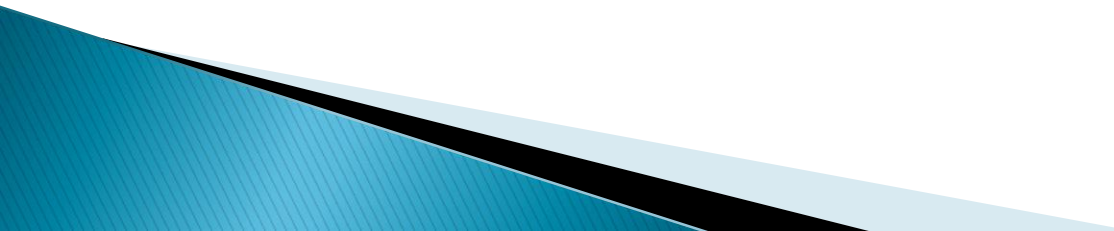


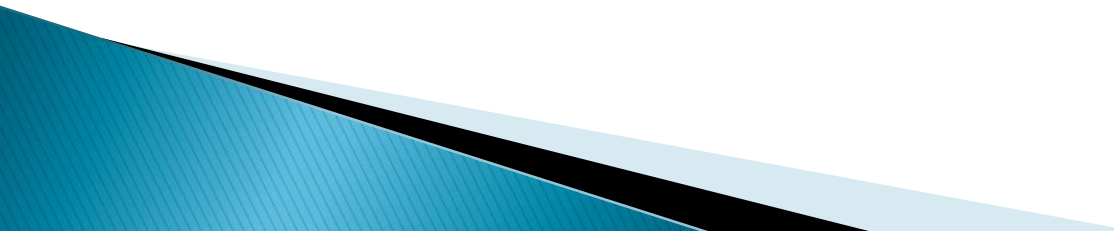
Privilégios de usuários no MySQL

Prof. Me. Sérgio Carlos Portari Júnior
Banco de Dados II

Administração de Usuários e Privilégios no Banco de Dados

- ▶ Todo agrupamento de bancos de dados possui um conjunto de usuários de banco de dados.
 - ▶ Estes usuários são distintos dos usuários gerenciados pelo sistema operacional onde o servidor executa.
 - ▶ Os usuários possuem objetos de banco de dados (por exemplo, tabelas), e podem conceder privilégios nestes objetos para outros usuários controlando, assim, quem pode acessar qual objeto.
- 

Usuários

- ▶ Conceitualmente, os usuários de banco de dados são completamente distintos dos usuários de sistema operacional.
 - ▶ Na prática, pode ser conveniente manter correspondência, mas não é requerido.
 - ▶ Os nomes dos usuários de banco de dados são globais para todo o agrupamento da base de dados (e não próprio para cada base de dados).
- 

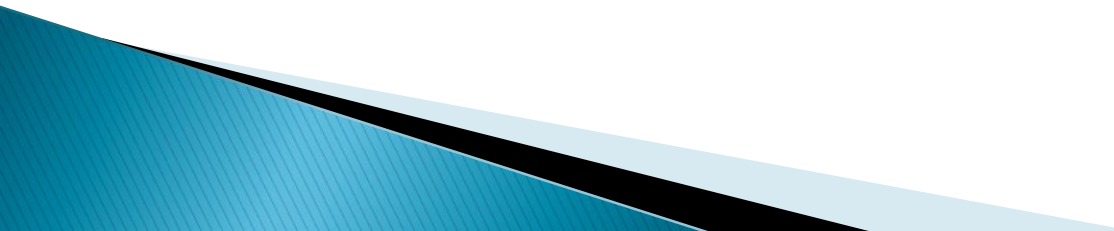
Criando usuários

- ▶ Para criar um usuário deve ser utilizado o comando SQL ***CREATE USER:***
- ▶ `CREATE USER 'diego'@'localhost' IDENTIFIED BY '123456';`
- ▶ Perceba que estamos criando um usuário com nome de “diego” no servidor local, com a senha (IDENTIFIED BY) “123456”.
- ▶ Caso você queira criar um usuário sem senha, o que torna o sistema inseguro, você pode digitar o comando até o nome do usuário e o servidor.

