**MYSQL – SEGURANÇA E CONTROLE DE ACESSO**

LUCAS LAMOUNIER GONCALVES DUARTE - 2016012688

KEVIN VIEIRA PEREIRA - 2016015385

RODRIGO APARECIDO SILVA MAIA - 2016013095

Itajubá, 20 de março de 2018

**LISTA DE FIGURAS**

[Figura 1 - Criando usuários. 3](#_Toc509315154)

[Figura 2 - Concedendo permissões aos usuários 6](#_Toc509315155)

[Figura 3- UserOne acessando a base de dados. 6](#_Toc509315156)

[Figura 4- UserOne fazendo uma consulta na tabela produto. 7](#_Toc509315157)

[Figura 5- UserTwo acessando a base de dados. 7](#_Toc509315158)

[Figura 6- Exibindo colunas da tabela user. 8](#_Toc509315159)

[Figura 7- Listando os usuários do SGBD 8](#_Toc509315160)

[Figura 8- Listando privilégios do usuários. 9](#_Toc509315161)

[Figura 9- Removendo as permissões do userOne 9](#_Toc509315162)

[Figura 10- Criando View. 9](#_Toc509315163)

[Figura 11- Concedendo permissão para userOne acessar a View recém criada. 9](#_Toc509315164)

[Figura 12- Acesso do usuário userTwo a base de dados. 10](#_Toc509315165)

[Figura 13- Criando usuário com especificação de host. 10](#_Toc509315166)

[Figura 14- Criando role. 11](#_Toc509315167)

[Figura 15- Atribuido permissões para a role. 11](#_Toc509315168)

[Figura 16- Atribuindo role para os usuários. 11](#_Toc509315169)

[Figura 17 - Espaço ocupado por cada backup. 14](#_Toc509315170)

[Figura 18– Comandos para o backup físico. 14](#_Toc509315171)

[Figura 19– Comandos para backup lógico. 15](#_Toc509315172)

**QUESTÃO 1: Estudar e praticar o controle de acesso de usuário em seu banco.**

No MySQL os usuários são identificados pelo seu nome e host de acesso dessa forma sempre que for criado um usuário novo você deve atribuir o host de onde ele acessará. Como consequência dessa política do MySQL sempre que você quiser atribuir um novo host de acesso a um usuário existente, um novo usuário com mesmo nome deve ser criado com o host desejado especificado.

Para que o usuário possa acessar a base dados de qualquer host pode ser especificado como host o símbolo **“%”** que significa todos.

Para criar um usuário no MySQL a seguinte sintaxe deve ser utilizada: **CREATE USER ‘< usuário >’ @’ < host de acesso >’ IDENTIFIED BY ‘< senha >’;** Caso o host não seja especificado por padrão o símbolo **“%”** será atribuído ao usuário recém criado.

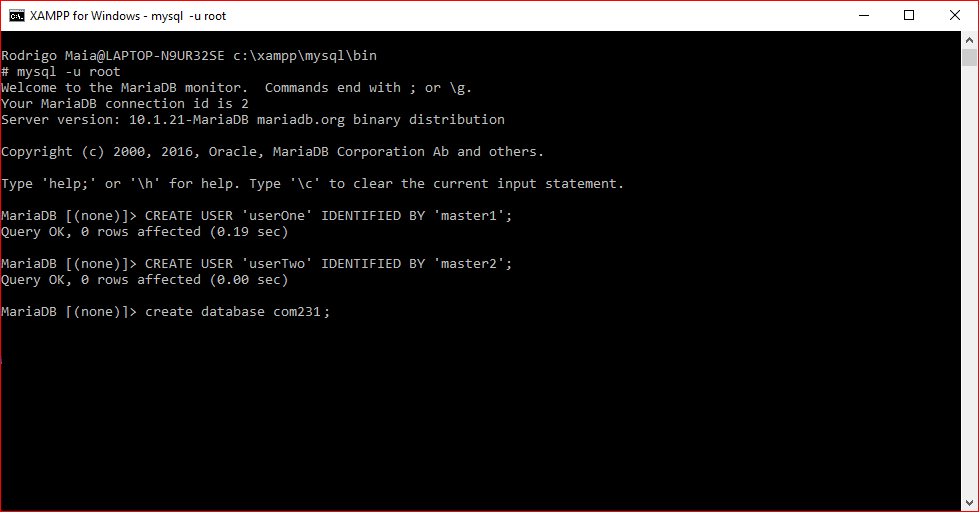


Figura 1 - Criando usuários.

