

Consolidando o seu conhecimento

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

1) Quando você instalou MySQL, foi criado um usuário **root**, com privilégios de administrador. Mas, normalmente, este usuário é apagado e substituído por um administrador real.

2) No **Workbench**, clique na aba **Administration**.

3) Clique em **Users and Privileges**.

4) Você terá, do lado esquerdo, a lista de usuários do ambiente, na qual você pode ver o usuário **root**:

User	From Host
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost

5) Clique no botão **Add Account**.

6) Na caixa de diálogo, aba **Login**, preencha os campos **Login Name**, **Limit to Hosts Matching**, **Password** e confirme a senha:

Details for account newuser@%

Login Account Limits Administrative Roles Schema Privileges

Login Name: admin01 You may create multiple accounts with to connect from different hosts.

Authentication Type: Standard For the standard password and/or host select 'Standard'.

Limit to Hosts Matching: localhost % and _ wildcards may be used

Password: ***** Type a password to reset it.

Weak password.

Confirm Password: ***** Enter password again to confirm.

Expire Password

7) Na aba **Administrative Roles**, escolha o que este usuário pode fazer no MySQL. Selecione **DBA**, assim tudo será marcado:

Details for account newuser@%

Login Account Limits Administrative Roles Schema Privileges

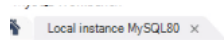
Role	Description
<input checked="" type="checkbox"/> DBA	grants the rights to pe
<input checked="" type="checkbox"/> MaintenanceAdmin	grants rights needed t
<input checked="" type="checkbox"/> ProcessAdmin	rights needed to asse
<input checked="" type="checkbox"/> UserAdmin	grants rights to create
<input checked="" type="checkbox"/> SecurityAdmin	rights to manage login
<input checked="" type="checkbox"/> MonitorAdmin	minimum set of rights
<input checked="" type="checkbox"/> DBManager	grants full rights on all
<input checked="" type="checkbox"/> DBDesigner	rights to create and re
<input checked="" type="checkbox"/> ReplicationAdmin	rights needed to setup
<input checked="" type="checkbox"/> BackupAdmin	minimal rights needed

Global Privileges

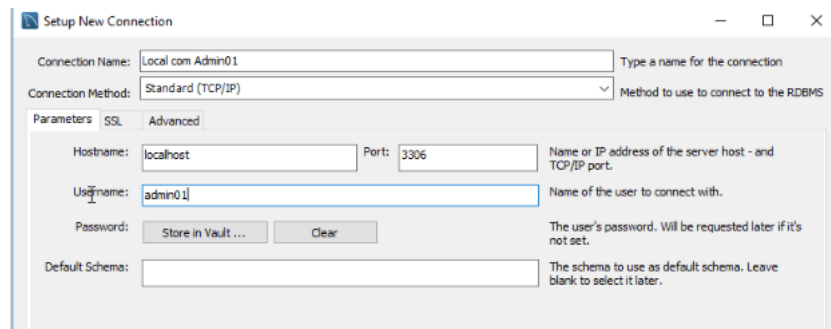
- ☒ ALTER
- ☒ ALTER ROUTINE
- ☒ CREATE
- ☒ CREATE ROUTINE
- ☒ CREATE TABLESPACE
- ☒ CREATE TEMPORARY TABLES
- ☒ CREATE USER
- ☒ CREATE VIEW
- ☒ DELETE
- ☒ DROP
- ☒ EVENT
- ☒ EXECUTE
- ☒ FILE
- ☒ GRANT OPTION
- ☒ INDEX
- ☒ INSERT

8) Clique em **Apply**, assim o usuário será criado.

9) Feche a aba da conexão do Workbench:



10) Você pode, na tela de conexões, criar uma nova, com o usuário criado nos passos anteriores:



11) Clique em **Test** e salve a conexão.

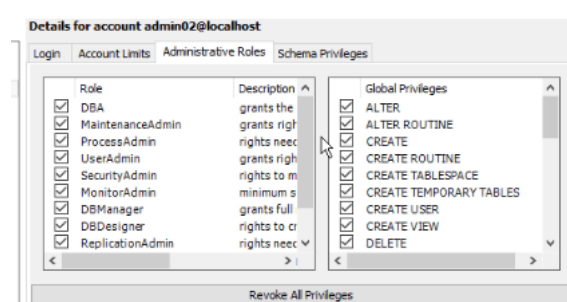
12) Entre no Workbench com o usuário criado nos passos anteriores.

