**MySql**

**Nomes**: LUCAS LAMOUNIER GONCALVES DUARTE - 2016012688

KEVIN VIEIRA PEREIRA - 2016015385

RODRIGO APARECIDO SILVA MAIA – 2016013095

Itajubá, 07 de março de 2018

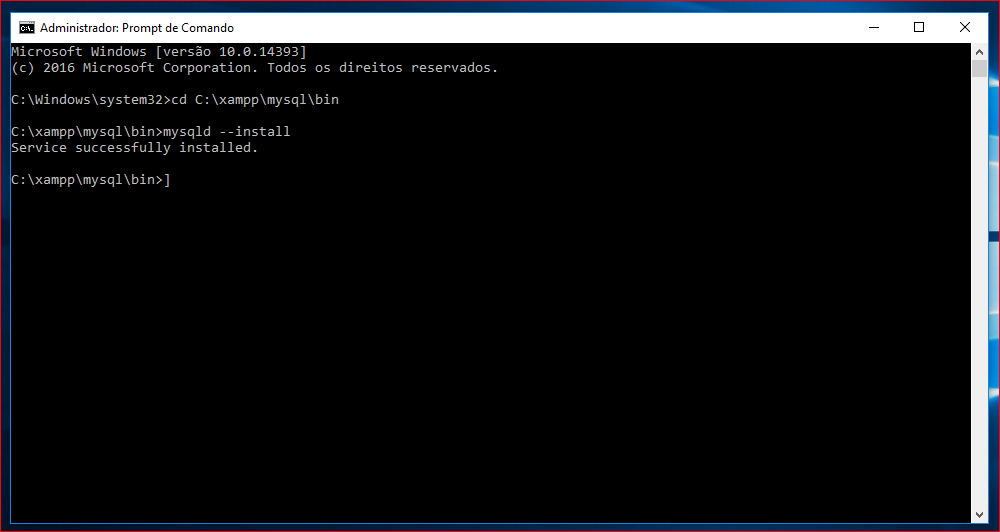
**Questão 1:** Estudar como se configura a arquitetura cliente-servidor em seu SGBD e implementá-la. Para tanto, um membro do grupo deverá ser servidor e os demais clientes.

O MySql é um SGBD relacional de código aberto que se faz presente em vários pacotes de ferramentas para servidores como XAMPP e WAMP. Nesse relatório será abordado uma instância do MySql que foi instalada juntamente com o pacote de ferramentas XAMPP, dessa forma algumas das configurações de instalação já foram pré-definidas.

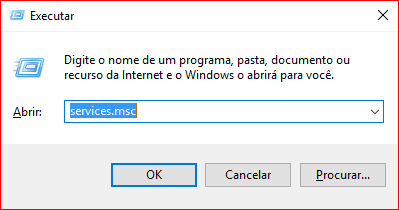
**Configuração do Servidor (Rodrigo)**

Quando MySql é instalado através do Xampp ele é iniciado a através do painel do mesmo, então inicialmente deve se cadastrar o MySql como serviço do S.O que está sendo utilizando no servidor para que ele se iniciei assim que a máquina seja ligada (Nosso caso é o Windows 10 64bits).

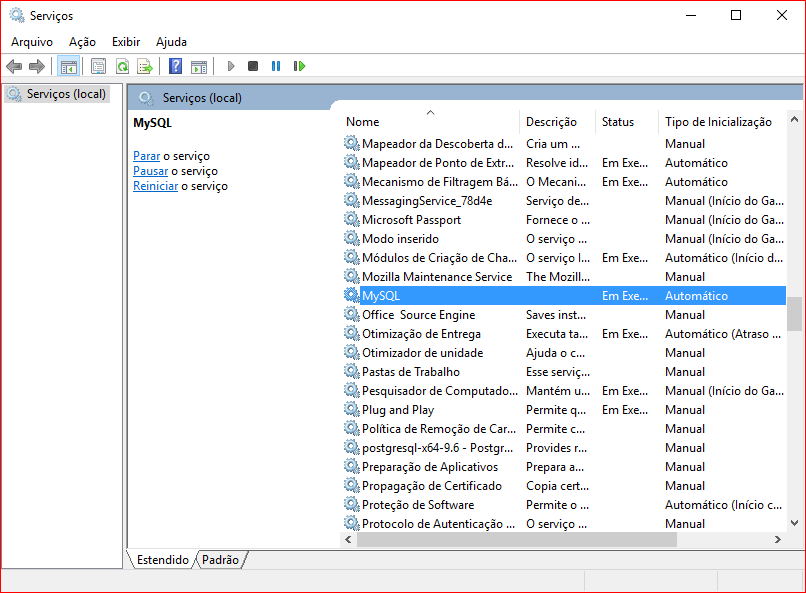
1. Abrindo o terminal do Windows seguir até a pasta onde se encontra os arquivos do MySql no servidor e em seguida através da linha ***mysqld –install***, instalar o serviço Mysql no Windows.