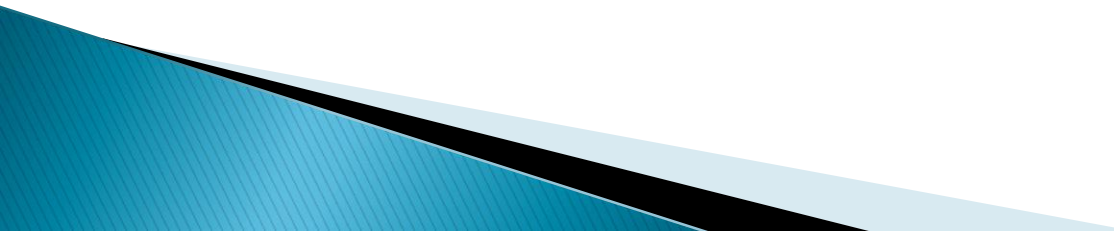
Alterando a senha de usuário

- ▶ Mesmo que você defina ou não uma senha ao criar um usuário, você pode alterar a senha dele posteriormente, através do comando SET PASSWORD, seguindo o exemplo:
- ▶ SET PASSWORD FOR diego = PASSWORD('abc123');
- ▶ O uso da função Password serve para criptografar a senha no banco.

Excluindo usuários

- ▶ Para remover um usuário existente deve ser utilizado o comando *DROP USER*:
 - ▶ `DROP USER 'diego'@'localhost';`
 - ▶ Ao excluir um usuário, todos os seus privilégios também serão excluídos.
- 

Privilégios

- ▶ O sistema de privilégios do MySQL garante que todos usuários possam fazer exatamente as operações que lhe é permitido.
 - ▶ Quando você conecta a um servidor MySQL, sua identidade é determinada pela máquina de onde você conectou e o nome de usuário que você especificou.
 - ▶ O sistema concede privilégios de acordo com sua identidade e com o que você deseja fazer.
- 

Privilégios

Privilégio	Descrição
ALL [PRIVILEGES]	Todos os privilégios exceto GRANT OPTION
ALTER	Permite executar ALTER TABLE
CREATE	Permite executar CREATE TABLE
CREATE TEMPORARY TABLES	Permite executar CREATE TEMPORARY TABLE
DELETE	Permite executar DELETE
DROP	Permite executar DROP TABLE
EXECUTE	Permite executar stored procedures (MySQL 5.0)
FILE	Permite executar SELECT ... INTO OUTFILE e LOAD DATA INFILE
INDEX	Permite executar CREATE INDEX e DROP INDEX
INSERT	Permite executar INSERT

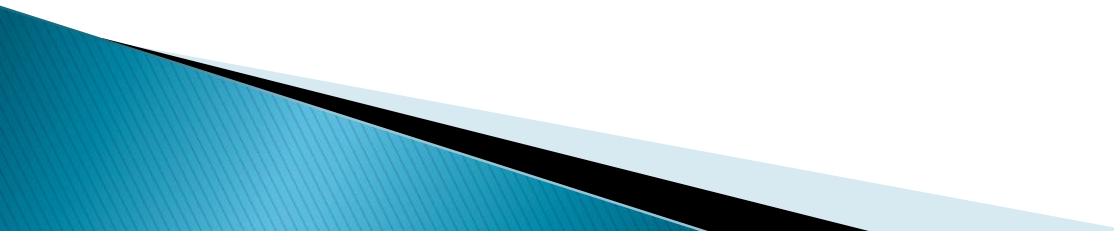
Privilégios

Privilégio	Descrição
LOCK TABLES	Permite executar LOCK TABLES em tabelas que você tenha o privilégio SELECT
PROCESS	Permite executar SHOW FULL PROCESSLIST
REFERENCES	Ainda não está implementado
RELOAD	Permite executar FLUSH
REPLICATION CLIENT	Permite ao usuário obter a localização do Master ou Slave
REPLICATION SLAVE	Necessário para a replicação Slave (leitura dos eventos do log binário do Master)
SELECT	Permite executar SELECT
SHOW DATABASES	exibe todos os bancos de dadosE
SHUTDOWN	Permite executar mysqladmin shutdownE