13) O usuário poderia ter sido criado via SQL. Crie um outro usuário administrador (**admin02**). Digite:

```
CREATE USER 'admin02'@'localhost' identified BY 'admin02';
GRANT ALL PRIVILEGES ON *.* TO 'admin02'@'localhost' WITH GRANT OPTION;
```

14) Se você voltar à tela de **Users and Privileges** e executar um **Refresh**, você irá ver este novo usuário, com seus privilégios:

User	From Host
admin01	localhost
admin02	localhost
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost



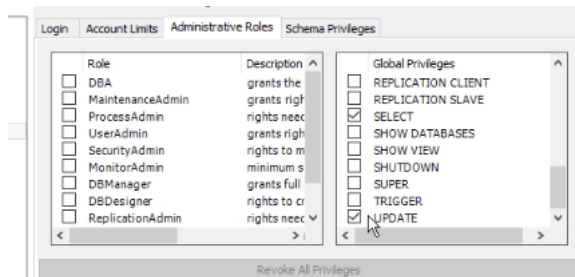
15) Para apagar o usuário **root** original, digite:

```
DROP USER 'root'@'localhost';
```

16) Se você tentar se conectar com usuário **root**, não irá mais conseguir, porque este usuário não existe mais.

17) O que vai determinar o que um usuário poderá fazer ou não, são seus parâmetros, tanto na caixa de diálogo do Workbench quanto via SQL.

18) Crie um usuário **user01**, usando a aba **Administration** e a opção **Users and Privileges**. Este usuário terá os privilégios: **CREATE TEMPORARY TABLES**, **DELETE**, **EXECUTE**, **INSERT**, **LOCK TABLES**, **SELECT** e **UPDATE**:

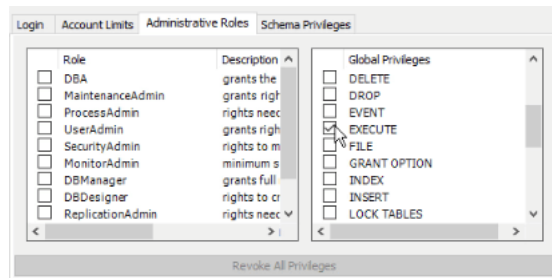


19) Clique em **Apply**.

20) Crie um outro usuário normal, mas agora via SQL. Digite:

```
CREATE USER 'user02'@'localhost' identified BY 'user02';
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE TEMPORARY TABLES,
LOCK TABLES, EXECUTE ON *.* TO 'user02'@'localhost';
```

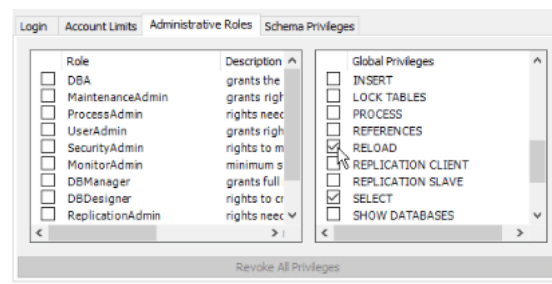
21) Crie um usuário somente leitura, com login **read01**, através da caixa de diálogo e o **read02**, via SQL. Para estes usuários, os privilégios serão **SELECT** e **EXECUTE**:



:

```
CREATE USER 'read02'@'localhost' identified BY 'read02';
GRANT SELECT, EXECUTE ON *.* TO 'read02'@'localhost';
```

22) Vá para mais dois usuários, com os logins **back01** e **back02**. Aqui, estes usuários somente podem criar backups:

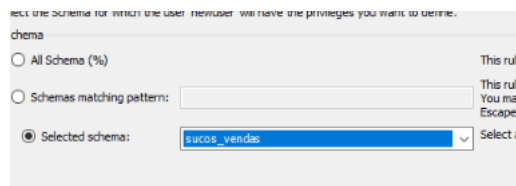


```
CREATE USER 'back02'@'localhost' identified BY 'back02';
GRANT SELECT, RELOAD, LOCK TABLES, REPLICATION CLIENT ON *.* TO 'back02'@'localhost';
```

