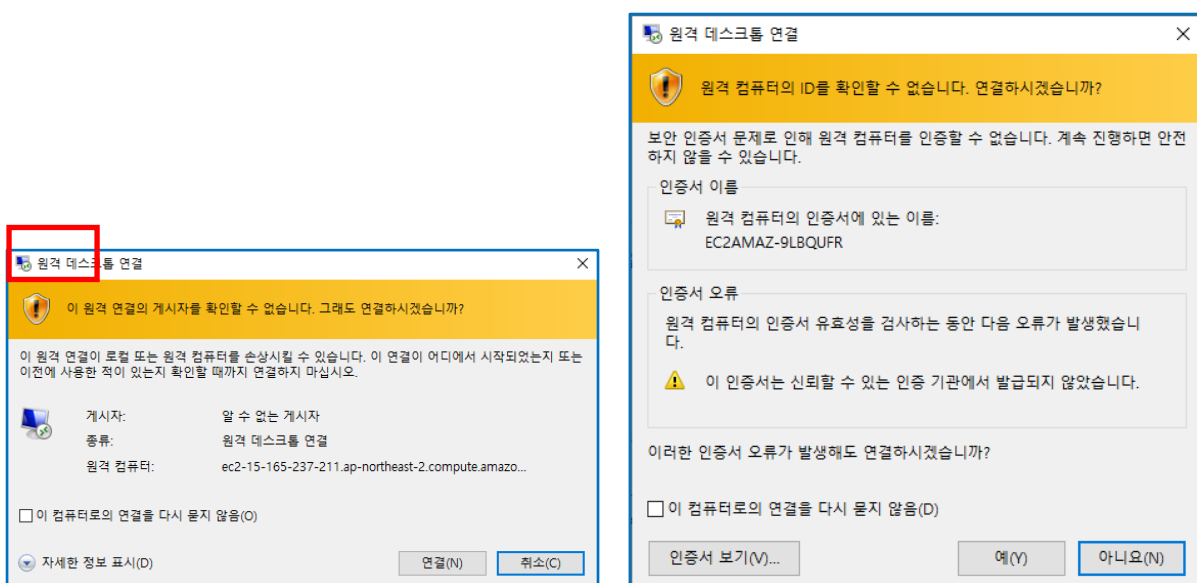
보안 그룹 할당: <input checked="" type="radio"/> 새 보안 그룹 생성 <input type="radio"/> 기존 보안 그룹 선택			
보안 그룹 이름: lab-windows-sg			
설명: launch-wizard-1 created 2022-05-22T16:05:25.486+09:00			
유형 ⓘ	프로토콜 ⓘ	포트 범위 ⓘ	소스 ⓘ
RDP ▼	TCP	3389	사용자 지정 ▼ 0.0.0.0/0
규칙 추가			

K. [키 페어] : **새 키 페어 생성 > windows-ec2-key.pem > 키 페어 다운로드**

4. **windows-ec2** 인스턴스가 생성되면 **[연결]** 버튼을 클릭하여 다음과 같이 **[인스턴스에 연결]** 페이지로 이동한다. **[RDP 클라이언트]** 탭으로 이동한 후, **[암호]**의 암호 가져오기를 클릭하여 **windows-ec2-key.pem**을 열고 암호를 해독한다. 그리고 **[원격 데스크톱 파일 다운로드]**를 클릭한다.



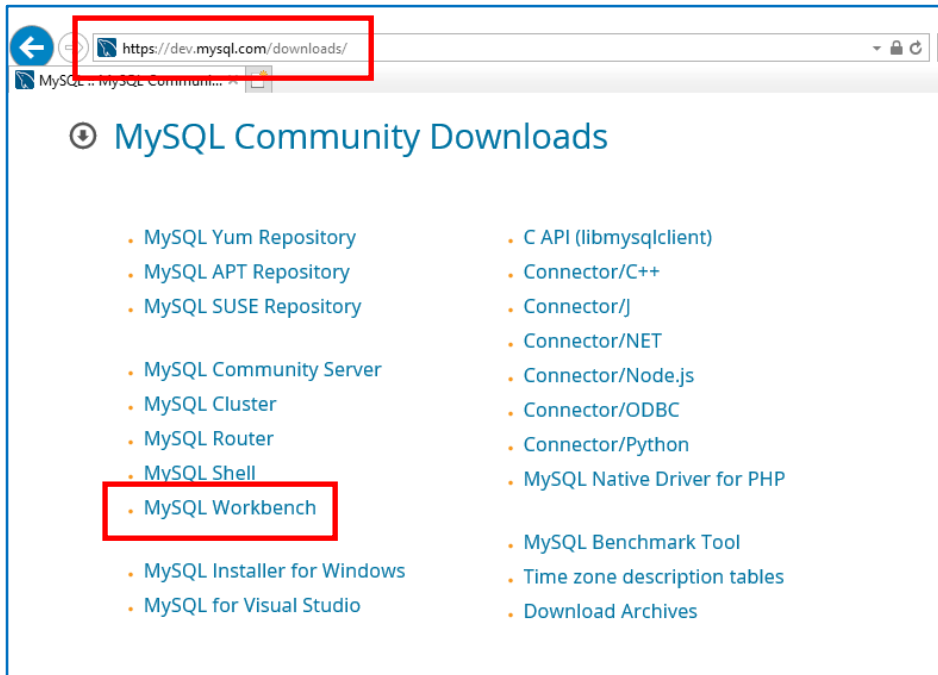
5. 만일 **[스캔]**으로 검색할 때, **[필터]**의 조건을 넣지 않으면 모든 항목을 다 검색하게 된다.



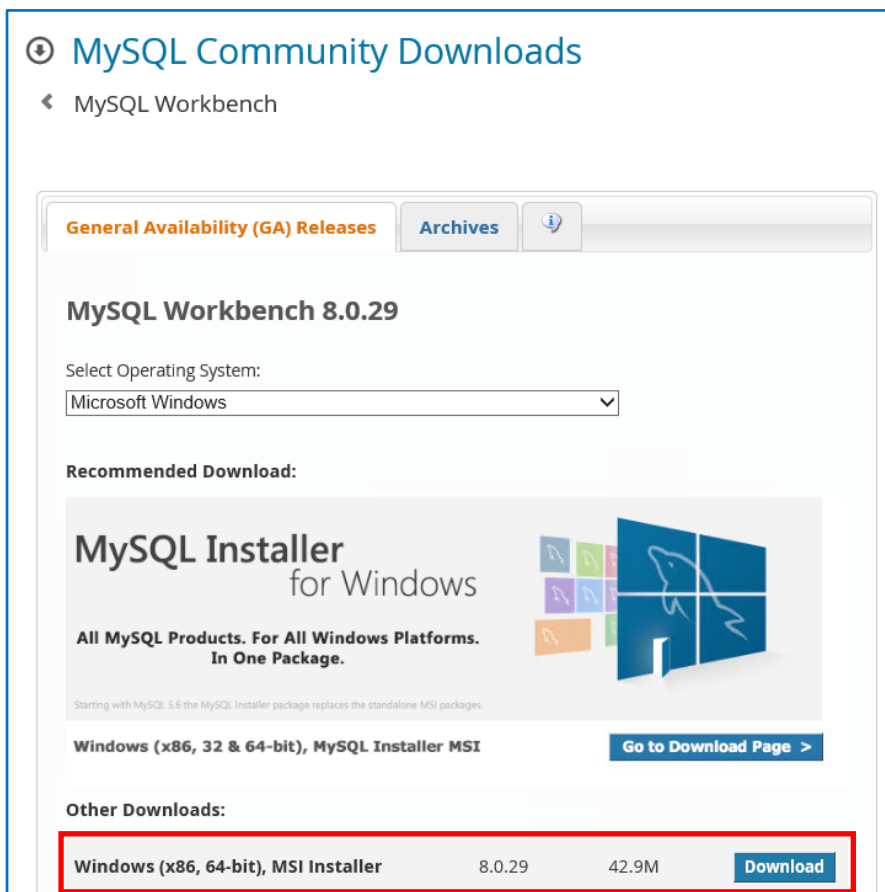
6. **Windows-ec2** 인스턴스에 연결한 후, 인터넷 익스플로러를 실행하여 다음 경로를 방문한다.