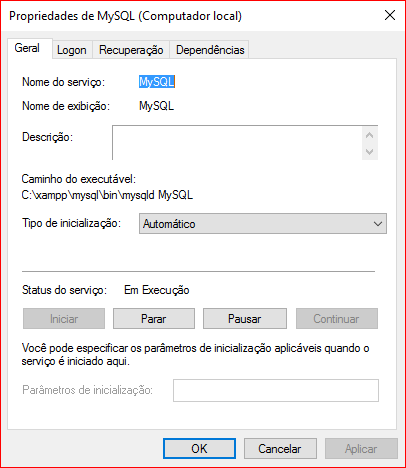
1. Em seguida verificar se instalação foi concluída com êxito, acessado a página de serviços do Windows através do comando executável **services.msc.**



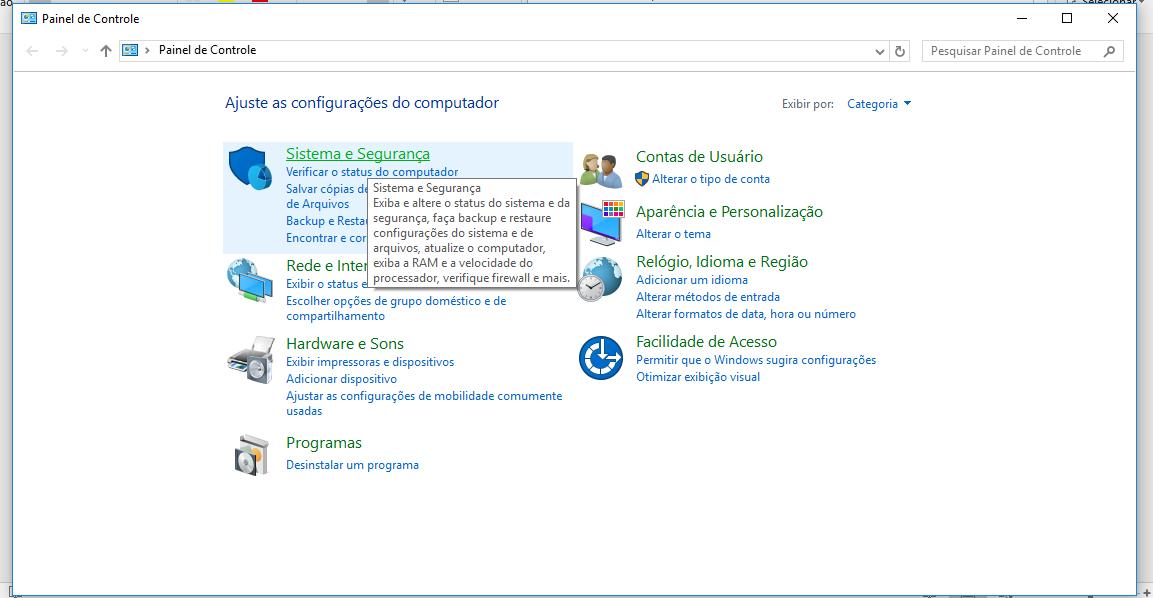
1. Na página de serviços do Windows buscar o serviço chamado **MySql** e verificar se na coluna **Tipo de inicialização** está definido o valor **Automático**.