23) Todos os usuários criados até aqui só podem acessar o banco de dados através da máquina **localhost**. Quando você cria um usuário, se você manter % ou _** no campo **Limit to Hosts Matching**, você irá determinar que outros IPs possam ser utilizados para acessar o banco. Estes caracteres funcionam como curinga:

```
CREATE USER 'admingeneric02'@'%' identified BY 'admingeneric02';
GRANT ALL PRIVILEGES ON *.* TO 'admingeneric02'@'%' WITH GRANT OPTION;
```

24) Você também pode limitar o acesso às bases e tabelas. Crie o usuário **user03**, mas, em vez de adicionar privilégios globais na aba **Administrative Roles**, acesse a aba **Schema Privileges** e adicione o esquema a ser acessado:

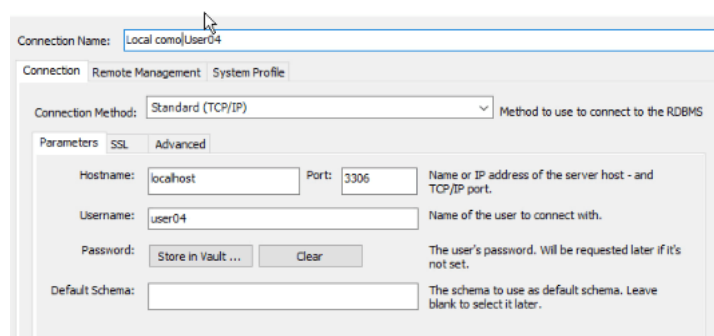


25) O mesmo comando pode ser digitado por SQL. Inclusive, abaixo, você pode criar o usuário simplesmente e depois adicionar, em outro comando, os privilégios:

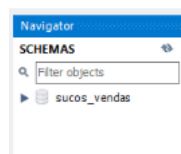
```
CREATE USER 'user04'@'%' IDENTIFIED BY 'user04';
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE
ON sucos_vendas.* TO 'user04'@'%';
```

O usuário criado acima somente pode ver a base **sucos_vendas**.

26) Crie uma conexão para o usuário **user04**:



27) Se conecte e note que somente a base **sucos_vendas** é disponibilizada para acesso:



28) Você pode limitar o acesso às tabelas, dando permissões do que pode ser feito nelas:

```
CREATE USER 'user05'@'%' IDENTIFIED BY 'user05';
GRANT SELECT, INSERT, UPDATE, DELETE
ON sucos_vendas.tabela_de_clientes TO 'user05'@'%';
GRANT SELECT
ON sucos_vendas.tabela_de_produtos TO 'user05'@'%';
```

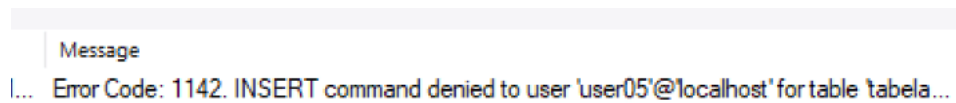
Estes comandos estão dando privilégios para inclusão, alteração, exclusão e consulta para o **user05** na **tabela_de_clientes**, mas somente leitura na **tabela_de_produtos**.

29) Crie uma conexão para o usuário **user05** e entre no Workbench.

30) Execute o comando:

```
INSERT INTO `sucos_vendas`.`tabela_de_produtos` (  
  `CODIGO_DO_PRODUTO`,  
  `NOME_DO_PRODUTO`,  
  `EMBALAGEM`,  
  `TAMANHO`,  
  `SABOR`,  
  `PRECO_DE_LISTA`  
) VALUES (  
  '999999',  
  'BNBNBNBNB',  
  'HJHJHJHJ',  
  'FGFGFGF',  
  'GHGHGH',  
  10  
);
```

Você verá erros ao tentar incluir um registro na tabela:



The screenshot shows a light gray message box with a title bar that says "Message". The text inside the box reads: "Error Code: 1142. INSERT command denied to user 'user05'@'localhost' for table 'tabela...'".

31) Existe um comando para verificar os usuários existentes:

```
SELECT * FROM mysql.user;
```

32) Há também um comando que mostra os acessos de um usuário, por exemplo:

```
SHOW GRANTS FOR 'user02'@'localhost';
```

33) O comando `REVOKE ALL` retira os privilégios de acesso dos usuários:

```
REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'user02'@'localhost';
```