

# TDSQL for MySQL

## 시작하기

## 제품 문서



## 저작권 고지

©2013–2026 Tencent Cloud. 모든 권리 보유.

본 문서의 저작권은 텐센트 클라우드(Tencent Cloud)에 단독으로 귀속됩니다. 텐센트 클라우드의 사전 서면 승인 없이는 어떠한 주체도 본 문서 내용의 전부 또는 일부를 복제·수정·표절·전송하는 등 어떠한 형태로도 이용할 수 없습니다.

## 상표 고지



텐센트 클라우드(Tencent Cloud) 및 관련 서비스의 모든 상표는 텐센트(Tencent) 그룹 산하 법인들(모회사, 자회사 및 계열사 포함)이 소유합니다. 본 문서에 언급된 제3자 상표는 해당 법적 권리자가 소유합니다.

## 서비스 고지

본 문서는 고객에게 텐센트 클라우드(Tencent Cloud) 제품 및 서비스 전부 또는 일부에 대한 현재적 개괄적 정보를 제공 하기 위한 것으로, 특정 제품·서비스의 내용은 수시로 변경될 수 있습니다. 고객이 실제 구매한 제품·서비스의 적용 기준은 고객과 텐센트 클라우드간 체결된 상업계약에 명시된 내용이 우선하며, 별도 서면 합의가 없는 한 텐센트 클라우드는 본 문서 내용에 대해 법적 효력이 있는 명시적·묵시적 진술이나 보증을 일체 하지 않습니다.

## 목록:

시작하기

InnoDB 엔진

인스턴스 생성

인스턴스 연결

샤딩된 테이블 관리

# 시작하기

## InnoDB 엔진

## 인스턴스 생성

최종 업데이트 날짜: 2024-01-06 15:14:47

본문은 콘솔에서 TDSQL for MySQL 인스턴스를 생성하는 방법을 설명합니다.

### 전제 조건

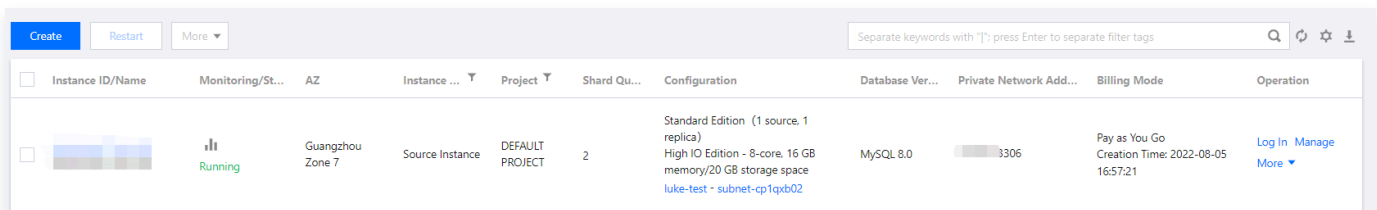
Tencent Cloud 계정을 등록하고 본인 확인을 완료합니다.



- Tencent Cloud 계정 생성 방법:
- 실명 인증 방법:

### 작업 단계

1. [TDSQL for MySQL 구매 페이지](#)에 로그인하여, 각 항목의 구성 정보를 필요에 따라 선택하고 오류가 없는지 확인한 뒤, 즉시 구매를 클릭합니다.
  - **과금 방식:** 정액 과금제 및 종량제가 지원됩니다.
    - 장기 수요가 안정적인 비즈니스라면 정액 과금제를 권장합니다.
    - 비즈니스의 요청량이 순간적으로 크게 변동하는 경우 종량제를 권장합니다.
  - **리전:** TDSQL for MySQL 인스턴스를 배포할 리전을 선택합니다. 연결할 CVM 인스턴스와 동일한 리전을 사용하는 것이 좋습니다. 다른 리전의 Tencent Cloud 서비스는 사설망을 통해 서로 통신할 수 없습니다. 리전은 구매 후 수정할 수 없습니다.
  - **네트워크 유형:** TDSQL for MySQL 인스턴스가 있는 네트워크를 선택합니다. 연결할 CVM 인스턴스와 동일한 리전에서 동일한 VPC를 선택하는 것이 좋습니다. 그렇지 않으면 사설망을 통해 CVM 인스턴스와 데이터베이스에 연결할 수 없습니다.
  - **인스턴스 버전:** 자세한 내용은 [인스턴스 아키텍처](#)를 참고하십시오.
  - **소스/복제본 AZ:** 다른 소스 및 복제본 AZ를 선택하여 장애 및 AZ 중단으로부터 데이터베이스를 보호합니다.
  - **커널 버전:** MySQL 5.6 커널은 분산 트랜잭션(XA)을 지원하지 않습니다. 이 기능이 필요한 경우 v5.7을 선택하십시오. 자세한 내용은 [개요](#)를 참고하십시오.
  - **샤드 수량/사양/디스크:** 데이터 왜곡을 방지하기 위해 2개, 4개 또는 8개의 샤드를 권장합니다. 샤드 설정에 대한 자세한 내용은 [인스턴스 설정 및 샤드 설정 선택](#)을 참고하십시오.
  - **프로젝트:** TencentDB 인스턴스가 속한 프로젝트를 선택합니다. 기본 프로젝트가 사용됩니다.
  - **태그:** 태그를 사용하여 리소스를 분류하고 관리합니다. 자세한 내용은 [Overview](#)를 참고하십시오.
  - **보안 그룹:** 보안 그룹의 생성 및 관리에 관한 내용은 [보안 그룹 설정](#)을 참고하십시오.

- **인스턴스 이름:** 지금 또는 나중에 인스턴스의 이름을 지정합니다.
  - **지원되는 문자 세트:** UTF8, LATIN1, GBK, UTF8MB4 및 GB18030이 지원됩니다.
  - **테이블 이름 대소문자 구분:** 대소문자 구분(lower\_case\_table\_names = 0) 또는 대소문자 구분하지 않음(lower\_case\_table\_names = 1)을 선택합니다. 초기화 매개변수이며 데이터베이스가 초기화된 후에는 수정할 수 없습니다.
  - **강제 동기화 사용:** 강제 동기화(다운그레이드 가능) 및 비동기화가 지원됩니다. 자세한 내용은 [강제 동기화](#)를 참고하십시오.
  - **과금에 대한 자세한 내용은** [제품 가격](#)을 참고하십시오.
2. 결제 후 인스턴스 리스트로 되돌아갑니다. 인스턴스 상태가 **실행 중**으로 변경되면, 정상적으로 사용할 수 있습니다.



Instance ID/Name	Monitoring/St...	AZ	Instance ...	Project	Shard Qu...	Configuration	Database Ver...	Private Network Add...	Billing Mode	Operation
<input type="checkbox"/>	 Running	Guangzhou Zone 7	Source Instance	DEFAULT PROJECT	2	Standard Edition (1 source, 1 replica) High IO Edition - 8-core, 16 GB memory/20 GB storage space <a href="#">luke-test - subnet-cp1qx602</a>	MySQL 8.0	 3306	Pay as You Go Creation Time: 2022-08-05 16:57:21	<a href="#">Log In</a> <a href="#">Manage</a> <a href="#">More</a>

## 후속 작업

사설망 또는 공중망 주소에서 MySQL용 TDSQL 인스턴스에 연결하는 방법에 대한 자세한 내용은 [인스턴스 연결](#)을 참고하십시오.

# 인스턴스 연결

최종 업데이트 날짜: 2024-01-06 15:14:47

## 연결 방식

다음과 같은 방법으로 TDSQL for MySQL 인스턴스에 연결할 수 있습니다.