**QUESTÃO 2: Estudar e praticar a concessão de privilégios em seu banco.**

A concessão de privilégios no MySQL assim como os demais SGBDs é efetuada através do comando **GRANT** na seguinte sintaxe: **GRANT <lista de tipos de privilégio > (<colunas privilegiadas>) ON < objeto alvo> TO ‘<usuário>’ @ ‘<host>’ WITH < propriedades >**, onde:

* Lista de tipos de privilégio – São os privilégios que serão concedidos ao usuário, podendo ser:

Tabela 1 - Privilégios disponíveis no MySQL.

|  |  |
| --- | --- |
| PRIVILÉGIOS | DESCRIÇÃO |
| [ALL [PRIVILEGES]](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_all) | Concede todos os privilégios no nível de acesso especificado, exceto GRANT OPTION e PROXY. |
| [ALTER](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_alter) | Permite ao usuário o uso de ALTER TABLE. Níveis: Global, banco de dados, tabela. |
| [ALTER ROUTINE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_alter-routine) | Permite ao usuário alterar e deletar rotinas armazenadas. Níveis: Global, banco de dados, procedimento. |
| [CREATE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create) | Permite ao usuário a criação de banco de dados e de tabelas. Níveis: Global, banco de dados, tabela. |
| [CREATE ROUTINE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create-routine) | Permite ao usuário a criação de rotina armazenada. Níveis: Global, banco de dados. |
| [CREATE TABLESPACE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create-tablespace) | Permite ao usuário a criação, alteração ou remoção de espaços de tabela e grupos de arquivos de log. Nível: Global. |
| [CREATE TEMPORARY TABLES](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create-temporary-tables) | Permite ao usuário o uso de CREATE TEMPORARY TABLE. Níveis: Global, banco de dados. |
| [CREATE USER](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create-user) | Permite ao usuário o uso de CREATE USER, DROP USER, RENAME USER e REVOKE ALL PRIVILEGES. Nível: Global. |
| [CREATE VIEW](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_create-view) | Permite ao usuário a criação e alteração de views. Níveis: Global, banco de dados, tabela. |
| [DELETE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_delete) | Permite ao usuário o uso de DELETE. Nível: Global, banco de dados, tabela. |
| [DROP](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_drop) | Permite ao usuário a exclusão de bancos de dados, tabelas e views. Níveis: Global, banco de dados, tabela. |
| [EVENT](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_event) | Permite ao usuário o uso de eventos para o Agendador de eventos. Níveis: Global, banco de dados |
| [EXECUTE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_execute) | Habilita o usuário para executar rotinas armazenadas. Níveis: Global, banco de dados, tabela. |
| [FILE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_file) | Habilita o usuário à efetuar no servidor leitura ou escrita de arquivos. Nível: Global. |
| GRANT OPTION | Permite que privilégios sejam concedidos ou removidos de outras contas. Níveis: Global, banco de dados, tabela, procedimento, proxy. |
| [INDEX](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_index) | Permite ao usuário a crição e remoção de índices. Níveis: Global, banco de dados, tabela. |
| [INSERT](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_insert) | Permite ao usuário o uso de INSERT. Níveis: Global, banco de dados, tabela, coluna. |
| [LOCK TABLES](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_lock-tables) | Permite ao usuário uso de LOCK TABLES em tabelas para as quais ele tenha o privilégio SELECT. Níveis: Global, banco de dados. |
| [PROCESS](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_process) | Permite que o usuário veja todos os processos com SHOW PROCESSLIST. Nível: Global. |
| [PROXY](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_proxy) | Habilita o proxy do usuário. Nível: De usuário para usuário. |
| [REFERENCES](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_references) | Permite ao usuário a criação de chaves estrangeiras. Níveis: Global, banco de dados, tabela, coluna. |
| [RELOAD](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_reload) | Permite ao usuário o uso de operações FLUSH. Nível: Global. |
| [REPLICATION CLIENT](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_replication-client) | Permite que o usuário pergunte onde são os servidores mestre ou escravo. Nível: Global. |
| [REPLICATION SLAVE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_replication-slave) | Permite ao usuário criar escravos de replicação para ler eventos de log binário do mestre. Nível: Global. |
| [SELECT](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_select) | Permite ao usuário o uso de SELECT. Níveis: Global, banco de dados, tabela, coluna. |
| [SHOW DATABASES](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_show-databases) | Permite ao usuário a visualização de todos os bancos na base de dados. Níveis: Global. |
| [SHOW VIEW](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_show-view) | Permite ao usuário o uso de SHOW CREATE VIEW. Níveis: Global, banco de dados, tabela. |
| [SHUTDOWN](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_shutdown) | Permite ao usuário o desligamento da base dados. Nível: Global. |
| [SUPER](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_super) | Permite o uso de outras operações administrativas, como o comando CHANGE MASTER TO, KILL, PURGE BINARY LOGS, SET GLOBAL e mysqladmin debug. Nível: Global. |
| [TRIGGER](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_trigger) | Permite ao usuário o uso de operações de disparo. Níveis: Global, banco de dados, tabela. |
| [UPDATE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_update) | Permite ao usuário o uso de UPDATE. Níveis: Global, banco de dados, tabela, coluna. |
| [USAGE](https://dev.mysql.com/doc/refman/5.7/en/privileges-provided.html#priv_usage) | Sinônimo de sem privilégios. |

