

Criação de Usuários

Sintaxe básica para criação de um usuário:

```
CREATE USER 'nome_do_usuario'@'localhost' IDENTIFIED BY 'senha';
```

Explicação dos component Explicação dos Componentes da Sintaxe

Quando criamos ou modificamos um usuário no MySQL, utilizamos uma sintaxe específica que inclui diversos componentes. Vamos detalhar cada um deles para entender melhor seu propósito e uso.

nome_do_usuario

Descrição: Representa o identificador único do usuário dentro do sistema de gerenciamento de banco de dados MySQL. O nome do usuário é utilizado para login e autenticação.

Uso: nome_do_usuario deve ser único no contexto do host a partir do qual o usuário se conecta. Ele pode conter letras, números e alguns símbolos, mas deve ser especificado entre aspas simples (' ') se incluir caracteres especiais ou for uma string que possa ser interpretada de outra forma pelo sistema.

localhost

Descrição: Especifica o host ou o endereço de rede a partir do qual o usuário tem permissão para se conectar ao servidor MySQL. localhost é um hostname especial que indica que a conexão é feita a partir da mesma máquina onde o servidor MySQL está executando.

Uso: Usar localhost como host significa que apenas tentativas de conexão originadas na mesma máquina física ou virtual serão permitidas para esse usuário. Tentativas de conexão de outras máquinas serão rejeitadas.

IDENTIFIED BY

Descrição: É uma cláusula utilizada na criação ou alteração de usuários para definir ou modificar a senha do usuário.

Uso: Seguida pela senha, indica o método de autenticação e a credencial que o usuário deve fornecer para se conectar ao servidor. A senha deve ser especificada entre aspas simples.

senha

Descrição: A senha é a chave de segurança que o usuário precisa fornecer para autenticar sua identidade ao tentar se conectar ao servidor MySQL.

Uso: Deve ser uma combinação segura de letras, números e símbolos para proteger o acesso ao banco de dados. É recomendado utilizar senhas fortes para evitar acesso não autorizado.

Diferença entre localhost e % para Especificar de Onde o Usuário Pode se Conectar

localhost

Significado: Indica que o usuário só pode se conectar ao banco de dados MySQL a partir da mesma máquina onde o servidor está instalado. Utilizar localhost é uma prática comum para usuários que interagem com o banco de dados localmente.

Uso: Ideal para cenários onde as aplicações que acessam o banco de dados estão hospedadas no mesmo servidor que o MySQL, garantindo que não haja conexões externas para esse usuário.

%

Significado: Representa um curinga que permite conexões a partir de qualquer host. Usar % na definição do host do usuário no MySQL significa que não há restrições de IP ou hostname para a conexão desse usuário.

Uso: Adequado para situações onde o usuário precisa acessar o banco de dados de múltiplas localizações ou quando a aplicação que acessa o banco de dados está hospedada em uma máquina diferente daquela do servidor MySQL. No entanto, é crucial aplicar medidas de segurança adicionais, como VPNs ou listas de permissão de IP, para mitigar riscos de segurança.

A escolha entre localhost e % deve ser cuidadosamente considerada, levando em conta os requisitos de acesso e as práticas de segurança adequadas para proteger o banco de dados contra acessos não autorizados.

es da sintaxe: nome_do_usuario, localhost, IDENTIFIED BY, senha.

Diferença entre localhost e % para especificar de onde o usuário pode se conectar.

Concessão de Privilégios

No MySQL, os privilégios definem o tipo de operações que um usuário pode executar. Eles são fundamentais para garantir a segurança e o controle adequado do acesso aos dados e às funcionalidades do banco de dados. Aqui está uma lista

detalhada dos tipos de privilégios disponíveis no MySQL, juntamente com uma breve descrição de cada um:

ALL PRIVILEGES - Concede todos os privilégios disponíveis para um banco de dados ou tabela.

ALTER - Permite modificar a estrutura de tabelas existentes, como adicionar, excluir ou modificar colunas e índices.

ALTER ROUTINE - Permite alterar ou excluir procedimentos armazenados e funções.

CREATE - Permite criar novas tabelas ou bancos de dados.

CREATE ROUTINE - Permite criar novos procedimentos armazenados e funções.

CREATE TEMPORARY TABLES - Permite criar tabelas temporárias.

CREATE VIEW - Permite criar novas views.

DELETE - Permite excluir linhas de tabelas.

DROP - Permite excluir tabelas ou views.

EVENT - Permite criar, alterar, excluir ou executar eventos para o Agendador de Eventos.

EXECUTE - Permite executar procedimentos armazenados.

FILE - Permite ler e escrever arquivos no servidor usando comandos como **LOAD DATA INFILE** e **SELECT ... INTO OUTFILE**.

GRANT OPTION - Permite conceder ou revogar privilégios a outros usuários.

INDEX - Permite criar ou excluir índices.

INSERT - Permite inserir linhas em tabelas.

LOCK TABLES - Permite usar o comando **LOCK TABLES** em tabelas às quais você tem privilégio de **SELECT**.

PROCESS - Permite visualizar todos os processos com **SHOW PROCESSLIST**.

PROXY - Permite fazer login como outro usuário.

REFERENCES - Não tem efeito prático; está presente por razões de compatibilidade com o SQL padrão.

RELOAD - Permite executar comandos **FLUSH**.

REPLICATION CLIENT - Permite ao usuário perguntar onde os servidores master ou slave estão.

REPLICATION SLAVE - Necessário para o servidor se conectar a um master como um slave.

SELECT - Permite ler dados de tabelas.

SHOW DATABASES - Permite visualizar a lista de todos os bancos de dados com o comando SHOW DATABASES.

SHOW VIEW - Permite executar o comando SHOW CREATE VIEW, que mostra como uma view foi criada.

SHUTDOWN - Permite usar o comando SHUTDOWN para desligar o servidor MySQL.

SUPER - Permite usar uma ampla variedade de comandos administrativos, como CHANGE MASTER TO, KILL, PURGE BINARY LOGS, SET GLOBAL, e log de consultas.

TRIGGER - Permite criar ou excluir triggers.

UPDATE - Permite atualizar linhas em tabelas.

USAGE - Um sinônimo para "nenhum privilégio". Geralmente usado para criar um usuário sem conceder nenhum privilégio específico.

Cada um desses privilégios pode ser concedido em diferentes níveis, variando de global, banco de dados, tabela, coluna, até o nível de procedimento armazenado ou função. A escolha dos privilégios a serem concedidos deve ser feita com cuidado, seguindo o princípio do menor privilégio necessário para realizar uma tarefa.

Como conceder privilégios de root a um usuário:

```
GRANT ALL PRIVILEGES ON *.* TO 'nome_do_usuario'@'localhost';
```

conceder ao usuario_exemplo os privilégios de SELECT, INSERT, e UPDATE somente na tabela tabela_exemplo do banco de dados banco_exemplo:

```
GRANT SELECT, INSERT, UPDATE ON banco_exemplo.tabela_exemplo TO 'usuario_exemplo'@'localhost';
```

Revogação de privilégios:

```
REVOKE ALL PRIVILEGES ON *.* FROM 'nome_do_usuario'@'localhost';
```

Como visualizar os privilégios de um usuário:

```
SHOW GRANTS FOR 'nome_do_usuario'@'localhost';
```

Práticas Recomendadas

Nunca usar o usuário root para aplicações.

Criar usuários específicos para tarefas específicas.

Regularmente revisar e atualizar os privilégios dos usuários.

Utilizar senhas fortes.