- **사설망 연결:** CVM 인스턴스를 사용하여 TDSQL for MySQL 인스턴스의 사설망 주소에 연결할 수 있습니다. 이 방식은 Tencent Cloud의 고속 사설망을 활용하며 지연 시간이 짧은 것이 특징입니다.
- 두 인스턴스는 동일한 계정 및 동일한 리전의 동일한 VPC에 있거나 클래식 네트워크에 있어야 합니다.
- 사설망 주소는 기본적으로 TencentDB에서 제공하며 [TDSQL for MySQL 콘솔](#)의 인스턴스 리스트 또는 인스턴스 세부 정보 페이지에서 볼 수 있습니다.

### 📌 설명:

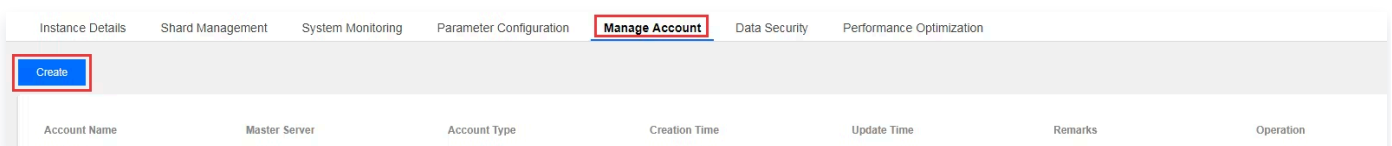
다른 VPC(동일하거나 다른 리전의 동일하거나 다른 계정)에 있는 CVM 및 TencentDB 인스턴스는 [Cloud Connect Network](#)를 통해 사설망으로 상호 연결할 수 있습니다.

- **공중망 주소 연결:** 사설망을 통해 액세스할 수 없는 경우 공중망 주소에서 TDSQL for MySQL 인스턴스에 연결할 수 있습니다. 공중망 주소는 [수동으로 활성화](#)해야 합니다. [TDSQL for MySQL 콘솔](#)의 인스턴스 세부 정보 페이지에서 볼 수 있으며 더 이상 필요하지 않은 경우 비활성화할 수 있습니다.
- 공중망 주소를 활성화하면 데이터베이스 서비스가 공중망에 노출되어 데이터베이스 침입이나 공격이 발생할 수 있습니다. 사설망을 사용하여 데이터베이스에 연결하는 것이 좋습니다.
- TencentDB에 대한 공중망 연결은 데이터베이스의 개발 또는 보조 관리에 적합하지만, 잠재적으로 제어할 수 없는 요인으로 인해 DDOS 공격 및 대량의 버스트 트래픽 등 공중망 연결을 사용할 수 없게 될 수 있으므로 프로덕션 환경에서의 비즈니스 액세스에는 적합하지 않습니다.
- 현재 공중망 액세스를 지원하는 리전은 광저우, 상하이, 베이징, 청두, 난징입니다.
- 공중망 액세스 활성화를 위해서는 보안 그룹을 바인딩해야 하며, 자세한 내용은 [보안 그룹 설정](#)을 참고하십시오.

## 준비 작업

### 계정 생성

1. [TDSQL for MySQL 콘솔](#)에 로그인합니다. 인스턴스 리스트에서 인스턴스 ID 또는 **작업** 열의 **관리**를 클릭하여 인스턴스 관리 페이지로 들어갑니다.
2. 인스턴스 관리 페이지에서 **계정 관리** 탭을 선택하고 **계정 생성**을 클릭합니다.



3. 팝업 창에서 계정 이름, 호스트, 비밀번호 등을 입력합니다. 모든 항목이 올바른지 확인한 후 **확인, 다음 단계**를 클릭합니다.

호스트 이름은 실제 네트워크 송신 주소입니다. 필요한 경우 %를 입력하여 이 인스턴스가 모든 IP에 액세스할 수 있음을 나타낼 수 있습니다.

4. 권한 수정 팝업 창에서 필요에 따라 권한을 부여하고 **수정**을 클릭합니다. 변경 사항을 취소하려면 **수정 취소**를 클릭합니다.

