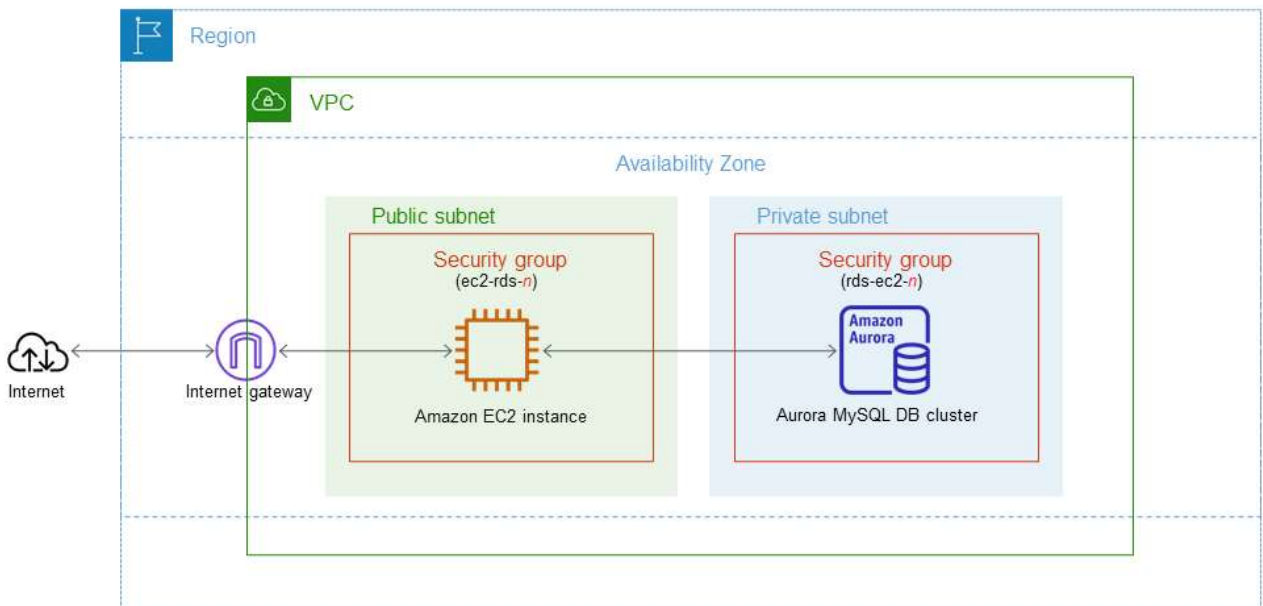


# Aurora MySQL DB 클러스터 생성 및 연결

이 Tutorial에서는 EC2 Instance 및 Aurora MySQL DB Cluster를 생성한다. Tutorial에서는 표준 MySQL Client를 사용하여 EC2 Instance에서 DB Cluster에 액세스하는 방법을 보여준다. 이 Tutorial에서는 모범 사례를 따라 Virtual Private Cloud(VPC)에서 Private DB Cluster를 생성한다. 대부분의 경우 EC2 Instance와 같이 동일한 VPC에 있는 다른 Resource는 DB Instance에 액세스할 수 있지만 VPC 외부의 Resource는 DB Cluster에 액세스할 수 없다.

자습서를 완료하면 VPC의 각 AZ에 Public Subnet과 Private Subnet이 있을 것이다. 한 AZ에서 EC2 Instance는 Public Subnet에 있고 DB Instance는 Private Subnet에 있다.

다음 Diagram은 이 자습서를 완료했을 때 구성을 보여 준다.



이 Tutorial에서는 AWS Management Console에서 **손쉬운 생성**을 사용하여 Aurora MySQL을 실행하는 DB Cluster를 생성한다. **손쉬운 생성**에서는 DB Engine Type, DB Instance 크기, DB Cluster 식별자만 지정한다. [**손쉬운 생성(Easy create)**]은 다른 구성 옵션에서도 기본 설정을 사용한다. **손쉬운 생성**으로 만든 DB Cluster는 Private Cluster이다.