* Colunas privilegiadas – São as colunas que o usuário obterá os privilégios especificado anteriormente, podendo ser eles SELECT, UPDATE e etc. Obs.: Apenas utilizadas em privilégios a nível de tabela.
* Objeto alvo – Determina o nível do privilegio, que pode ser Global; banco de dados; tabela; coluna, e o alvo, nome de uma tabela, banco ou etc..
* Usuário e host – Determina o usuário que receberá os privilégios especificados anteriormente.
* Propriedades – Em geral determina limites sobre a concessão especificada anteriormente. Segui a tabela abaixo das opções:

Tabela 2- Propriedades disponiveis no MySQL

|  |  |
| --- | --- |
| OPÇÕES | DESCRIÇÃO |
| ****MAX\_QUERIES\_PER\_HOUR**** | Máximo de execuções por hora. |
| ****MAX\_UPDATES\_PER\_HOUR**** | Máximo de atualizações por hora. |
| ****MAX\_CONNECTIONS\_PER\_HOUR**** | Máximo de conexões por hora. |
| GRANT OPTION | Permitir ao usuário conceder a outros usuários o privilégio especificado anteriormente. |

Assim como a concessão de privilégios é efetuada pelo comando **GRANT** a remoção é efetuada pelo comando **REVOKE** que pode ser utilizado com a seguinte sintaxe: **REVOKE <lista de tipos de privilégio > (<colunas privilegiadas>) ON < objeto alvo> FROM ‘<usuário>’ @ ‘<host>’.**

Após a execução de ambos os comandos GRANT e REVOKE deve se atualizar o banco os privilégios que estão em memória, isso é possível através do comando **FLUSH PRIVILEGES.**

**Imagens dos conceitos anteriores sendo aplicado no banco:**

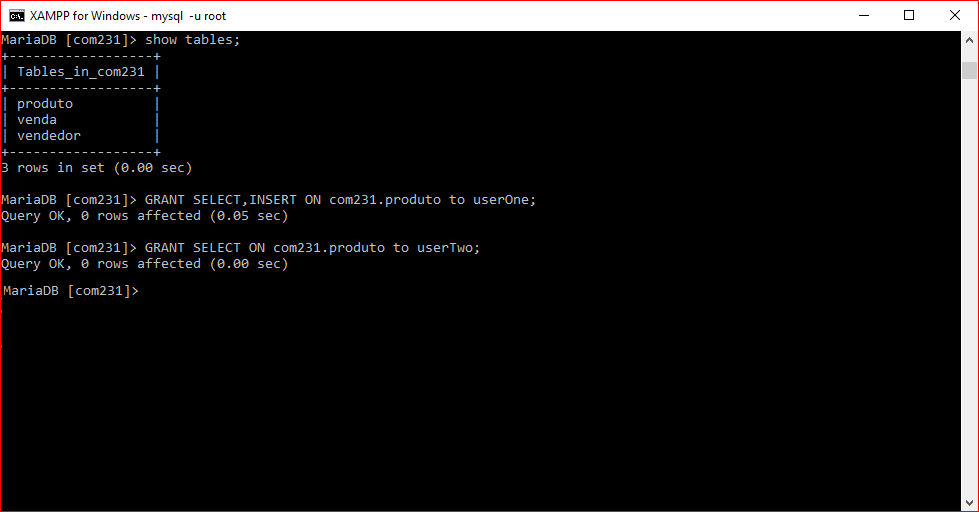


Figura 2 - Concedendo permissões aos usuários

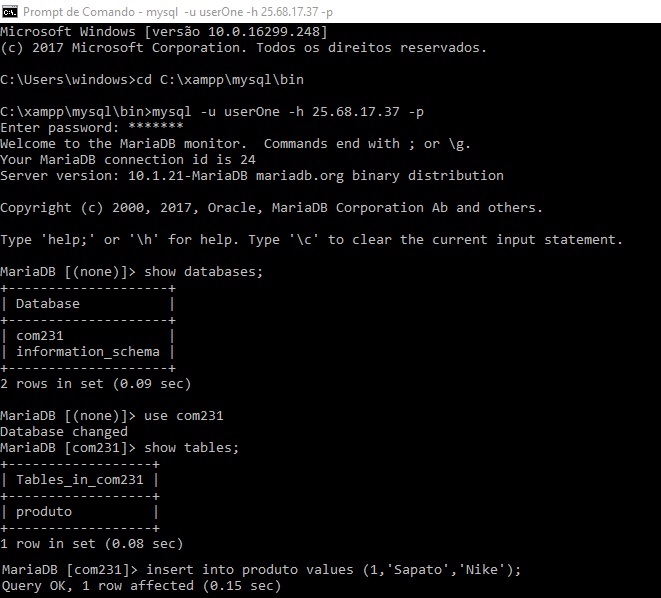


Figura 3- UserOne acessando a base de dados.