왼쪽의 사이드바는 MySQL 관리와 완전히 호환되는 그래픽 인터페이스를 제공합니다. 권한은 열 수준에서 관리할 수 있습니다.

**Modification Permission** ✕

Account name: test@%

Set Database Permissions Purge Reset

Global Privileges		
<span style="color: #007bff;">+</span> Object-Level Privilege	<input type="checkbox"/> SELECT	<input type="checkbox"/> INSERT
	<input type="checkbox"/> UPDATE	<input type="checkbox"/> DELETE
	<input type="checkbox"/> CREATE	<input type="checkbox"/> DROP
	<input type="checkbox"/> REFERENCES	<input type="checkbox"/> INDEX
	<input type="checkbox"/> ALTER	<input type="checkbox"/> CREATE TEMPORARY TABLES
	<input type="checkbox"/> LOCK TABLES	<input type="checkbox"/> EXECUTE
	<input type="checkbox"/> CREATE VIEW	<input type="checkbox"/> SHOW VIEW
	<input type="checkbox"/> CREATE ROUTINE	<input type="checkbox"/> ALTER ROUTINE
	<input type="checkbox"/> EVENT	<input type="checkbox"/> TRIGGER
	<input type="checkbox"/> SHOW DATABASES	<input type="checkbox"/> PROCESS
	<input type="checkbox"/> REPLICATION CLIENT	<input type="checkbox"/> REPLICATION SLAVE
	<input type="checkbox"/> Select All	

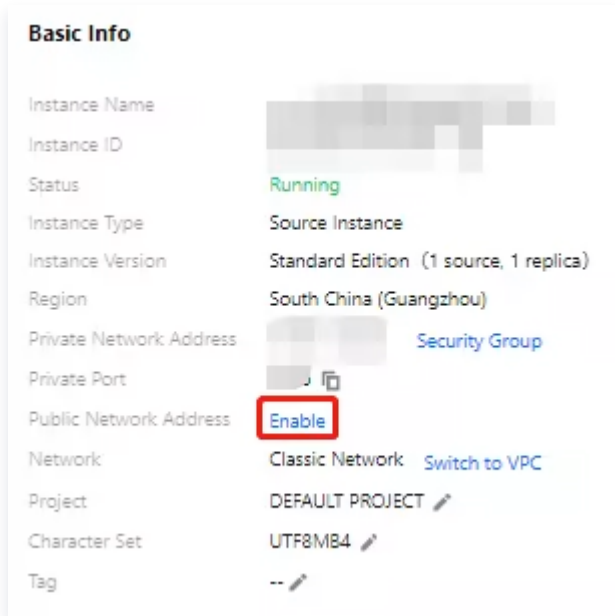
Save Settings
Set Later

5. 계정 리스트로 돌아가서 **권한 수정**을 클릭하여 사용자 권한을 수정하거나 **계정 복제**를 클릭하여 현재 계정 권한을 완전히 복사하여 새 계정을 생성할 수 있으며, **더 보기**를 클릭하여 비밀번호를 재설정 또는 계정을 삭제할 수 있습니다.

帐号名	主机	帐号类型	创建时间	更新时间	备注	操作
hao	%	常规帐号	2020-06-24 16:57:54	2020-06-24 16:57:54		<a href="#">修改权限</a> <a href="#">克隆帐号</a> <span style="font-size: small;">更多 ▾</span>

## (선택 사항)공중망 주소 활성화

1. [TDSQL for MySQL 콘솔](#)에 로그인하고 인스턴스 리스트에서 인스턴스 ID를 클릭하여 인스턴스 세부 정보 페이지로 들어간 다음 기본 정보 섹션의 **공중망 주소** 옆에 있는 **활성화**를 클릭합니다.