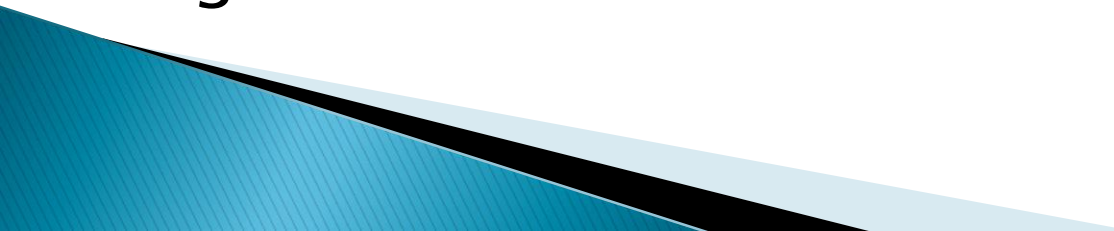
Privilégios

Privilégio	Descrição
SUPER	Permite executar CHANGE MASTER, KILL , PURGE MASTER LOGS e SET GLOBAL. Permite conectar-se ao servidor uma vez, mesmo que o max_connections tenha sido atingido
UPDATE	Permite executar UPDATE
USAGE	Sinônimo para “no privileges”
GRANT OPTION	Permite ao usuário repassar os seus privilégios

Privilégios

- ▶ Uma vez informados os privilégios do usuário, você deverá indicar o nível ao qual o privilégio se aplica.
 - ▶ Os comandos GRANT e REVOKE permitem aos administradores do sistema criar usuários e conceder e revogar direitos aos usuários do MySQL em quatro níveis de privilégios:
- 


Privilégios

- ▶ **Nível Global** – Privilégios globais aplicam para todos os bancos de dados em um determinado servidor.
 - ▶ Estes privilégios são armazenados na tabela `mysql.user`.
 - ▶ **GRANT ALL ON *.* e REVOKE ALL ON *.*** concederão e revogarão apenas privilégios globais.
- 

Privilégios

- ▶ **Nível dos bancos de dados** – Privilégios de bancos de dados aplicam-se a todas as tabelas em um determinado banco de dados.
- ▶ Estes privilégios são armazenados nas tabelas `mysql.db` e `mysql.host`.
- ▶ **GRANTALL ON db.*** e **REVOKE ALL ON db.*** concederão e revogarão apenas privilégios de banco de dados.

Privilégios

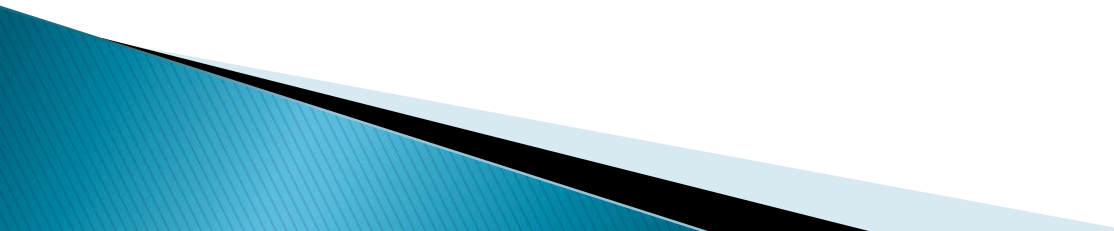
- ▶ **Nível das tabelas** – Privilégios de tabelas aplicam-se a todas as colunas em uma determinada tabela.
 - ▶ Estes privilégios são armazenados na tabela `mysql.tables_priv`.
 - ▶ **GRANT ALL ON db.table** e **REVOKE ALL ON db.table** concederão e revogarão apenas privilégios de tabelas.
- 

Privilégios

- ▶ **Nível das colunas** – Privilégios de colunas aplicam-se a uma única coluna em uma determinada tabela.
- ▶ Estes privilégios são armazenados na tabela `mysql.columns_priv`.
- ▶ **GRANT SELECT (coluna, coluna), Update (coluna) ON banco.tabela TO usuario@local IDENTIFIED BY “senha”;**