**손쉬운 생성** 대신 **표준 생성**을 사용하는 경우에는 DB Cluster를 생성할 때 가용성, 보안, 백업 및 유지 관리 등에 대한 옵션을 포함하여 더 많은 구성 옵션을 지정할 수 있다. Public DB Cluster를 만들려면 **표준 생성**을 사용해야 한다. [**표준 생성(Standard Create)**]을 사용하여 DB Cluster를 생성하는 방법에 대한 자세한 내용은 [Amazon Aurora DB 클러스터 생성](#) 섹션을 참조한다.

## 주제

- [필수 조건](#)

- [1 단계: EC2 인스턴스 생성](#)
  - [2 단계: Aurora MySQL DB 클러스터 생성](#)
  - [3 단계: Aurora MySQL DB 클러스터에 연결](#)
  - [4 단계: EC2 인스턴스 및 DB 클러스터 삭제](#)
- 

## 필수 조건

시작하기 전에 다음 섹션에서 다음 단계를 완료한다.

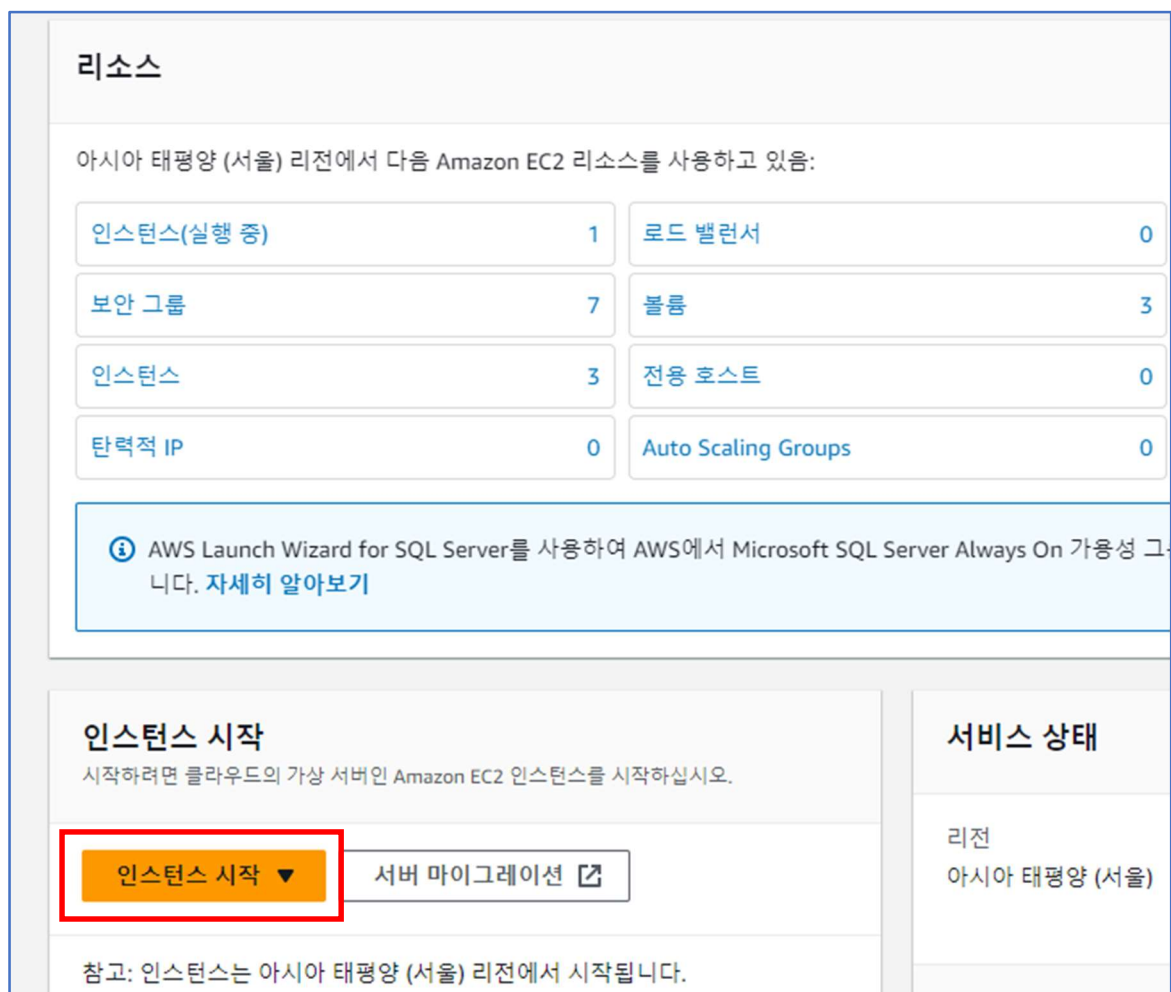
- [AWS 계정에 가입](#)
- [관리 사용자 생성](#)