2. 성공적으로 활성화하면 **공중망 주소** 옆에 있는 공중망 주소와 포트 번호를 볼 수 있습니다. TDSQL for MySQL은 각 인스턴스에 고유한 공용 IP 및 포트 쌍을 할당합니다.

계정을 만들고 사설/공중망 주소를 얻은 후 3rd party 도구 및 프로그램 드라이버를 통해 TDSQL for MySQL에 연결할 수 있습니다.

- Windows에서는 명령 라인, 클라이언트, JDBC 드라이버의 연결 방법을 예로 들 수 있습니다.
- Linux에서는 명령 라인 연결 방법을 예로 들 수 있습니다.

## Windows 장치에서 연결

### Windows 명령 라인으로 연결

1. Windows 명령 라인 툴을 열고 MySQL의 올바른 경로 아래에 다음 명령을 입력합니다.

```
mysql -h 사설/공중망 주소 -P 포트 번호 -u 사용자 이름 -p
Enter password: ***** (비밀번호 입력)
```

2. 다음과 같은 정보가 표시되면 데이터베이스가 성공적으로 연결된 것이며 작업을 진행할 수 있습니다.

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

### Windows 클라이언트 연결

1. MySQL Workbench 및 SQLyog와 같은 표준 SQL 클라이언트를 다운로드합니다. 여기에서는 SQLyog를 예로 듭니다.
2. SQLyog를 열고 **파일 > 새 연결**을 선택하고 호스트 주소, 포트, 사용자 이름 및 암호를 입력한 후 **연결**을 클릭합니다.

- MySQL 호스트 주소: 이전에 얻은 인스턴스의 사설/공중망 주소를 입력합니다.
- 사용자 이름: 이전에 생성한 계정 이름을 입력합니다.
- 비밀번호: 계정의 비밀번호를 입력합니다. 암호를 잊어버린 경우 [콘솔](#)에서 재설정하십시오.
- 포트: 이전에 얻은 인스턴스의 포트 번호를 입력합니다.

3. 접속 성공 후의 인터페이스는 아래와 같습니다. 이 인터페이스에서 데이터베이스를 조작할 수 있습니다.

## Windows JDBC 드라이버 연결

TDSQL for MySQL은 프로그램 드라이버와의 연결을 지원합니다. 이 문서에서는 Java JDBC Driver for MySQL(Connector/J)을 예로 사용합니다.

1. [MySQL 공식 웹사이트](#)에서 JDBC jar 패키지를 다운로드하고 Java에서 참고하는 Library로 가져옵니다.
2. 다음 코드를 실행하여 JDBC를 호출합니다.

```
public static final String url = "사설/공중망 주소";
public static final String name = "com.mysql.jdbc.Driver"; //JDBC 드라이버 호출
public static final String user = "사용자 이름";
public static final String password = "비밀번호";
//JDBC
Class.forName("com.mysql.jdbc.Driver");
Connection conn=DriverManager.getConnection("url, user, password");
//
conn.close();
```

3. 연결에 성공하면 데이터베이스에서 다른 작업을 수행할 수 있습니다.

## Linux 장치에서 연결

### Linux 명령 라인으로 연결

본 문서에서는 CentOS 7.2 64비트의 CVM 인스턴스를 예로 들어 설명합니다. CVM 인스턴스 구매에 대한 자세한 내용은 [구매 방식](#)을 참고하십시오.