Concedendo Privilégios

```
GRANT priv [(colunas)] [, priv [(colunas)]] ...  
ON {*. * | db.* | db.tabela}  
TO usuario [IDENTIFIED BY 'senha']  
[, usuario [IDENTIFIED BY 'senha']] ...  
[WITH [GRANT OPTION |  
MAX_QUERIES_PER_HOUR contador |  
MAX_UPDATES_PER_HOUR contador |  
MAX_CONNECTIONS_PER_HOUR contador]]
```



Privilégios – Exemplos

- ▶ **GRANT select ON blog.pessoas TO "diego";**
Concede comando "select" para a tabela "pessoas" dentro do BD "blog" ao usuário "diego".
- ▶ **GRANT all ON blog.* TO "diego" WITH GRANT OPTION;**
Concede todos os comandos para todas as tabelas dentro do BD "blog" ao usuário "diego", e ele terá habilidade de fornecer à outros usuários quaisquer privilégios que ele tenha em um nível específico de privilégio. Cuidado com este tipo de comando, pois pode ser uma brecha de segurança.

Privilégios – Exemplos

- ▶ **GRANT select, insert, update ON blog.* TO "diego";**
Concede os comandos de SELECT, INSERT e UPDATE para todas as tabelas dentro do BD "blog" ao usuário "diego".
- ▶ **GRANT all ON *.* TO "diego" WITH GRANT OPTION;**
O usuário "diego" terá todos os privilégios em todos os BDs e tabelas, com a possibilidade de conceder privilégios a outros usuários. Neste caso, seria praticamente um root, mas não poderia tomar a vaga dele e nem alterar seus privilégios.

Privilégios – Exemplos

- ▶ **GRANT select ON blog.* TO "diego"
IDENTIFIED BY "123456";**

Aqui, cria-se o usuário "diego" com a senha "123456" e já concedendo o privilégio de utilizar o SELECT dentro do BD "blog", em qualquer tabela ().*

Revogando Privilégios

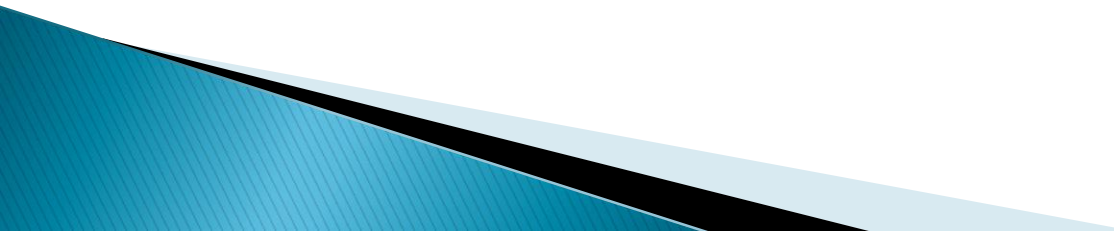
- ▶ Quando damos algum privilégio e ele precisa ser retirado depois, ou por um erro de concessão ou porque não é mais necessário o usuário ter aquele privilégio, podemos remover utilizar o comando REVOKE,

```
REVOKE priv [(colunas)] [, priv [(colunas)]] ...  
ON {*. * | db.* | db.tabela} FROM usuario [, user]  
...
```