# 1 단계: EC2 인스턴스 생성

Database 에 연결하는 데 사용할 Amazon EC2 인스턴스를 생성한다.

## EC2 인스턴스를 생성하려면

1. AWS Management Console 에 로그인하고 <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. AWS Management Console 의 오른쪽 상단에서 EC2 인스턴스를 생성하려는 AWS Region 을 선택합니다.
3. **EC2 대시보드**를 선택한 다음, **인스턴스 시작**을 선택한다.



인스턴스 시작 페이지가 열린다.

4. 인스턴스 시작 페이지에서 다음 설정을 선택한다.
  - a. **Name and tags**(이름 및 태그) 아래의 **Name**(이름)에 **ec2-database-connect** 을 입력한다.
  - b. **Application and OS Images (Amazon Machine Image)**(애플리케이션 및 OS 이미지(Amazon Machine Image))에서 **Ubuntu ubuntu** 를 선택한 다음 **Ubuntu**

Server 22.04 LTS(HVM), SSD Volume Type 를 선택한다. 다른 선택 항목에 대해서는 기본값을 그대로 유지한다.

**Quick Start**

Amazon Linux  
aws


macOS  
Mac

**Ubuntu  
ubuntu**

Windows  
Microsoft

Red Hat  
Red Hat

SUSE Li  
SUS

  
더 많은 AMI 찾아보기  
AWS, Marketplace 및 커뮤니티의 AMI 포함

Amazon Machine Image(AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
ami-0c9c942bd7bf113a2 (64비트(x86)) / ami-00fdfe418c69b624a (64비트(Arm))  
가상화: hvm   ENA 활성화됨: true   루트 디바이스 유형: ebs

프리 티어 사용 가능 ▼

설명  
Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-05-16

아키텍처      AMI ID  
64비트(x86) ▼      ami-0c9c942bd7bf113a2      **확인된 공급 업체**

- c. 인스턴스 유형에서 **t2.micro** 를 선택한다.
- d. 키 페어(로그인)에서 기존 키 페어를 사용할 **키 페어 이름**을 선택한다. Amazon EC2 Instance 에 대한 새 Key Pair 를 생성하려면 **새 키 페어 생성**을 선택한 다음 **키 페어 생성** 창을 사용하여 생성한다. 생성할 Key Pair 이름은 Instance 이름에 key 를 붙인다(예: ec2-database-connect-key.pem)
- e. 네트워크 설정은 다음 이미지를 참고한다.

**▼ 네트워크 설정 정보**

VPC - 필수 정보

vpc-06cc1e03aaa8fd14e (Default-VPC) (기본값) ▼  
172.31.0.0/16

서브넷 정보

subnet-0539a07bde3b1c1af  
VPC: vpc-06cc1e03aaa8fd14e   소유자: 789534828835   가용 영역: ap-northeast-2a ▼  
사용 가능한 IP 주소: 4090   CIDR: 172.31.0.0/20

퍼블릭 IP 자동 할당 정보

활성화 ▼

f. 보안 그룹은 새 보안그룹을 생성하고 다음 이미지를 참고한다.