<https://dev.mysql.com/downloads/>

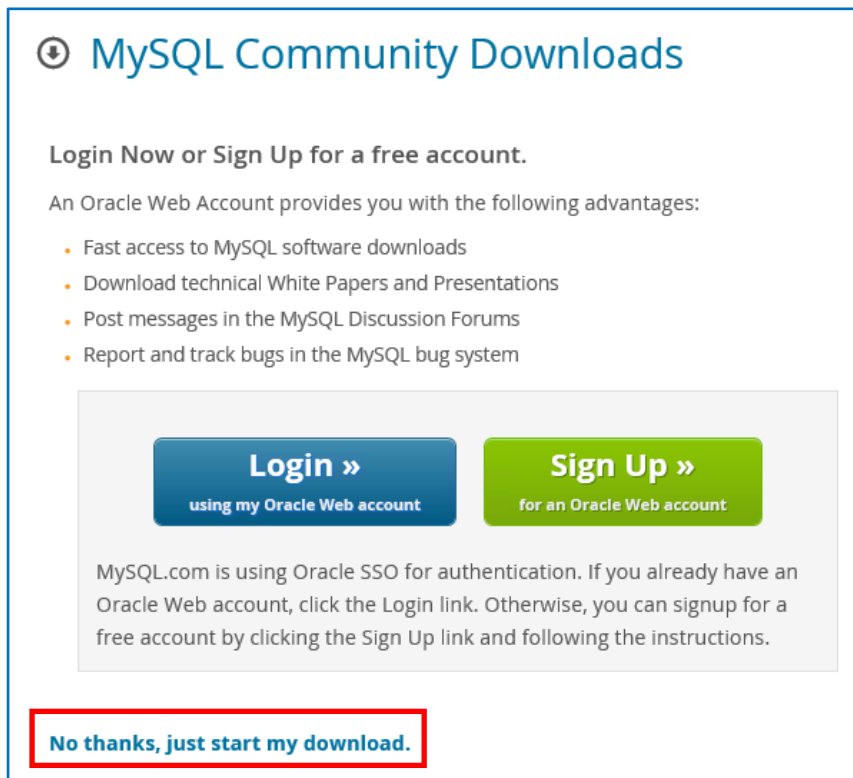
웹 페이지에서 **MySQL Workbench** 링크를 클릭한다.



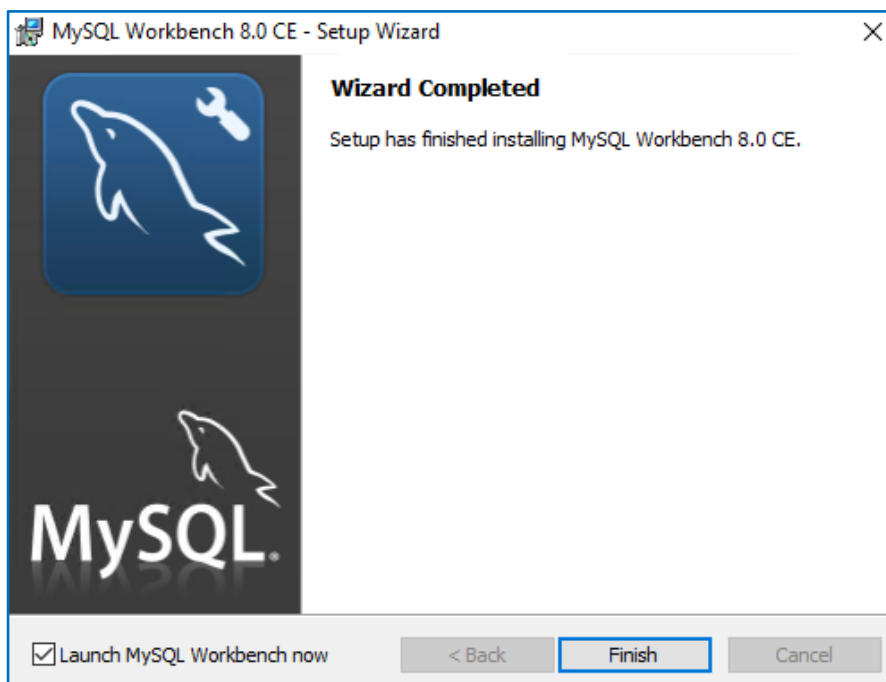
7. **[MySQL Workbench 8.0.29]** 페이지에서 **Windows 8(x86, 64-bit), MSI Installer**의 **[Download]** 버튼을 클릭하여 프로그램을 다운로드 한다.



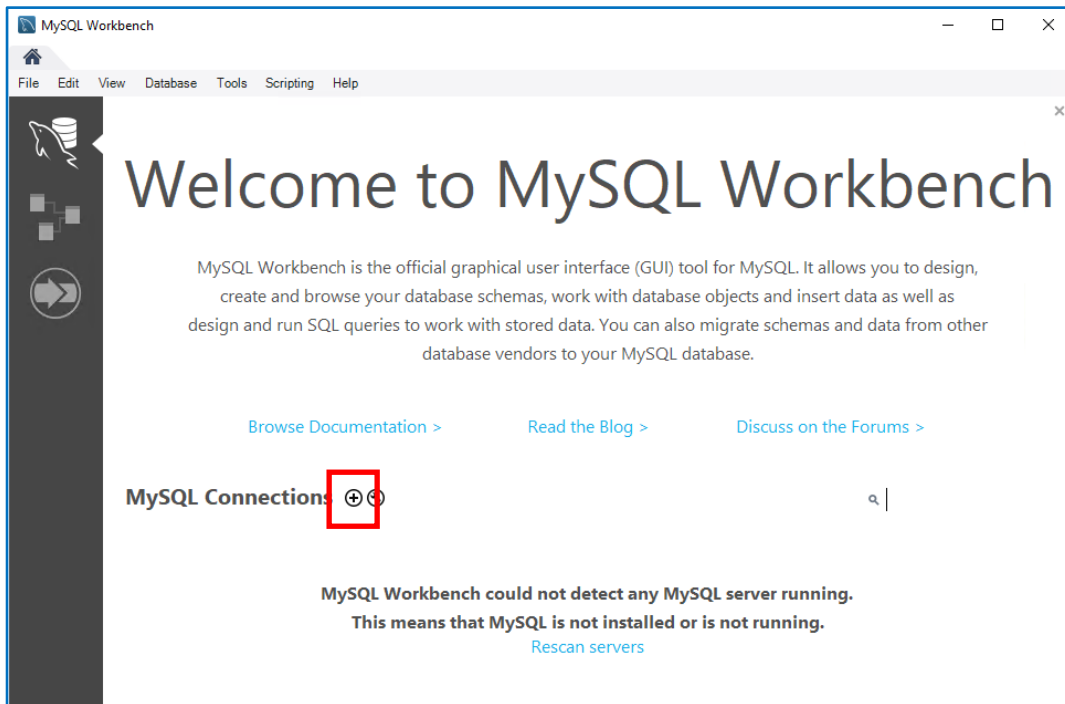
8. 아래 그림과 같은 페이지에서 **[No thanks, just start my download.]** 링크를 클릭하여 다운로드를 계속한다.



9. 다운로드 받은 **MySQL Workbench 8.0 CE** 프로그램을 설치한다.