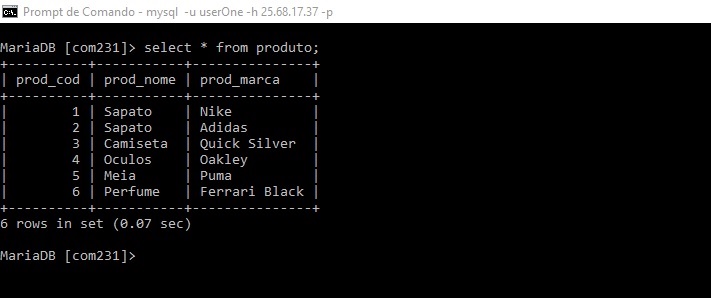
****

Figura 4- UserOne fazendo uma consulta na tabela produto.

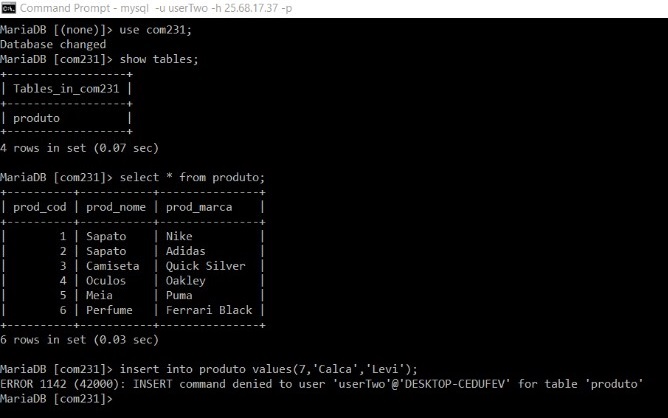
****

Figura 5- UserTwo acessando a base de dados.

**QUESTÃO 3: Como listar os usuários cadastrados no banco?**

Os usuários do MySQL são armazenados fisicamente em uma tabela de nome **“user”** no banco de dados **“mysql”** presente por padrão no SGBD, dessa forma todos os usuários podem ser listado através de um comando SELECT na tabela **user**.

A tabela **user** também armazena privilégios globais concedidos aos usuários, então é aconselhável fazer um filtro sobre as colunas desejadas que no caso desse relatório serão as colunas **host, user, e password.**