**방화벽(보안 그룹) 정보**  
보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 특정 트래픽이 인스턴스에 도달하도록 허용하는 규칙을 추가합니다.

☒ 보안 그룹 생성 ☐ 기존 보안 그룹 선택

보안 그룹 이름 - 필수  
  
이 보안 그룹은 모든 네트워크 인터페이스에 추가됩니다. 보안 그룹을 만든 후에는 이름을 편집할 수 없습니다. 최대 길이는 255자입니다. 유효한 문자는 a~z, A~Z, 0~9, 공백 및 . \_ - / () # , @ [] + = & ; {} ! \$ \* 입니다.

설명 - 필수 정보

Inbound Security Group Rules  
▼ 보안 그룹 규칙 1 (TCP, 22, 0.0.0.0/0) 제거

| 유형 정보                              | 프로토콜 정보  | 포트 범위 정보                                      |
|------------------------------------|--|---|
| <input type="text" value="ssh"/>   | <input type="text" value="TCP"/>   | <input type="text" value="22"/>               |
| 소스 유형 정보                           | 원본 정보  | 설명 - optional 정보                              |
| <input type="text" value="위치 무관"/> | <input type="text" value="CIDR, 접두사 목록 또는 보안 그룹"/><br><input type="text" value="0.0.0.0/0"/> <input type="text" value="::/0"/> | <input type="text" value="예: 관리자 데스크톱용 SSH"/> |

g. 요약 패널에서 EC2 인스턴스 구성 요약을 검토하고 준비가 되면 **인스턴스 시작**을 선택한다.

## ▼ Summary

인스턴스 개수 [정보](#)

1

[소프트웨어 이미지\(AMI\)](#)

Canonical, Ubuntu, 22.04 LTS, ... [더 보기](#)  
ami-0c9c942bd7bf113a2

[가상 서버 유형\(인스턴스 유형\)](#)



t2.micro

[방화벽\(보안 그룹\)](#)

새 보안 그룹

[스토리지\(볼륨\)](#)

1개의 볼륨 - 8GiB

 **프리 티어:** 첫 해에는 월별 프리 티어 AMI에 대한 t2.micro(또는 t2.micro를 사용할 수 없는 리전의 t3.micro) 인스턴스 사용량 750시간, EBS 스토리지 30GiB, IO 2백만 개, 스냅샷 1GB, 인터넷 대역폭 100GB가 포함됩니다. 

[취소](#)

**인스턴스 시작**

[명령 검토](#)

5. EC2 Instance 식별자를 선택하여 EC2 Instance 목록을 열고 EC2 Instance 를 선택한다.
6. **세부 정보** 탭에서 SSH 를 사용하여 연결할 때 필요한 다음 값을 기록한다.
  - a. **인스턴스 요약**에서 **퍼블릭 IPv4 DNS** 의 값을 기록한다.

|   |   |   |
|---|---|---|
| EC2 > 인스턴스 > i-0a25d3b7bc7504202  |   |   |
| <b>i-0a25d3b7bc7504202 (ec2-database-connect)에 대한 인스턴스 요약 정보</b><br>less than a minute 전에 업데이트됨 |   |   |
| <div> <div>🔄</div> <div>연결</div> <div>인스턴스 상태 ▼</div> <div>작업 ▼</div> </div>                    |   |   |
| 인스턴스 ID<br>i-0a25d3b7bc7504202 (ec2-database-connect)   | 퍼블릭 IPv4 주소<br>3.36.77.60   <a href="#">개방 주소법</a>                          | 프라이빗 IPv4 주소<br>172.31.12.15  |
| IPv6 주소<br>-  | 인스턴스 상태<br>실행 중   | <div>         퍼블릭 IPv4 DNS<br/>         ec2-3-36-77-60.ap-northeast-2.compute.amazonaws.com   <a href="#">개방 주소법</a> </div> |
| 호스트 이름 유형<br>IP 이름: ip-172-31-12-15.ap-northeast-2.compute.internal                             | 프라이빗 IP DNS 이름(IPv4만 해당)<br>ip-172-31-12-15.ap-northeast-2.compute.internal |   |

b. 인스턴스 세부 정보에서 키 페어 이름의 값을 기록한다.