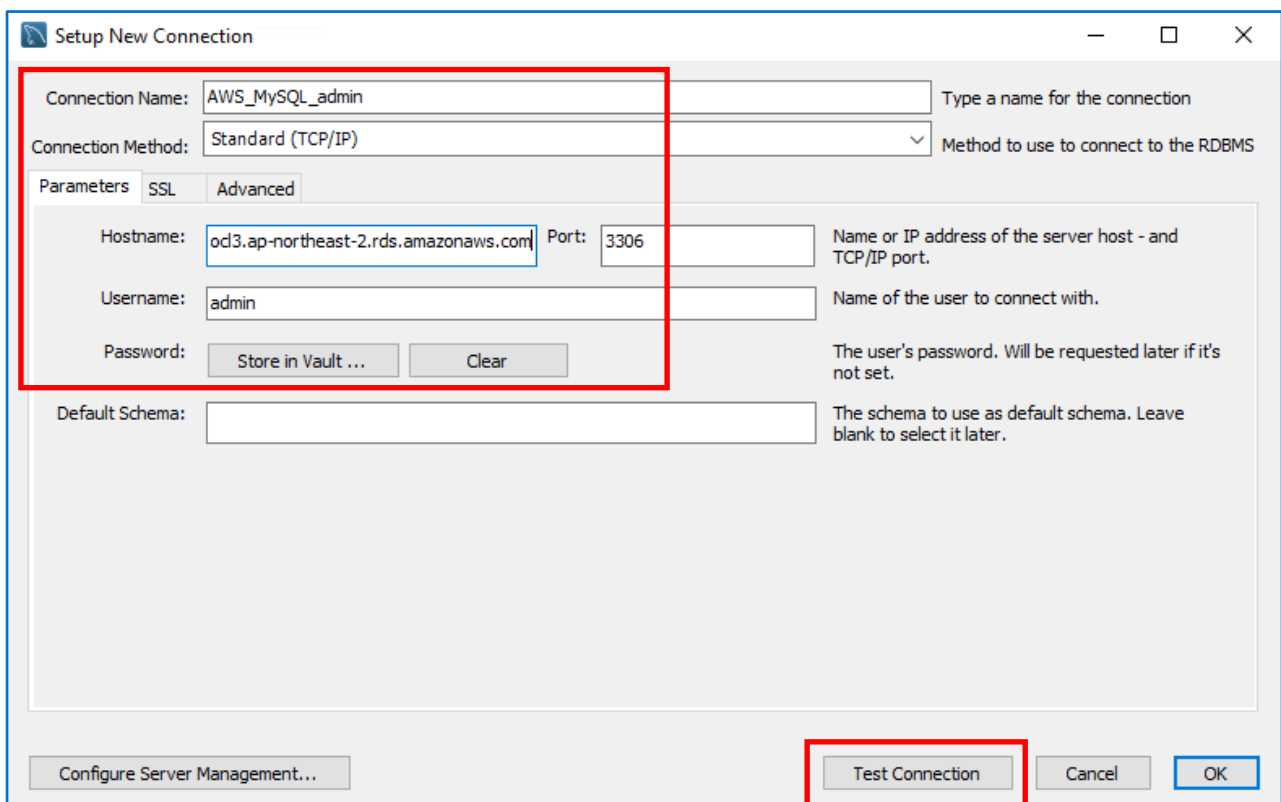


10. 설치를 모두 마치고 프로그램을 실행하면 다음 그림과 같다. 여기서 **MySQL Connections** 옆 + 아이콘을 클릭한다.

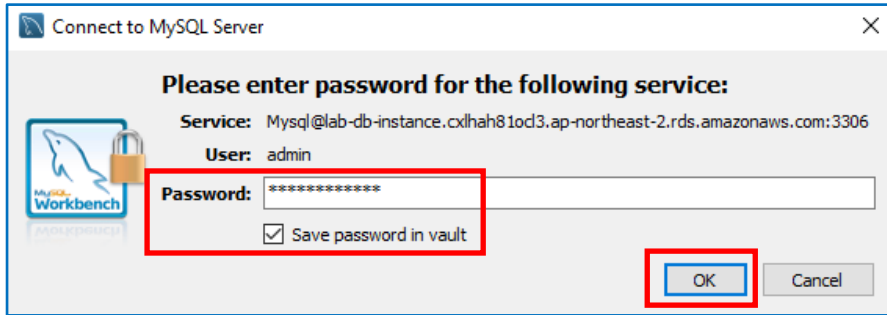


11. [Setup New Connection] 창에서 다음과 같이 입력하고, [Test Connection] 버튼을 클릭한다.

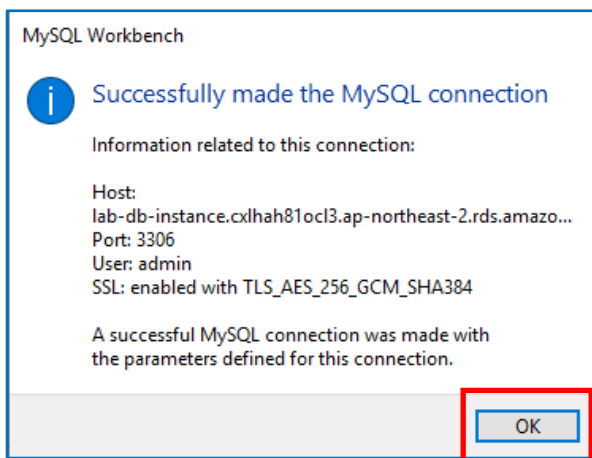
- A. [Connection Name] : AWS\_MySQL\_admin
- B. [Hostname] : MySQL 인스턴스의 엔드포인트
- C. [Username] : admin



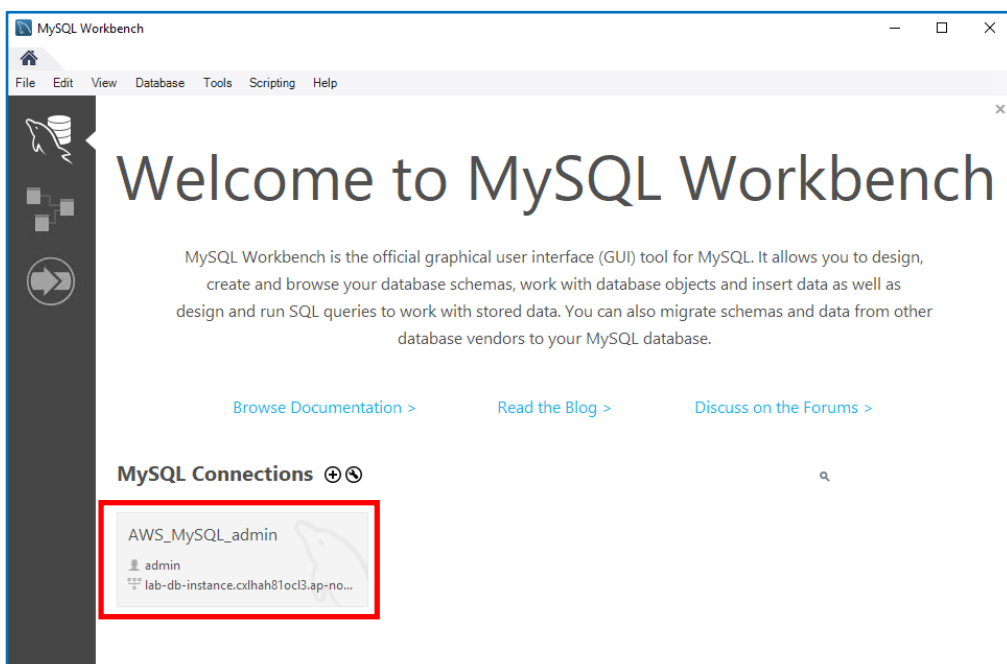
12. [Connect to MySQL Server] 창에서 비밀번호 **suwonmymysql**을 입력하고 체크박스에 체크한 후, [OK]를 클릭한다.



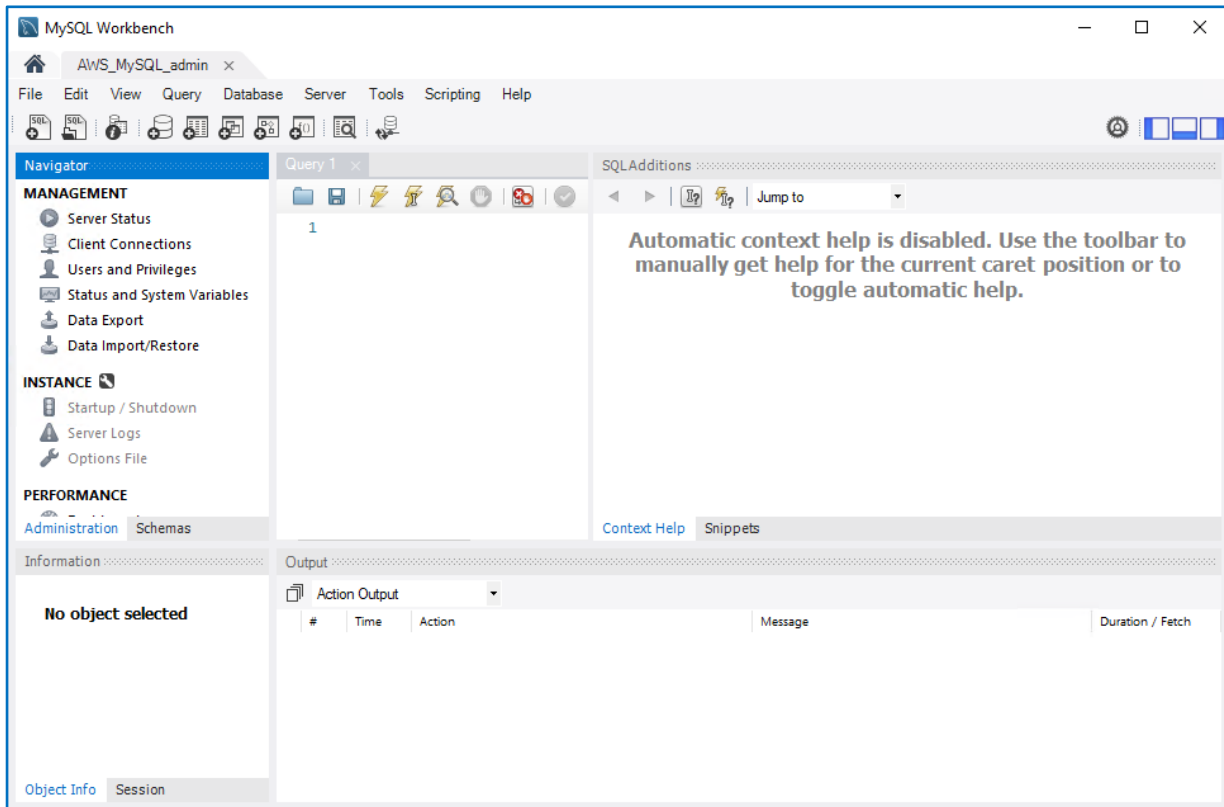
13. **Windows-ec2** 인스턴스가 **MySQL 데이터베이스 인스턴스**에 성공적으로 접속됨을 확인할 수 있다. [OK] 버튼을 클릭하여 창을 닫는다.