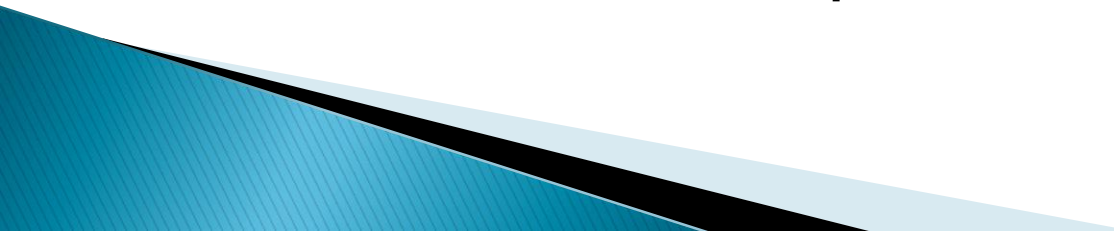
Revogando Privilégios

- ▶ Para revogar os privilégios concedidos anteriormente, na mesma ordem. Vejamos os exemplos abaixo:
- ▶ **REVOKE select ON blog.pessoas FROM "diego";**
- ▶ **REVOKE all, grant option FROM "diego";**
- ▶ **REVOKE select, insert, update ON blog.* FROM "diego";**
- ▶ **REVOKE all, grant option FROM "diego";**
- ▶ **REVOKE select ON blog.* FROM "diego";**

Verificando Privilégios

- ▶ Utilize o comando SHOW GRANTS FOR:
 - ▶ SHOW GRANTS FOR usuario;
 - ▶ Assim, será exibido toda a lista de privilégios que o determinado usuário tem, demonstrando os BDs e tabelas.
- 

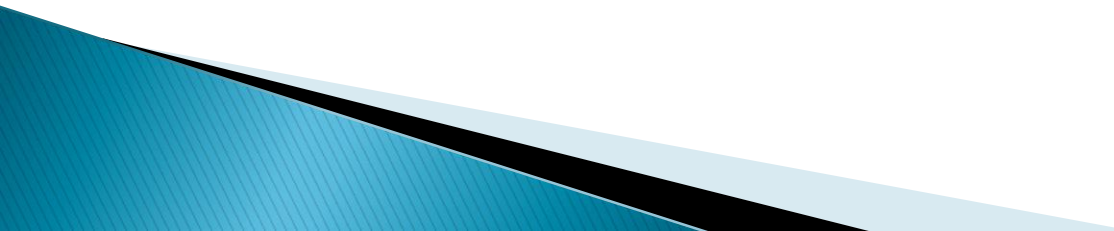
Limitando Recursos

- ▶ O gerenciamento de três recursos é introduzido no nível de usuário individual:
 - ▶ **Número de todas as consultas por hora:** Todos os comandos que podiam ser executados por um usuário.
 - ▶ **Número de todas as atualizações por hora:** Qualquer comando que altera qualquer tabela ou banco de dados.
 - ▶ **Número de conexões feitas por hora:** Novas conexões abertas por hora.
- 

Limitando Recursos

- ▶ Todos os usuários não são limitados por padrão no uso dos recursos acima, a menos que os limites sejam garantidos a eles.
- ▶ Estes limites podem ser concedidos **apenas através** do GRANT (*.*) global, usando esta sintaxe:
 - ▶ GRANT ... WITH
 - MAX_QUERIES_PER_HOUR N1
 - MAX_UPDATES_PER_HOUR N2
 - MAX_CONNECTIONS_PER_HOUR N3;

Limitando Recursos

- ▶ Se os usuários alcançam o limite de conexões dentro de uma hora, não será aceita mais nenhuma conexão até o fim desta hora.
 - ▶ De forma parecida se o usuário alcança o limite do número de consultas ou atualizações, consultas ou atualizações adicionais serão rejeitadas até que a hora acabe.
 - ▶ Em todos os casos, uma mensagem de erro apropriada é enviada.
- 

Backup e Restore no MySQL

- ▶ Vamos mostrar como realizar duas tarefas extremamente importantes: o backup (e a restauração) dos bancos de dados criados com MySQL.
- ▶ Para isso vamos usar o comando **mysqldump**, que será executado no prompt de comando, de acordo com a sintaxe a seguir (criando o arquivo de backup cujo nome você deverá especificar, com a extensão **.sql**):

```
mysqldump -u root -p nome_banco > arquivo_backup.sql
```

Backup e Restore

- ▶ Se o banco de dados apresentar algum problema e precisar ser restaurado a partir do backup, siga o procedimento a seguir:
- ▶ Crie um banco de dados novo no servidor, com o nome que desejar. Por exemplo, "teste-restore";
- ▶ Execute o seguinte comando (no prompt):
- ▶ **mysql -u root -p banco_criado < backup.sql**