|                    |  |   |
|--------------------|--|---|
| 중지 방지<br>비활성       | 시작 시간<br>Wed Jul 19 2023 00:31:13 GMT+0900 (한국 표준시) (1 minute)                 | AMI 위치<br>amazon/ubuntu/images/hvm-ssd/ubuntu-jammy-22.04-amd64-server-20230516 |
| 인스턴스 자동 복구<br>기본값  | 수명 주기<br>정상  | 중지-최대 절전 모드 동작<br>비활성   |
| AMI 시작 인덱스<br>0    | <div>         시작 시 할당된 키 페어<br/>         ec2-database-connect-key       </div> | 상태 전환 이유<br>-   |
| 크레딧 사양<br>standard | 커널 ID<br>-   | 상태 전환 메시지<br>-  |

7. 계속하기 전에 EC2 인스턴스의 인스턴스 상태가 실행 중이 될 때까지 기다린다.

## 2 단계: Aurora MySQL DB 클러스터 생성

이 예시에서는 손쉬운 생성을 사용하여 db.r6g.large DB Instance Class 로 Aurora MySQL DB Cluster 를 생성한다.

손쉬운 생성을 사용하여 Aurora MySQL DB 클러스터를 생성하려면

1. AWS Management Console 에 로그인한 후 <https://console.aws.amazon.com/rds/>에서 Amazon RDS 콘솔을 엽니다.
2. Amazon RDS 콘솔의 오른쪽 상단에서 DB Cluster 를 생성하려는 AWS Region 을 선택한다.
3. 탐색 창에서 데이터베이스를 선택한다.
4. [데이터베이스 생성(Create database)]을 선택하고 [손쉬운 생성(Easy Create)]이 선택되어 있는지 확인한다.

## 데이터베이스 생성 방식 선택 정보

### ☐ 표준 생성

가용성, 보안, 백업 및 유지 관리에 대한 옵션을 포함하여 모든 구성 옵션을 설정합니다.

### ☒ 손쉬운 생성


권장 모범 사례 구성을 사용합니다. 일부 구성 옵션은 데이터베이스를 생성한 후 변경할 수 있습니다.


5. 구성의 엔진 유형에서 **Aurora(MySQL 호환)**를 선택한다.


6. DB instance size(DB 인스턴스 크기)에서 **개발/테스트**를 선택한다.


### 구성


#### 엔진 유형 정보


☒ Aurora (MySQL Compatible)  



☐ Aurora (PostgreSQL Compatible)  


☐ MySQL  


☐ MariaDB  


☐ PostgreSQL  


☐ Oracle  


☐ Microsoft SQL Server  


#### DB 인스턴스 크기

☐ 프로덕션  
db.r6g.2xlarge  
8 vCPUs  
64 GiB RAM

☒ 개발/테스트  
db.r6g.large  
2 vCPUs  
16 GiB RAM

7. DB 클러스터 식별자에서 **database-test1** 을 입력한다.

8. **마스터 사용자 이름**에 마스터 사용자의 이름을 입력하거나 기본 이름을 그대로 유지한다.

9. DB 클러스터에 자동 생성된 마스터 암호를 사용하려면 **암호 자동 생성**을 선택한다.  
마스터 암호를 입력하려면 **암호 자동 생성** 선택을 해제한 다음, **마스터 암호**와 **암호 확인**에 동일한 암호를 입력한다.



### DB 클러스터 식별자

DB 클러스터 이름을 입력합니다. 이름은 현재 AWS 리전에서 AWS 계정이 소유하는 모든 DB 클러스터에 대해 고유해야 합니다.

database-test1

DB 클러스터 식별자는 대소문자를 구분하지 않지만 모두 소문자로 저장됩니다(예: "mydbcluster"). 제약 조건: 1~60자의 영숫자 또는 하이픈으로 구성되어야 합니다. 첫 번째 문자는 글자여야 합니다. 하이픈은 연속으로 2개를 포함할 수 없습니다. 끝에 하이픈이 올 수 없습니다.

### 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).

### 마스터 암호 확인 정보

.....