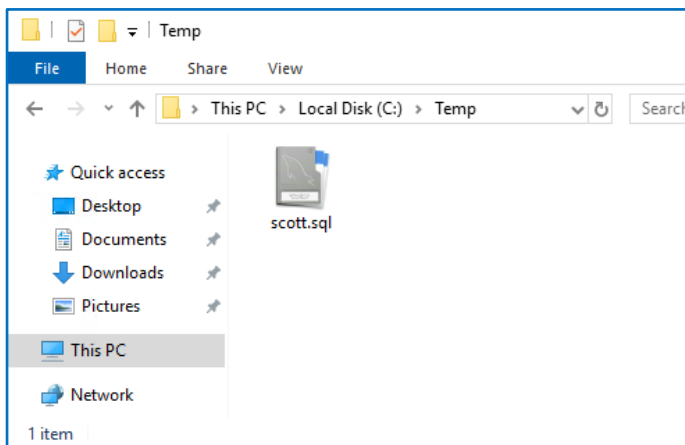
14. 이제 **MySQL 데이터베이스 인스턴스**와 연결할 때는 커넥션 카드를 클릭하면 된다.



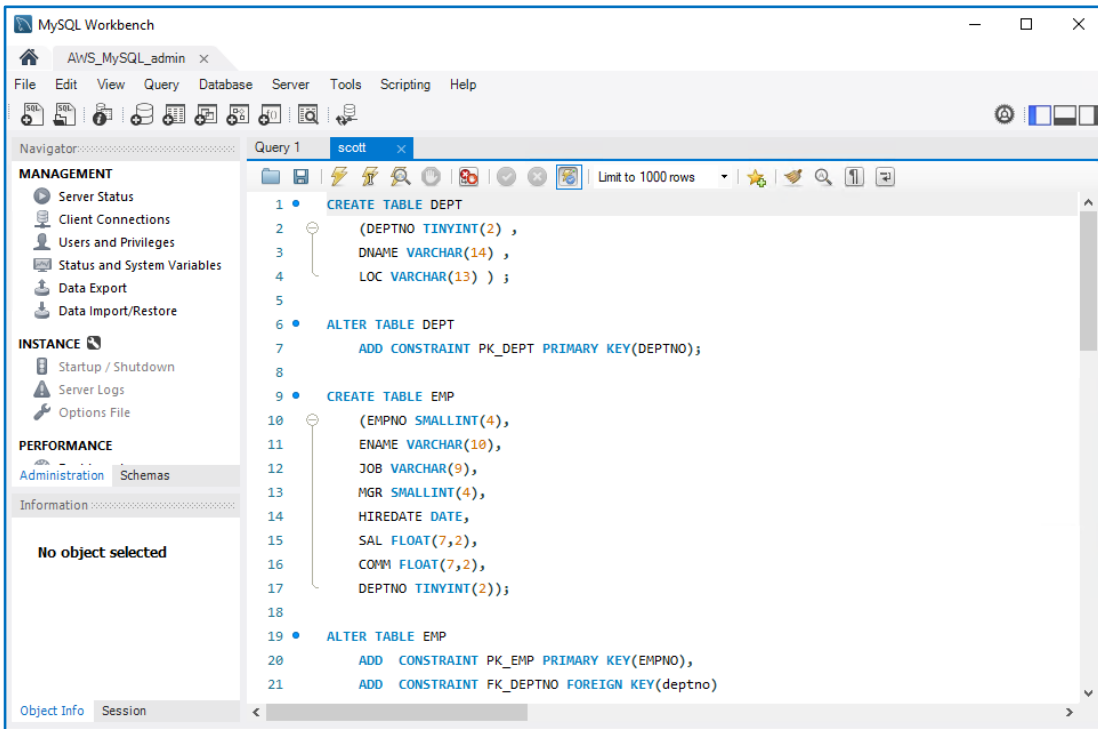
15. 사각 카드를 클릭하여 **MySQL 데이터베이스 인스턴스**와 연결한다.



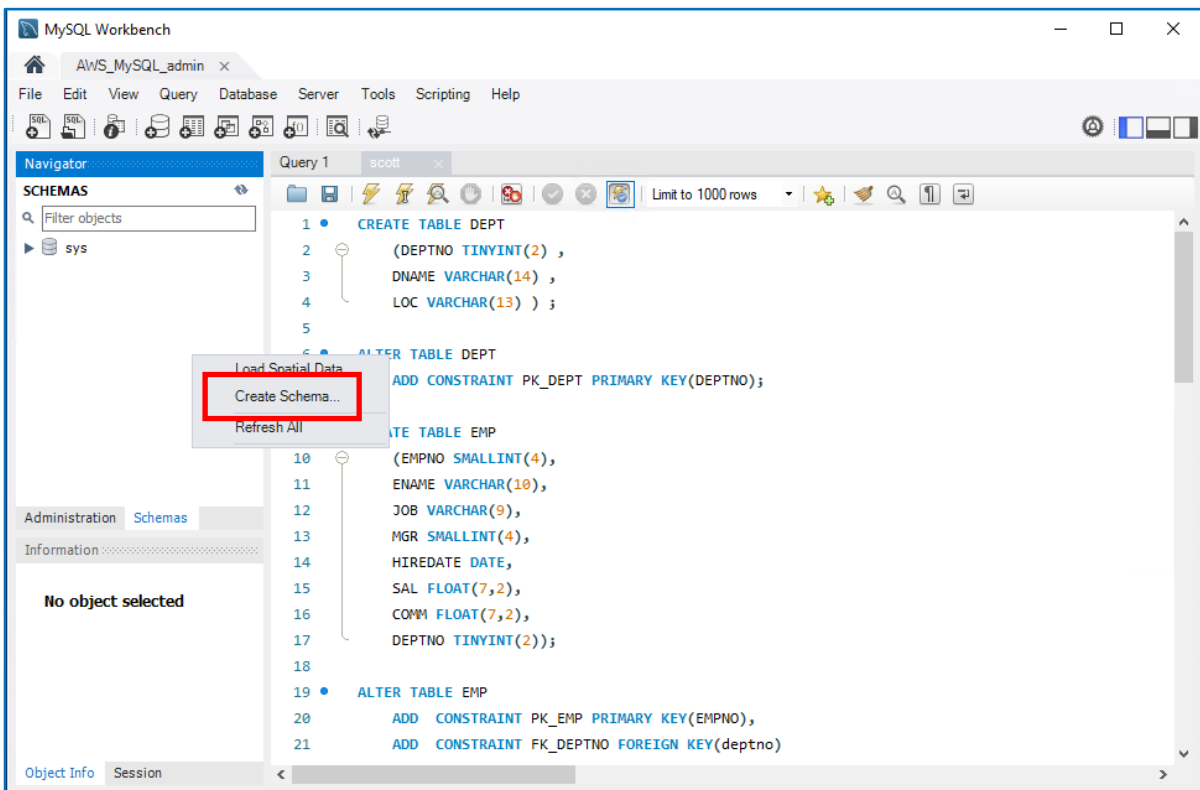
16. 샘플 스크립트 **scott.sql**을 다운로드하여 **C:/Temp**에 저장하였다.