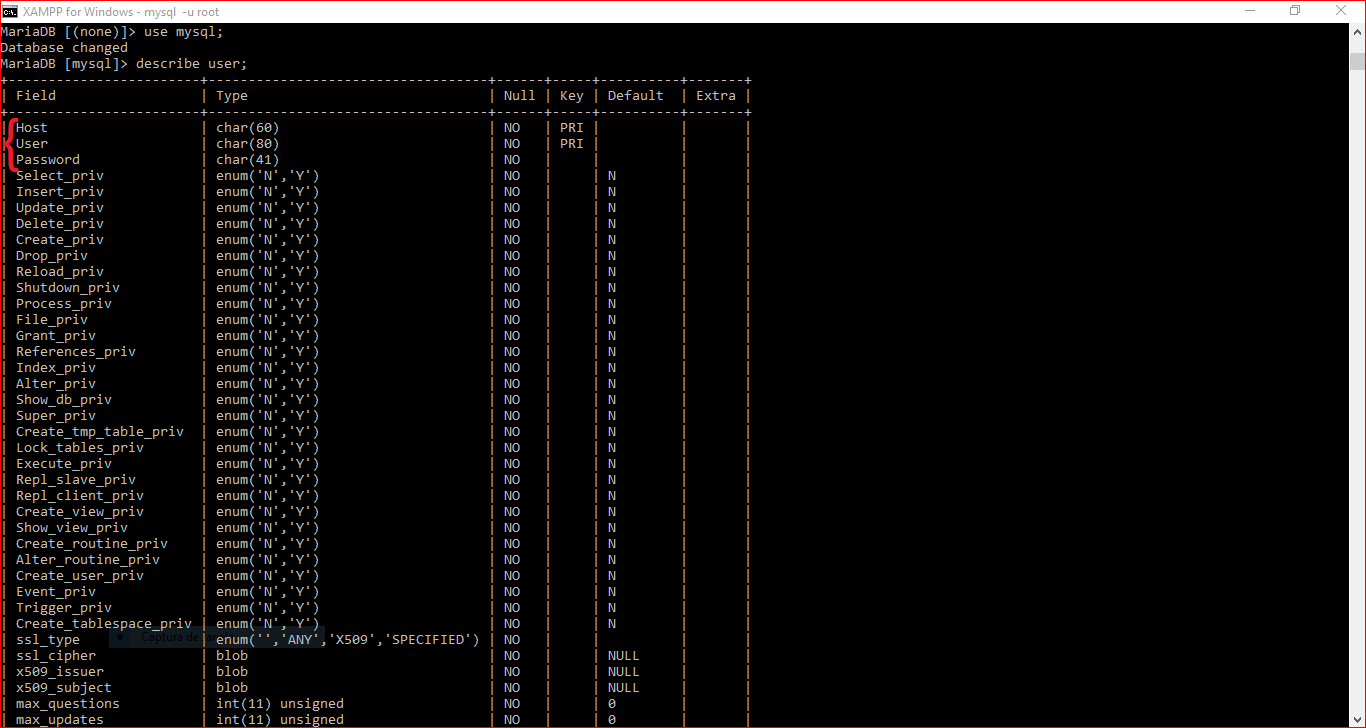


Figura 6- Exibindo colunas da tabela user.

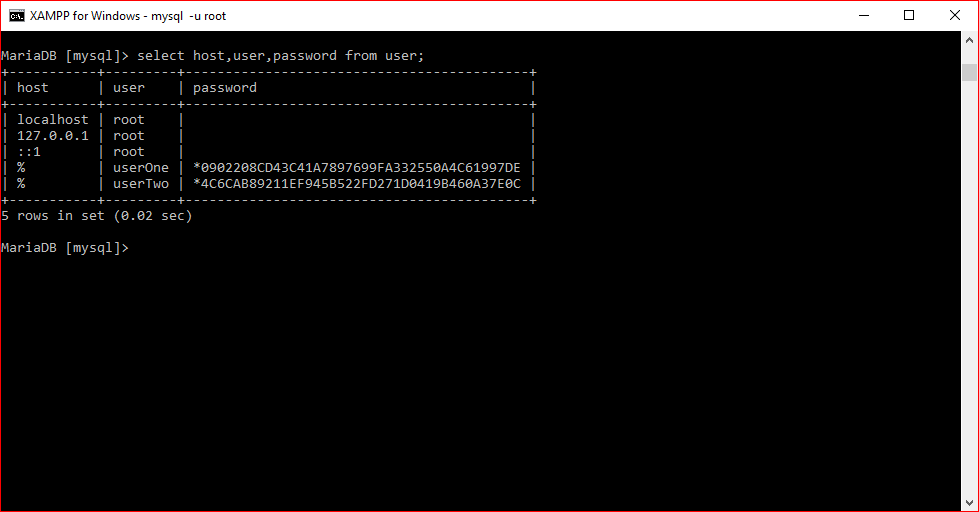


Figura 7- Listando os usuários do SGBD

**QUESTÃO 4: Como listar os privilégios dos usuários cadastrados no banco?**

Como dito anteriormente os privilégios globais estão armazenados na tabela **user** juntamente com seus usuários, enquanto os privilégios a nível de tabela estão presente na tabela **“tables\_priv”**, com essas informações seria possível construir uma consulta que mostrasse todos os privilégios de um usuário no entanto o MySQL já disponibiliza uma ferramenta para essa função, executando o comando **show GRANTS for < usuários> @ < host >.**

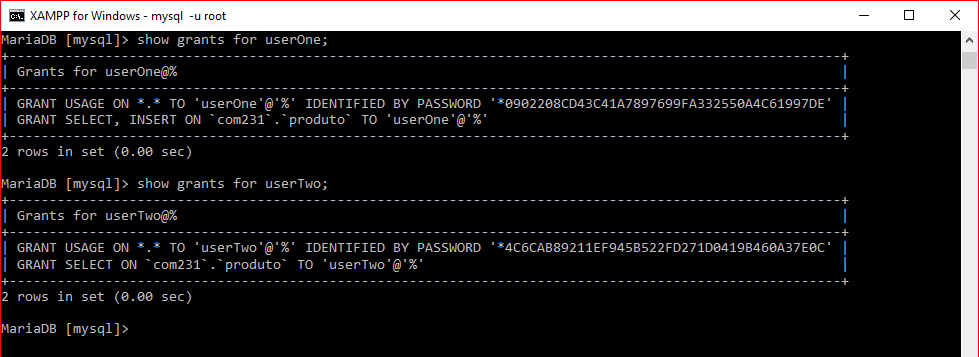
****

Figura 8- Listando privilégios do usuários.

**QUESTÃO 5: Criar uma view no banco. Dar privilégios para o usuário apenas na view e verificar como o banco de dados é apresentado para esse usuário. O que ele pode ver?**

1. O usuário utilizado para responder essa questão será o userTwo, por isso incialmente será removido todos os seus privilégios.

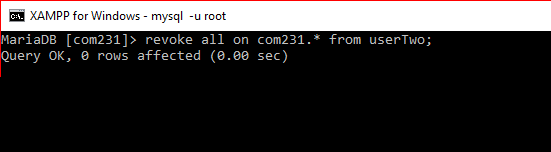


Figura 9- Removendo as permissões do userOne

1. Agora será criada a view a ser utilizada, que será uma view construída a partir de uma junção entre as tabelas de vendas, produto e vendedor.