10. 이전에 생성한 EC2 인스턴스와의 연결을 설정하려면 **EC2 연결 설정 - 선택 사항**을 엽니다.

**Connect to an EC2 compute resource**(EC2 컴퓨팅 리소스에 연결)를 선택한다. 이전에 생성한 EC2 인스턴스를 선택한다.

### ▼ EC2 연결 설정 - 선택 사항

데이터베이스를 생성한 후 EC2 인스턴스에 대한 연결을 설정할 수도 있습니다. 데이터베이스 목록 페이지 또는 데이터베이스 세부 정보 페이지로 이동하여 **Actions**(작업)를 선택한 다음 **Set up to EC2 connection**(EC2 연결 설정)을 선택합니다.

#### 컴퓨팅 리소스

이 데이터베이스의 컴퓨팅 리소스에 대한 연결을 설정할지를 선택합니다. 연결을 설정하면 컴퓨팅 리소스가 이 데이터베이스에 연결할 수 있도록 연결 설정이 자동으로 변경됩니다.

#### ☐ EC2 컴퓨팅 리소스에 연결 안 함

이 데이터베이스의 컴퓨팅 리소스에 대한 연결을 설정하지 않습니다. 나중에 컴퓨팅 리소스에 대한 연결을 수동으로 설정할 수 있습니다.

#### ☒ EC2 컴퓨팅 리소스에 연결

이 데이터베이스의 EC2 컴퓨팅 리소스에 대한 연결을 설정합니다.

#### EC2 인스턴스 정보

이 데이터베이스의 컴퓨팅 리소스로 추가할 EC2 인스턴스를 선택합니다. VPC 보안 그룹이 이 EC2 인스턴스에 추가됩니다. VPC 보안 그룹은 EC2 인스턴스가 데이터베이스에 액세스하도록 허용하는 인바운드 규칙과 함께 데이터베이스에 추가됩니다.

i-0a25d3b7bc7504202  
ec2-database-connect



## 11. 손쉬운 생성에 대한 기본 설정 보기를 오픈한다.

| ▼ 손쉬운 생성에 대한 기본 설정 보기  |                                     |                     |
|--|-------------------------------------|---------------------|
| 손쉬운 생성은 다음 구성을 기본값으로 설정하며, 이 중 일부는 나중에 변경할 수 있습니다. 지금 설정을 변경하려면 다음을 사용하세요. <a href="#">표준 생성</a> . |                                     |                     |
| 구성 ▼   | 값                                   | 데이터베이스 생성 후 편집 가능 ▲ |
| 암호화  | 활성화됨                                | 아니요                 |
| VPC  | Default VPC (vpc-06cc1e03aaa8fd14e) | 아니요                 |
| 옵션 그룹  | default:aurora-mysql-8-0            | 아니요                 |
| 서브넷 그룹   | rds-ec2-db-subnet-group-2           | 예                   |
| 자동 백업  | 활성화됨                                | 예                   |
| VPC 보안 그룹  | rds-ec2-1                           | 예                   |
| 퍼블릭 액세스 가능   | 아니요                                 | 예                   |
| 데이터베이스 포트  | 3306                                | 예                   |
| DB 클러스터 식별자  | database-test1                      | 예                   |
| DB 인스턴스 식별자  | database-1                          | 예                   |
| DB 엔진 버전   | 8.0.mysql_aurora.3.02.0             | 예                   |
| DB 파라미터 그룹   | default:aurora-mysql8.0             | 예                   |
| DB 클러스터 파라미터 그룹  | default:aurora-mysql8.0             | 예                   |

[손쉬운 생성(Easy Create)]과 함께 사용되는 기본 설정을 검토할 수 있다. **데이터베이스 생성 후 편집 가능** 열에는 데이터베이스 생성 후 어떤 옵션을 변경할 수 있는지 나와 있다.

- 설정의 해당 열에 **아니요**라고 되어 있지만 다른 설정을 원하는 경우 **표준 생성**을 사용하여 DB 클러스터를 만들 수 있다.
- 설정의 해당 열에 **예**라고 되어 있으며 다른 설정을 원하는 경우 **표준 생성**을 사용하여 DB 클러스터를 만들거나, DB 클러스터를 생성한 후 수정하여 설정을 변경할 수 있다.