17. **Scott.sql** 파일을 오픈한다.

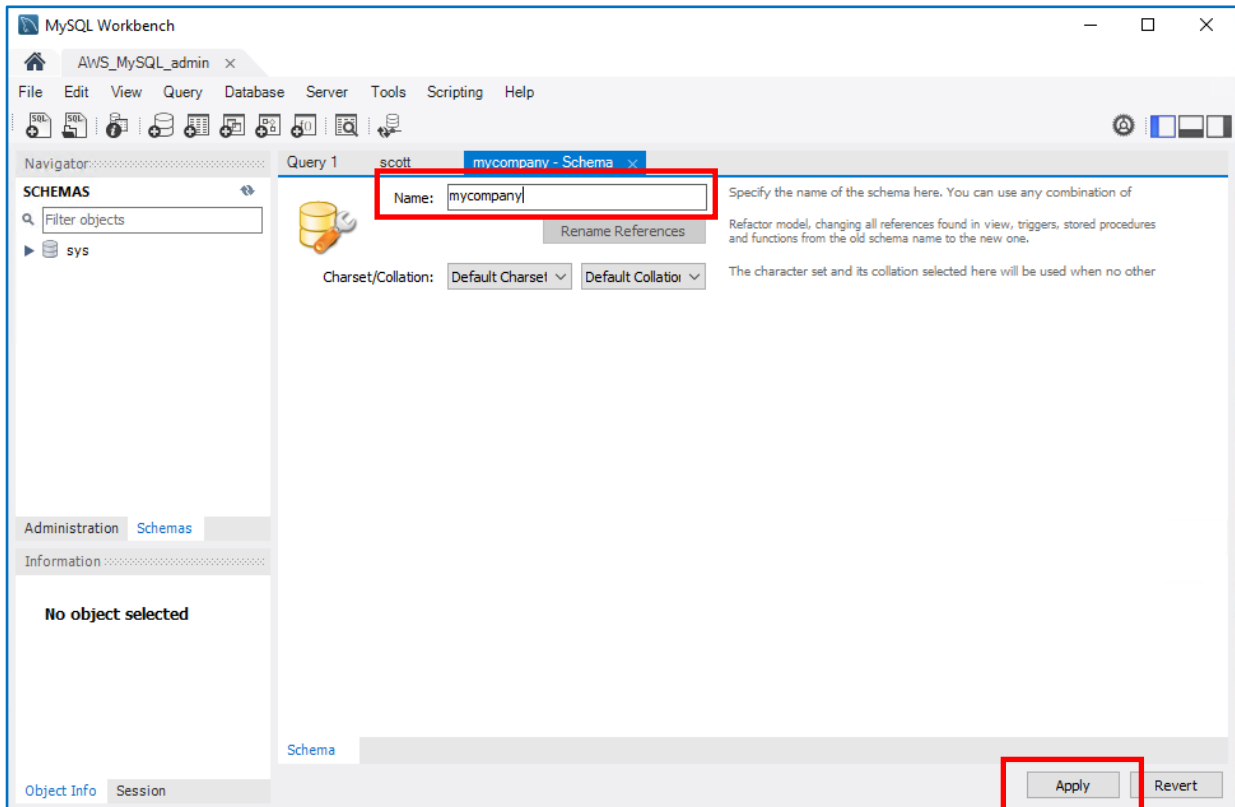


18. **Scott.sql** 스크립트를 실행하기 전, 왼쪽 **SCHEMAS**에서 마우스 오른쪽 버튼을 클릭하여 **[Create Schema]** 메뉴를 클릭한다.