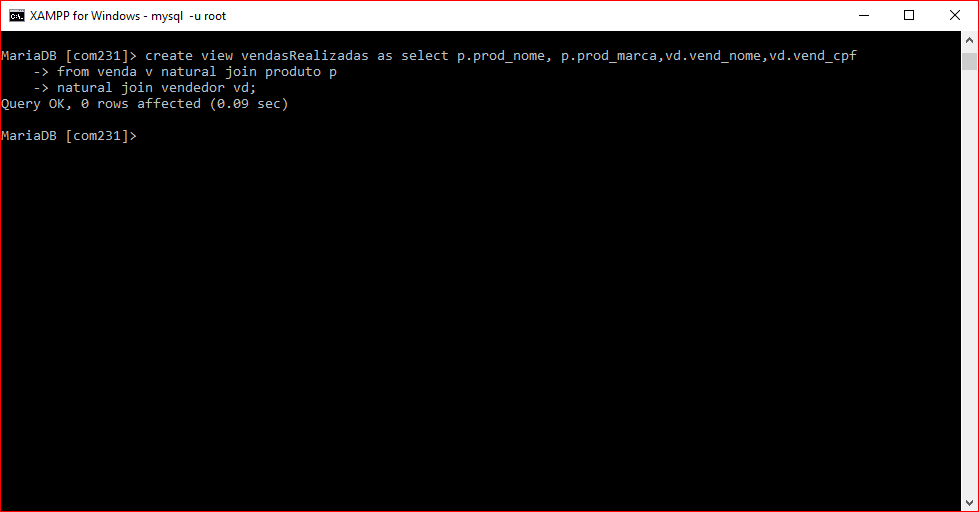


Figura 10- Criando View.

1. Concedendo permissão sobre a view para o usuário userTwo.

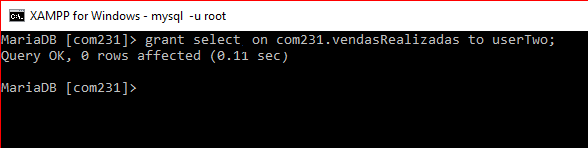


Figura 11- Concedendo permissão para userOne acessar a View recém criada.

1. Acesso do usuário userTwo a base de dados.

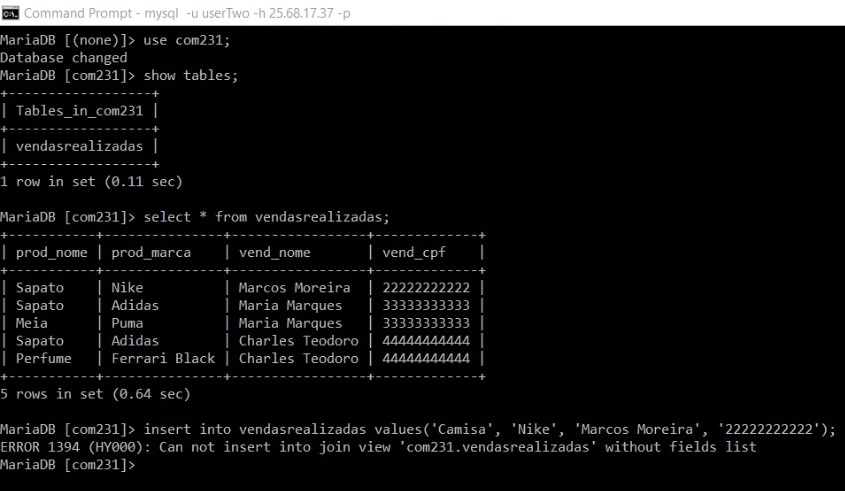


Figura 12- Acesso do usuário userTwo a base de dados.

**QUESTÃO 6: Verificar:**

1. **O SGBD permite configurar acessos em diferentes redes? Por exemplo, se o usuário estiver na rede da empresa, ele tem determinados privilégios. Caso contrário, ele possui outros privilégios.**

Sim, o SGBD permite que configurações de acesso para diferentes hosts sejam efetuadas para os usuários, no entanto como referido na questão 1, o MySQL trata como identificador o host de acesso e o nome do usuário, dessa forma para esse tipo de configuração deve ser criado um novo usuário com o mesmo nome e senha, mas com um host diferente.

Na questão 1 foi criado usuários que possui acesso a todos os hosts nessa questão será criado um novo usuário com especificações de host.

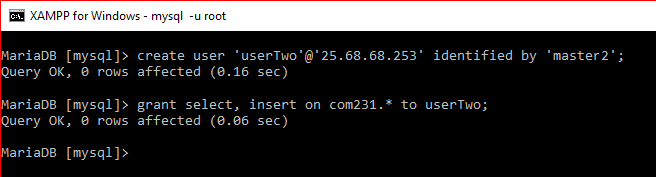


Figura 13- Criando usuário com especificação de host.