## 12. Create database(데이터베이스 생성)를 선택한다.

DB 클러스터의 마스터 사용자 이름 및 암호를 보려면 **자격 증명 세부 정보 보기**를 선택한다.

DB 클러스터를 마스터 사용자로 연결하려면 화면에 나타난 사용자 이름과 암호를 사용한다.

## 중요

마스터 사용자 암호를 다시 볼 수는 없다. 따라서 기록을 해두지 않으면 이를 변경해야 한다.

DB Cluster 가 사용 가능한 상태가 되고 난 후에 마스터 사용자 암호를 변경해야 하는 경우에는 다음과 같은 방법으로 DB 클러스터를 수정할 수 있다. DB 클러스터 수정에 대한 자세한 정보는 [Amazon Aurora DB 클러스터 수정](#) 단원을 참조한다.

13. 데이터베이스 목록에서 새 Aurora MySQL DB 클러스터의 이름을 선택하면 세부 정보가 표시된다.

DB Cluster 를 사용할 준비가 될 때까지 **라이터 인스턴스**의 상태는 **생성 중**이다.

database-test1

관련

Q 데이터베이스(를) 기준으로 필터링

| DB 식별자                    | 상태    | 역할       | 엔진           | 리전 및 AZ         | 크기           | 작업 | CPU | 현재 활동 | 유지 관리 | VP |
|---------------------------|-------|----------|--------------|-----------------|--------------|----|-----|-------|-------|----|
| database-test1            | 사용 가능 | 리전 클러스터  | Aurora MySQL | ap-northeast-2  | 1 인스턴스       | -  | -   | -     | -     | -  |
| database-test1-instance-1 | 생성 중  | 라이터 인스턴스 | Aurora MySQL | ap-northeast-2a | db.r6g.large | -  | -   | -     | -     | vp |

엔드포인트 (2)

Find resources

| 엔드포인트 이름  | 상태   | 유형       | 포트   |
|---|------|----------|------|
| database-test1.cluster-cx1hah81oc13.ap-northeast-2.rds.amazonaws.com    | 생성 중 | 라이터 인스턴스 | 3306 |
| database-test1.cluster-ro-cx1hah81oc13.ap-northeast-2.rds.amazonaws.com | 생성 중 | 리더 인스턴스  | 3306 |

라이터 인스턴스의 상태가 **사용 가능**으로 변경되면 DB 클러스터에 연결할 수 있다. DB Instance Class 와 Storage 의 용량에 따라 새 DB Instance 를 사용할 수 있을 때까지 최대 20 분이 걸릴 수 있다.

### 3 단계: Aurora MySQL DB 클러스터에 연결

표준 SQL Client Application 을 사용해 DB Cluster 에 연결할 수 있다. 이 예시에서는 mysql 명령줄 Client 를 사용하여 Aurora MySQL DB 클러스터에 연결한다.

#### Aurora MySQL DB Cluster 에 연결하려면

1. DB Cluster 에 대한 **라이터 인스턴스**의 엔드포인트(DNS 이름)와 포트 번호를 찾는다.
  - a. AWS Management Console 에 로그인한 후 <https://console.aws.amazon.com/rds/>에서 Amazon RDS 콘솔을 오픈한다.
  - b. Amazon RDS 콘솔의 오른쪽 상단에서 DB Cluster 의 AWS Region 을 선택한다.
  - c. 탐색 창에서 **데이터베이스**를 선택한다.
  - d. Aurora MySQL DB Cluster 이름을 선택하여 세부 정보를 표시한다.
  - e. **연결 및 보안** 탭에서 **라이터 인스턴스**의 엔드포인트를 복사한다. 또한 포트 번호를 적어 놓는다. DB Cluster 에 연결하려면 엔드포인트와 포트 번호가 모두 필요하다.