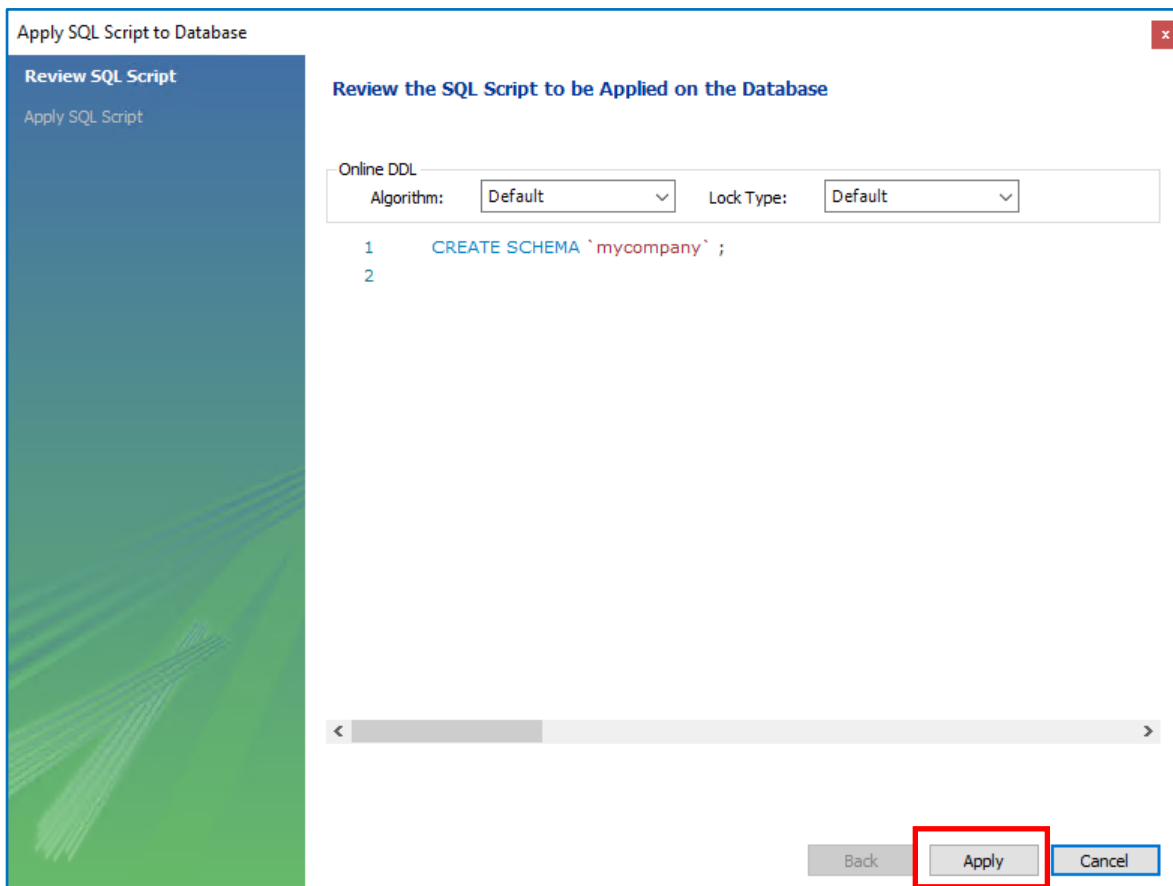




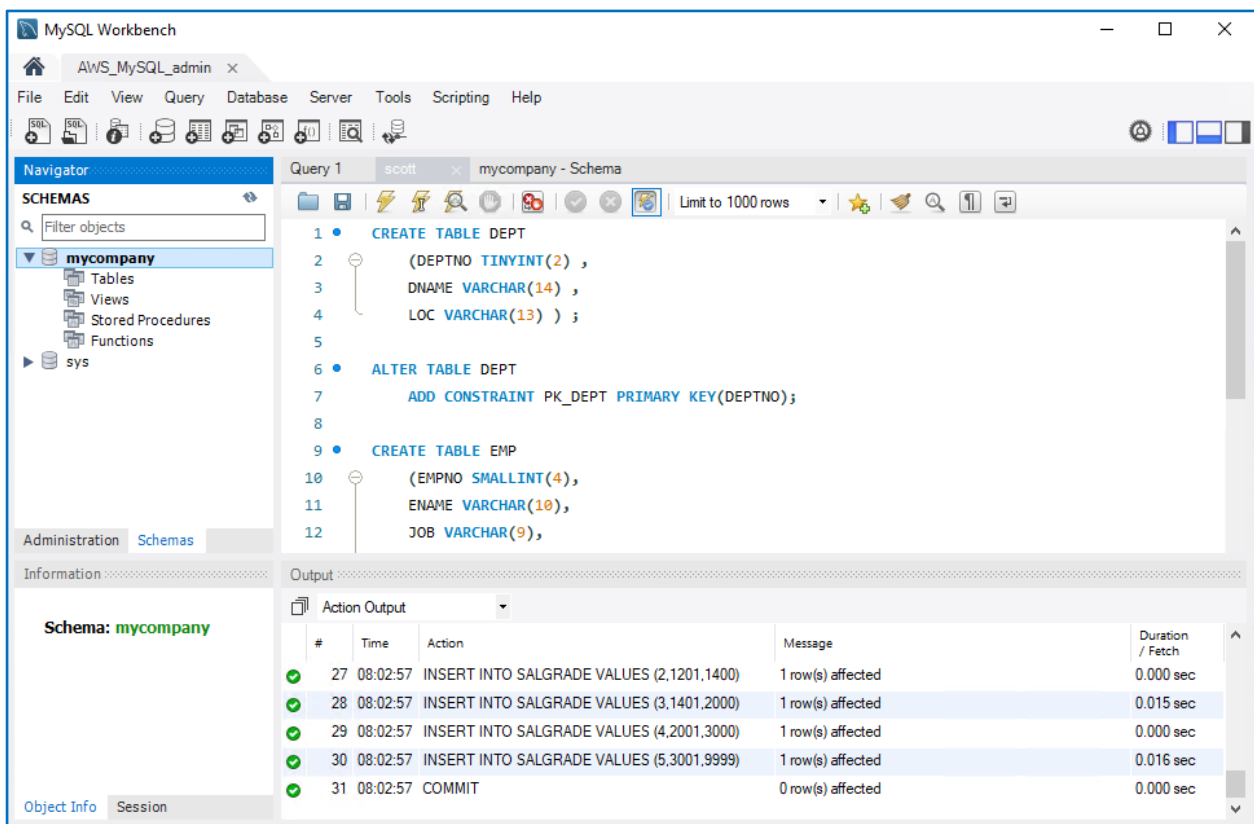
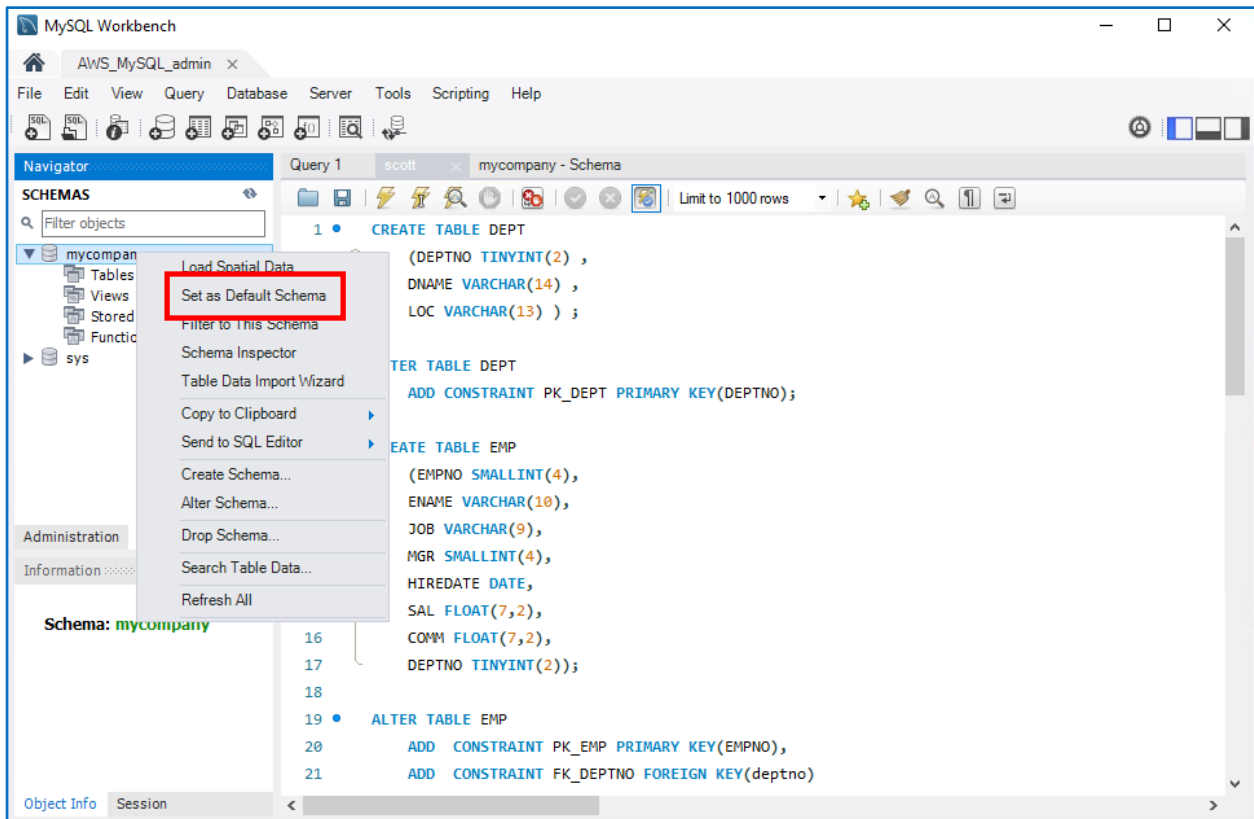
19. [Name]에 **mycompany**를 입력한다. [Apply]를 클릭한다.

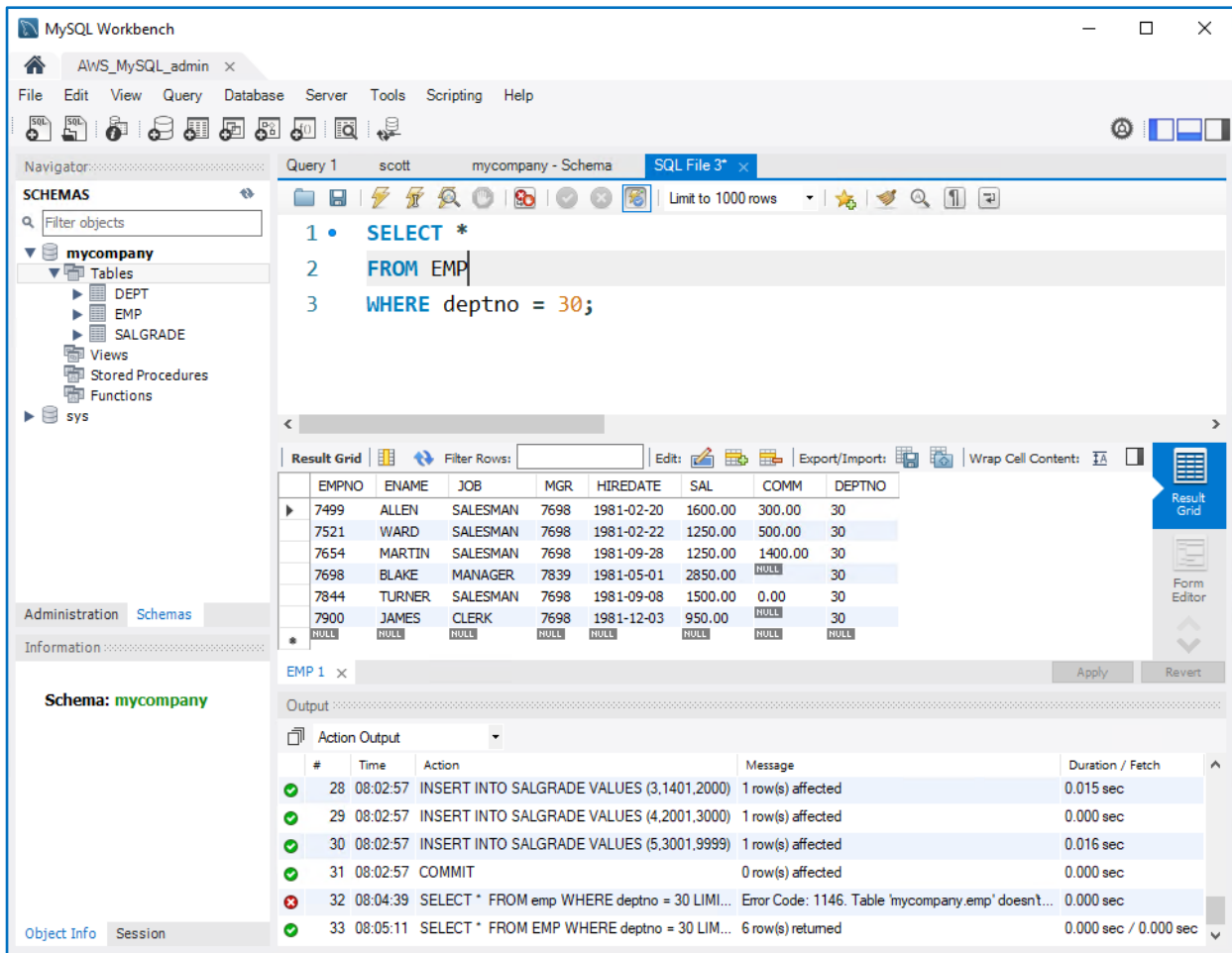


20. 기본값 그대로 사용하기로 하고, [Apply] 버튼을 클릭한다.