1. Linux CVM에 로그인한 후, `yum install mysql` 을 입력하고 CentOS의 패키지 관리 소프트웨어 Yum을 사용하여 Tencent Cloud의 이미지 소스에서 MySQL 클라이언트를 다운로드하여 설치합니다.

```
CentOS Linux 7 (Core)
Kernel 3.10.0-327.36.3.el7.x86_64 on an x86_64

UM_213_43_centos login: root
Password:
[root@UM_213_43_centos ~]# yum install mysql
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
--> Package mariadb.x86_64 1:5.5.52-1.el7 will be installed
--> Processing Dependency: mariadb-libs(x86-64) = 1:5.5.52-1.el7 for package: 1:mariadb-5.5.52-1.el7.x86_64
--> Running transaction check
---> Package mariadb-libs.x86_64 1:5.5.50-1.el7_2 will be updated
---> Package mariadb-libs.x86_64 1:5.5.52-1.el7 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                               Arch                               Version                               Repository
=====
Installing:
mariadb                               x86_64                             1:5.5.52-1.el7                         os
Updating for dependencies:
mariadb-libs                           x86_64                             1:5.5.52-1.el7                         os

Transaction Summary
=====
Install 1 Package
Upgrade ( 1 Dependent package)

Total download size: 9.5 M
Is this ok [y/d/N]:
```

2. 명령 라인에 complete가 표시되면 MySQL 클라이언트가 성공적으로 설치된 것입니다.
3. `mysql -h 사설/공중망 주소 -P 포트 -u 사용자 이름 -p` 를 입력하여 TDSQL for MySQL에 연결합니다. 그런 다음 샤딩을 수행할 수 있습니다.

아래의 이미지는 'show databases;'를 예시로 합니다.

```
[root@UM_3_86_centos ~]# mysql -hdcdbt ██████████ myqcloud.com -P12 -uexample1 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 493901845
Server version: 10.0.10-proxy Source distribution

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

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

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| SysDB |
| information_schema |
| mysql |
| performance_schema |
| test |
+-----+
5 rows in set (0.01 sec)

MySQL [(none)]> _
```

# 샤딩된 테이블 관리

최종 업데이트 날짜: 2024-01-06 15:14:47

본문은 TDSQL for MySQL의 몇 가지 간단한 데이터베이스 작업을 설명하기 위해 샤딩된 테이블을 예로 들어 설명합니다.

## 테이블 생성

- 분할된 테이블, 분할되지 않은 테이블 및 브로드캐스트 테이블 간의 차이점에 대한 자세한 내용은 [개요](#)를 참고하십시오.
- 샤드키(shardkey)에 대한 제한 사항에 대한 자세한 내용은 [테이블 생성](#)을 참고하십시오.
- 샤딩된 테이블을 생성하려면 샤드키(shardkey)를 지정해야 합니다. 샘플 코드는 다음과 같습니다.

```
mysql> create table test1(id int primary key,name varchar(20),addr
varchar(20)) shardkey=id;
Query OK,0 rows affected(0.15 sec)
```

## 데이터 삽입

### ⚠ 주의 사항:

샤드키는 insert 문에 포함되어야 하며 그렇지 않으면 작업이 거부됩니다.

방금 만든 테이블에 데이터를 삽입합니다. 샘플 코드는 다음과 같습니다.

```
mysql> insert into test1(id,name) VALUES(1,'test');
Query OK,1 rows affected(0.08 sec)
mysql> insert into test3(name,addr) values('example','shenzhen');
ERROR 7013 (HY000): Proxy ERROR:get_shardkeys return error
```

## 데이터 쿼리

### ⚠ 주의 사항:

데이터를 쿼리할 때 요청이 분산 경로에 따라 해당 샤드로 자동 라우팅되어 최고의 효율성을 달성할 수 있도록 명령문에 샤드키를 포함하는 것이 좋습니다. 그렇지 않으면 TDSQL이 자동으로 전체 테이블을 스캔한 다음 게이트웨이에서 결과를 집계하므로 효율성이 저하됩니다.

데이터 쿼리를 위한 샘플 코드는 다음과 같습니다.

```
mysql> select id from test1 where id=1;
```

## 데이터 삭제

### ⚠ 주의 사항:

delete 문에는 where 절이 포함되어야 하며 where 절에 샤드키를 포함하는 것이 좋습니다.

데이터 삭제를 위한 샘플 코드는 다음과 같습니다.

```
mysql> delete from test1 where id=1;  
Query OK, 1 row affected (0.02 sec)
```