The screenshot shows the AWS Management Console for the RDS database 'database-test1'. The 'Endpoints (2)' section is highlighted with a red box, showing two endpoints:

| 엔드포인트 이름  | 상태    | 유형       | 포트   |
|---|-------|----------|------|
| database-test1.cluster-cx1hah81ocl3.ap-northeast-2.rds.amazonaws.com    | 사용 가능 | 라이터 인스턴스 | 3306 |
| database-test1.cluster-ro-cx1hah81ocl3.ap-northeast-2.rds.amazonaws.com | 사용 가능 | 리더 인스턴스  | 3306 |

2. Linux Instance 용 Amazon EC2 사용 설명서에 있는 [Linux 인스턴스에 연결](#)의 단계를 따라 앞에서 만든 EC2 Instance 에 연결한다.

SSH 를 사용하여 EC2 Instance 에 연결하는 것이 좋다. Windows, Linux 또는 Mac 에 SSH Client Utility 가 설치된 경우 다음 명령 형식을 사용하여 Instance 에 연결할 수 있다.

```
ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
```

예를 들어 ec2-database-connect-key-pair.pem 이 Linux 의 /dir1 에 저장되어 있고, EC2 Instance 의 Public IPv4 DNS 가 ec2-12-345-678-90.compute-1.amazonaws.com 이라고 가정해 본다면, SSH 명령은 다음과 같이 표시된다.

```
ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-user@ec2-12-345-678-90.compute-1.amazonaws.com
```

3. EC2 Instance 에서 Software 를 업데이트하여 최신 버그 수정 및 보안 업데이트를 받는다. 이렇게 하려면 다음 명령을 사용한다.

```
$ sudo apt update
```

```
$ sudo apt upgrade
```

4. Ubuntu 22.04 LTS 에서 MariaDB 의 mysql 명령줄 Client 를 설치하려면 다음 명령을 실행한다.

```
$ sudo apt install mariadb-client -y
```

5. Aurora MySQL DB 클러스터에 연결한다. 예를 들어, 다음 명령을 입력한다. 이 작업을 통해 MySQL Client 를 사용하여 Aurora MySQL DB Cluster 에 연결할 수 있다.

*endpoint* 는 **라이터 인스턴스** 엔드포인트로 대체하고, *admin* 는 사용된 마스터 사용자 이름으로 대체한다. 암호를 묻는 메시지가 표시되면 사용한 마스터 암호를 제공한다.

```
$ mysql -h endpoint -P 3306 -u admin -p
```

사용자에 대한 암호를 입력하면 다음과 유사한 출력이 나타난다.

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 217
Server version: 8.0.23 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

6. SQL 명령을 실행한다.

예를 들어, 다음 SQL 명령은 현재 날짜 및 시간을 보여준다.

```
MySQL [(none)] > SELECT CURRENT_TIMESTAMP;
```

---

## 4 단계: EC2 인스턴스 및 DB 클러스터 삭제

생성한 Sample EC2 Instance 및 DB Cluster 에 연결하고 탐색한 후에는 요금이 더 이상 부과되지 않도록 삭제하도록 한다.

### EC2 인스턴스를 삭제하는 방법

1. AWS Management Console 에 로그인하고 <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엿는다.
2. 탐색 창에서 **인스턴스(Instances)**를 선택한다.
3. EC2 인스턴스를 선택하고 **인스턴스 상태**, **인스턴스 종료**를 차례로 선택한다.
4. 확인 메시지가 나타나면 **종료**를 선택한다.

EC2 인스턴스 삭제에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 종료](#)를 참조한다.

### DB 클러스터를 삭제하려면

1. AWS Management Console 에 로그인한 후 <https://console.aws.amazon.com/rds/>에서 Amazon RDS 콘솔을 엿는다.
2. **데이터베이스**를 선택하고 DB 클러스터에 연결된 DB 인스턴스를 선택한다.
3. [ **Actions**]에 대해 [Delete]를 선택한다.
4. **최종 스냅샷 생성 여부**를 선택 해제한다.
5. 확인을 완료하고 **삭제**를 선택한다.

DB 클러스터에 연결된 모든 DB 인스턴스가 삭제되고 나면 DB 클러스터가 자동으로 삭제된다.