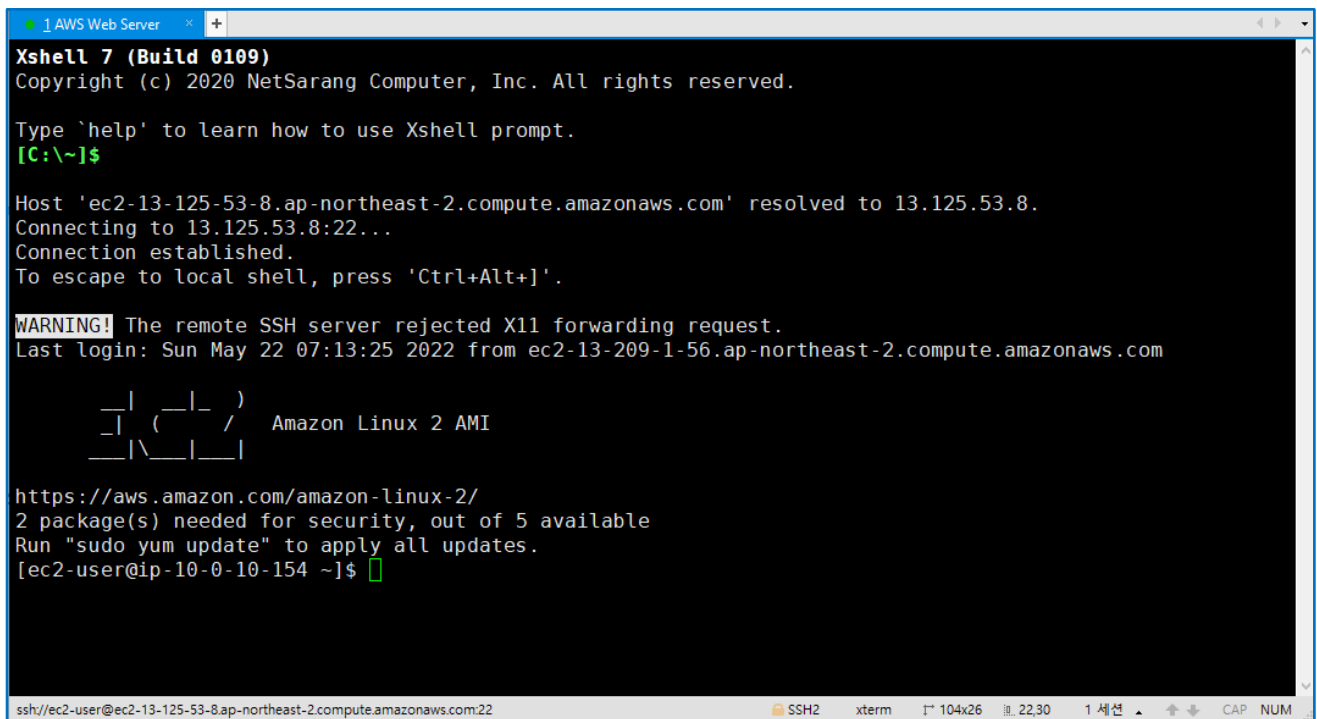


21. **mycompany**의 마우스 오른쪽 버튼을 클릭하여 **[Set as Default Schema]**로 선택하고, **scott.sql**을 실행한다.  
정상적으로 잘 실행되는지 확인한다.



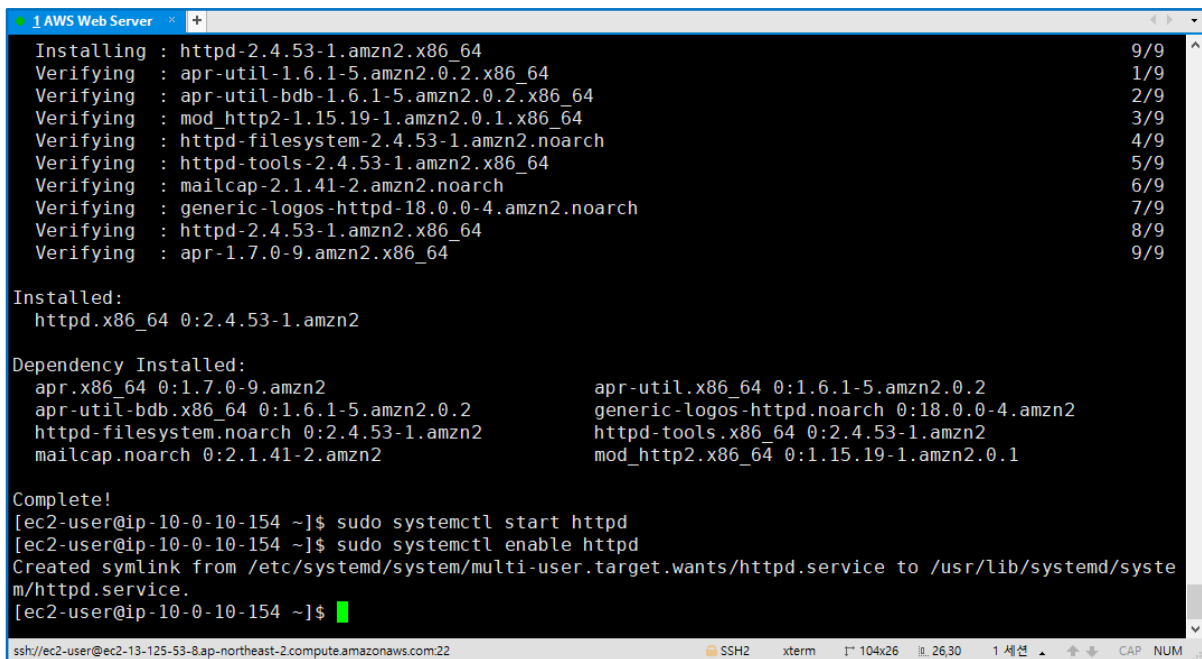


22. WebServer EC2 인스턴스에 연결한다.



23. **WebServer EC2** 인스턴스는 **PHP**와 함께 **Apache 웹 서버**를 설치할 것이다. **EC2 인스턴스** 연결이 되면 다음의 명령 순서대로 진행하여 **PHP**가 포함된 **Apache 웹 서버**를 설치한다.

- A. `$ sudo yum update -y`
- B. `$ sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2`
- C. `$ cat /etc/system-release`
- D. `$ sudo yum install -y httpd`
- E. `$ sudo systemctl start httpd`
- F. `$ sudo systemctl enable httpd`