A partir de agora o usuário userTwo através da máquina de IP 25.68.68.253 é capaz de selecionar e inserir em todas as tabelas do banco com231, enquanto em outras maquinas ele apenas é capaz de visualizar a view criada na questão 5.

**b) O SGBD permite configurar privilégios para grupos de usuários? Como?**

Sim, o SGBD permite que o root configure privilégios tanto para usuários específicos, como para grupos de usuários. Essa segunda opção é denominada “Roles”. A seguir é apresentado um exemplo de uma Role sendo criada e definindo privilégios para os usuários que estiverem dentro dela.

Primeiro vamos criar a Role:

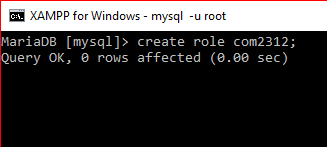


Figura 14- Criando role.

Em seguida ao invés de atribuirmos privilégios a um usuário, vamos atribuir estes privilégios a Roler com2312 que criamos no passo anterior.

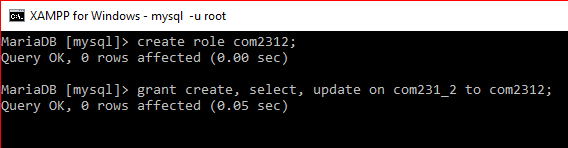


Figura 15- Atribuido permissões para a role.

Agora que temos uma Role criada e com os seus privilégios definidos vamos criar um usuário modelo para inserirmos neste grupo com2312.

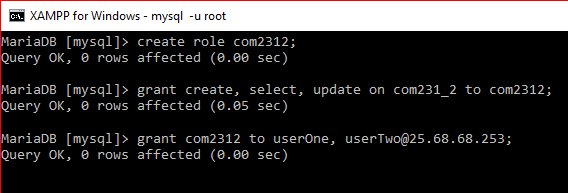


Figura 16- Atribuindo role para os usuários.

Da mesma maneira que atribuímos privilégios para uma Role, podemos remover utilizando o comando REVOKE na Role com2312, que os usuários perdem os privilégios automaticamente.

**c) Veja o exemplo: o usuário deve ter acesso a todas as tabelas de um banco, exceto a tabela ‘x’. Neste caso, é possível dar privilégios ao banco de dados e posteriormente remover o privilégio apenas da tabela x?**

Na versão do MySQL que está sendo utilizado pelo grupo não é permitido esse tipo de operação.

**QUESTÃO 7: Uma boa prática de segurança em qualquer SGBD é manter sempre o backup atualizado. Verifique como é o sistema de backup do banco. Quais opções ele dá? É possível agendar? É possível salvar o backup como binário? Faça um teste e verifique a diferença de tamanho dos arquivos**

**Tipos de backups:**

**Backup Físico:** Consiste em copiar os diretórios e arquivos que contém os conteúdos do banco de dados, é recomendado para base de dados importantes que necessitam ser recuperados rapidamente quando um problema ocorre;

Comando do Backup: no terminal de comando, executar o comando para se encontra o MySQL, no caso desse computador é necessário executar o comando **cd C:\xampp\mysql\bin** primeiro, após isso pode-se executar o comando:

**FLUSH TABLES listaDeTabelas WITH READ LOCK**; este comando faz com que as tabelas estejam disponíveis apenas para a leitura e não escrita, possibilitando assim a cópia física das tabelas sem risco de inserção de dados; existe a variação nesse comando que permite salvar em binário o backup;

\*Pode ser usado o comando **SELECT \* INTO OUTFILE 'nomeDoArquivo' FROM nomeDaTabela**, este comando salvará a tabela escolhida no formato de texto, contento apenas os dados e não a estrutura da tabela. Salvará apenas no servidor o arquivo, e para a realização do backup do banco inteiro será necessário realizar o comando para cada tabela, apesar de ser lento para ser feito, a recuperação do banco pode chegar a ser 20 vezes mais rápida que qualquer outro método;

**Backup Lógico:** Salva informações representadas de maneira lógica pelo banco, como os comandos usados na base de dados, **create table, insert**; mais adequado para bancos em que possa haver a necessidade de se editar o conteúdo;

Comando do Backup: no terminal de comando, executar o comando para se encontra o MySQL, no caso desse computador é necessário executar o comando **cd C:\xampp\mysql\bin** primeiro, após isso pode-se executar o comando:

**mysqldump -u nomeDoUsuario -p nomeDoBanco> nomeDoArquivo.sql**, depois é necessário fornecer a senha do banco;

**Backup Incremental**: consistem em salvar apenas as mudanças dado um ponto no tempo, para ser possível isso é necessário habilitar o **binary** **log** do servidor, que permite que o servidor salve a data das mudanças e quais comandos foram realizados;

Comando do Backup: depois de ativado o **binary log,** será necessário escrever os logs em um arquivo para que o MySQL seja capaz de ler, então para tal será usado o comando no terminal de comandos **mysqlbinlog binlog.001 > Localizacao/nomeDoArquivo.sql** e depois será executado o comando para realizar tal backup **mysql -u nomeDoUsuário -p -e "Localizacao/nomeDoArquivo.sql";**

**Snapshot**: esse tipo permite que seja o feito o backup lógico dado em um ponto no tempo, sem realizar a cópia física do sistema inteiro; é disponibilizado por terceiros como Veritas, LVM e ZFS;

**Opções para o Backup:**

**Online**: backup pode ser realizado quando o servidor está funcionando, é menos intrusivo para os clientes, pois como o servidor vai estar funcionando ainda, será possível para os mesmos acessar os dados da base de dados;

**Offline**: nesse caso o servidor dever estar parado para ser possível realizar o backup, os clientes podem ser afetados, caso desejem ter acesso aos dados;

**Local**: o backup é iniciado no mesmo host em que a aplicação MySQL está sendo executado;

**Remoto**: é possível iniciar o backup em um host diferente do que contém a aplicação;

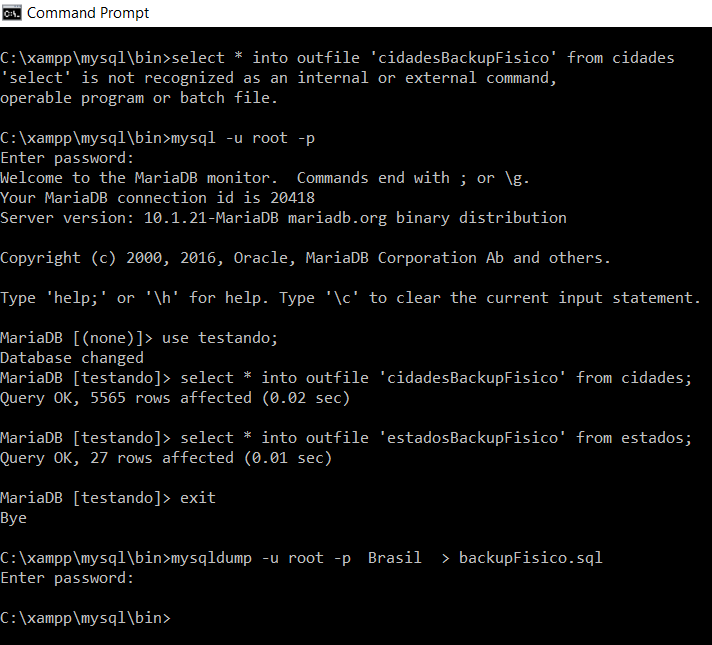
\*As opções de Agendamento, Criptografar, e comprimir o backup, está disponível apenas para a versão MySQL Enterprise Backup, a versão empresarial do MySQL; terceiros podem possibilitar tais ações também;

**Diferenças de Tamanhos:**

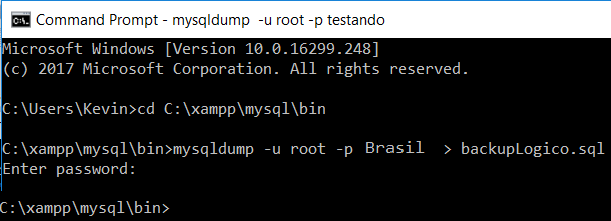
Foi realizado tanto o backup lógico, como o físico, neste se foi necessário realizar o backup de cada tabela e depois da estrutura do banco. O banco de dados utilizado era composto de 2 tabelas uma com todas as cidades do Brasil e o outro de todos os estados, pode-se perceber que o backup lógico consumiu 365KB, enquanto o físico 329KB (considerando os 3 itens dele, cidadesBackupFisico(325KB), estadosBackupFisico(1KB) e o backupFisico(3KB)), em questão de tempo o lógico foi mais rápido por ter de se realizar apenas um comando. Nas figuras abaixo apresentam-se as informações da quantidade de espaço ocupado por cada arquivo, e os comandos para realizar cada backup; ambos os backup foram realizados de maneira local, e não foi realizado o backup incremental pois o mesmo é necessário realizar o físico e depois para cada operação realizada se salva os conteúdos, como a única operação pedida era a de comparação de tamanho, então ele possuiria o mesmo tamanho que o backup físico.



Figura 17 - Espaço ocupado por cada backup.



*Figura 18– Comandos para o backup físico.*



*Figura 19– Comandos para backup lógico.*

**REFERÊNCIA**:

MySQL 5.7 Reference Manual, Oracle, 2018. Disponível em: <https://dev.mysql.com/doc/refman/5.7/en/data-types.html >. Acesso em 18/03/2018.