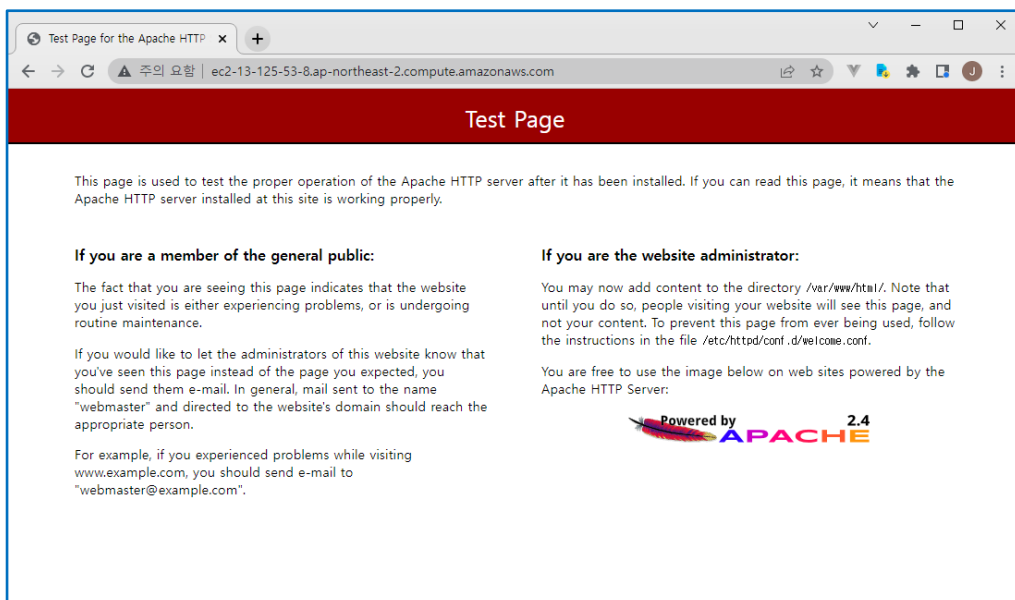
```
1 AWS Web Server
Installing : httpd-2.4.53-1.amzn2.x86_64 9/9
Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 3/9
Verifying : httpd-filesystem-2.4.53-1.amzn2.noarch 4/9
Verifying : httpd-tools-2.4.53-1.amzn2.x86_64 5/9
Verifying : mailcap-2.1.41-2.amzn2.noarch 6/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 7/9
Verifying : httpd-2.4.53-1.amzn2.x86_64 8/9
Verifying : apr-1.7.0-9.amzn2.x86_64 9/9

Installed:
  httpd.x86_64 0:2.4.53-1.amzn2

Dependency Installed:
  apr.x86_64 0:1.7.0-9.amzn2          apr-util.x86_64 0:1.6.1-5.amzn2.0.2
  apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2  generic-logos-httpd.noarch 0:18.0.0-4.amzn2
  httpd-filesystem.noarch 0:2.4.53-1.amzn2  httpd-tools.x86_64 0:2.4.53-1.amzn2
  mailcap.noarch 0:2.1.41-2.amzn2          mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[ec2-user@ip-10-0-10-154 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-10-154 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-10-0-10-154 ~]$
```

24. 웹 서버가 제대로 설치되고 시작되었는지 직접 웹 브라우저를 열고 **WebServer EC2 인스턴스**의 [퍼블릭 **Domain Name System(DNS)**] 이름을 입력한다.



25. 다음은 Apache 웹 서버에 대한 파일 권한을 설정한다.

A. `$ sudo usermod -a -G apache ec2-user`

B. `$ exit`

다시 로그인한 후, 계속해서 명령을 수행한다.

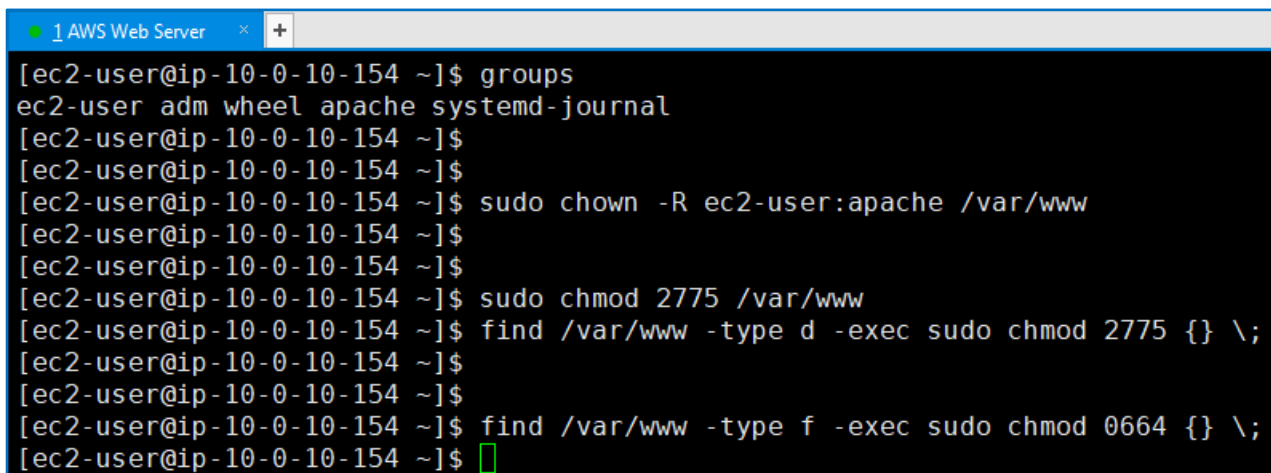
C. `$ groups`

D. `$ sudo chown -R ec2-user:apache /var/www`

E. `$ sudo chmod 2775 /var/www`

F. `$ find /var/www -type d -exec sudo chmod 2775 {} \;`

G. `$ find /var/www -type f -exec sudo chmod 0664 {} \;`



The screenshot shows a terminal window titled "1 AWS Web Server". The user is logged in as "ec2-user" on the IP address "10-0-10-154". The terminal displays the following commands and their outputs:

```
[ec2-user@ip-10-0-10-154 ~]$ groups
ec2-user adm wheel apache systemd-journal
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$ sudo chmod 2775 /var/www
[ec2-user@ip-10-0-10-154 ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$
[ec2-user@ip-10-0-10-154 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-10-0-10-154 ~]$
```

26. 다음은 DB 인스턴스에 연결되는 Apache 웹 서버에 콘텐츠를 추가한다. EC2 인스턴스에 계속 연결되어 있을 때 디렉터리를 `/var/www`로 변경하고 `inc`라는 새로운 하위 디렉터리를 생성한다. `inc`라는 `dbinfo.inc` 디렉터리에서 새 파일을 생성한 다음 `nano` 또는 선택한 편집기를 호출하여 파일을 편집한다. 다음 내용을 `dbinfo.inc` 파일에 추가한다. 여기서 `db_instance_endpoint`는 포트가 없는 DB 인스턴스 엔드포인트이고, `master password`는 DB 인스턴스의 마스터 암호이다.

- A. `$ cd /var/www`
- B. `$ mkdir inc`
- C. `$ cd inc`
- D. `$ nano dbinfo.inc`

```
GNU nano 2.9.8                                dbinfo.inc                                Modified
<?php
define('DB_SERVER', 'lab-db-instance.cxlhah81ocl3.ap-northeast-2.rds.amazonaws.com');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'suwonmymysql');
define('DB_DATABASE', 'mycompany');
?>
```

27. `dbinfo.inc` 파일을 저장하고 닫는다.

```
[ec2-user@ip-10-0-10-154 inc]$ cat dbinfo.inc
<?php
define('DB_SERVER', 'lab-db-instance.cxlhah81ocl3.ap-northeast-2.rds.amazonaws.com');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'suwonmymysql');
define('DB_DATABASE', 'mycompany');
?>
[ec2-user@ip-10-0-10-154 inc]$
```

28. 다음과 같이 html 디렉토리에 `SamplePage.php` 파일을 다운로드한다.

- A. `$ cd /var/www/html`
- B. `$ wget https://github.com/swacademy/suwonuniv\_cloud\_architect/blob/main/8%EC%A3%BC%EC%B0%A8/SamplePage.php`

```
1 AWS Web Server x +
$query = "CREATE TABLE EMPLOYEES (
    ID int(11) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    NAME VARCHAR(45),
    ADDRESS VARCHAR(90)
)";

if(!mysqli_query($connection, $query)) echo("<p>Error creating table.</p>");
}
}

/* Check for the existence of a table. */
function TableExists($tableName, $connection, $dbName) {
    $t = mysqli_real_escape_string($connection, $tableName);
    $d = mysqli_real_escape_string($connection, $dbName);

    $checktable = mysqli_query($connection,
        "SELECT TABLE_NAME FROM information_schema.TABLES WHERE TABLE_NAME = '$t' AND TABLE_SCHEMA = '$d'");
};

if(mysqli_num_rows($checktable) > 0) return true;

return false;
}
?>

[ec2-user@ip-10-0-10-154 html]$
```

29. 웹 브라우저를 열고 **http://{EC2 instance endpoint}/SamplePage.php**를 검색하여 웹 서버에서 DB 인스턴스에 제대로 연결되는지 확인한다.

Sample page

NAME ADDRESS

ID	NAME	ADDRESS
----	------	---------

30. 다음과 같이 값을 넣고 저장하여 실제로 데이터베이스에 제대로 저장되는지 확인한다.

Sample page

NAME ADDRESS

ID	NAME	ADDRESS
1	Henry	경기도 화성시 봉담읍 수원대학교
2	Smith	서울시 강남구 역삼동 메가존빌딩
3	Sujan	경기도 수원시 수원역