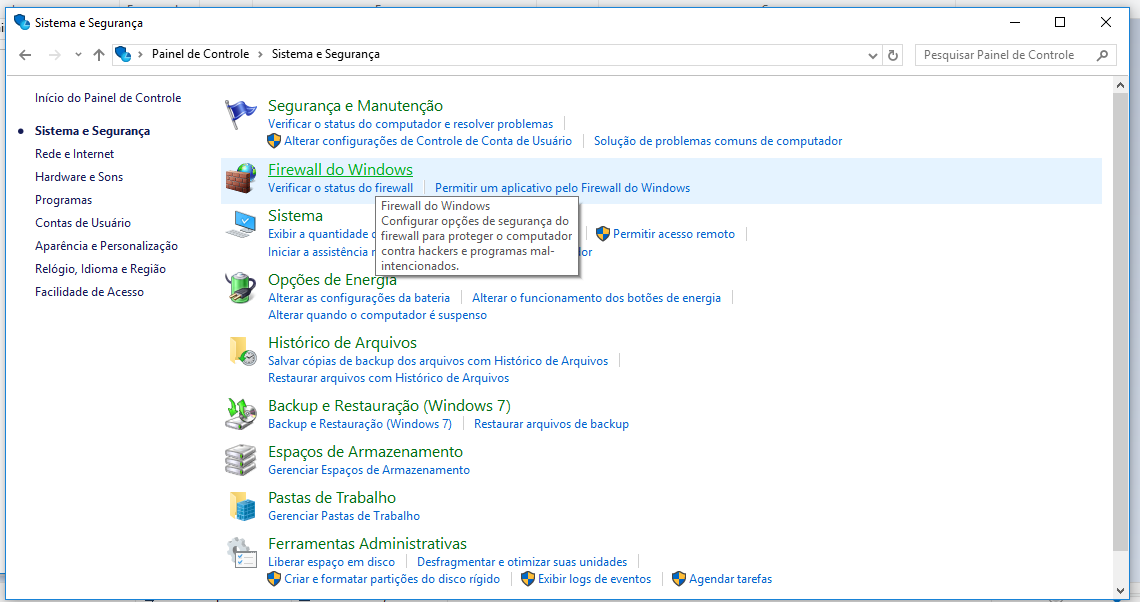


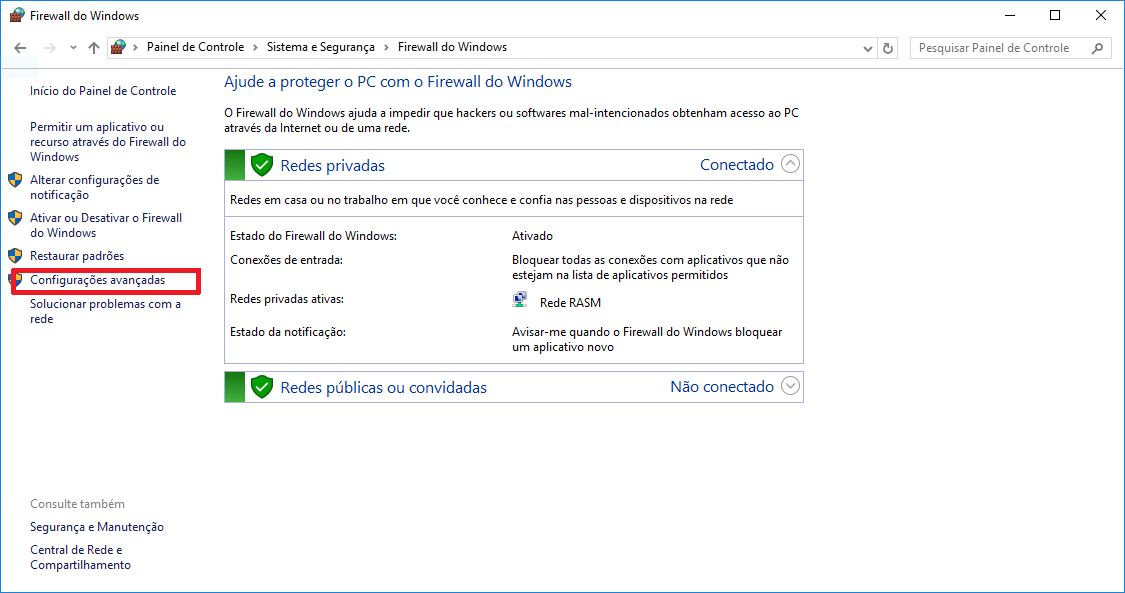
* 1. Caso não esteja definido automático, clique duas vezes sobre o serviço, será aberta a janela de **Propriedades do Serviço MySql** na aba **Geral**, em **Tipo de inicialização** definir o valor **Automático**.



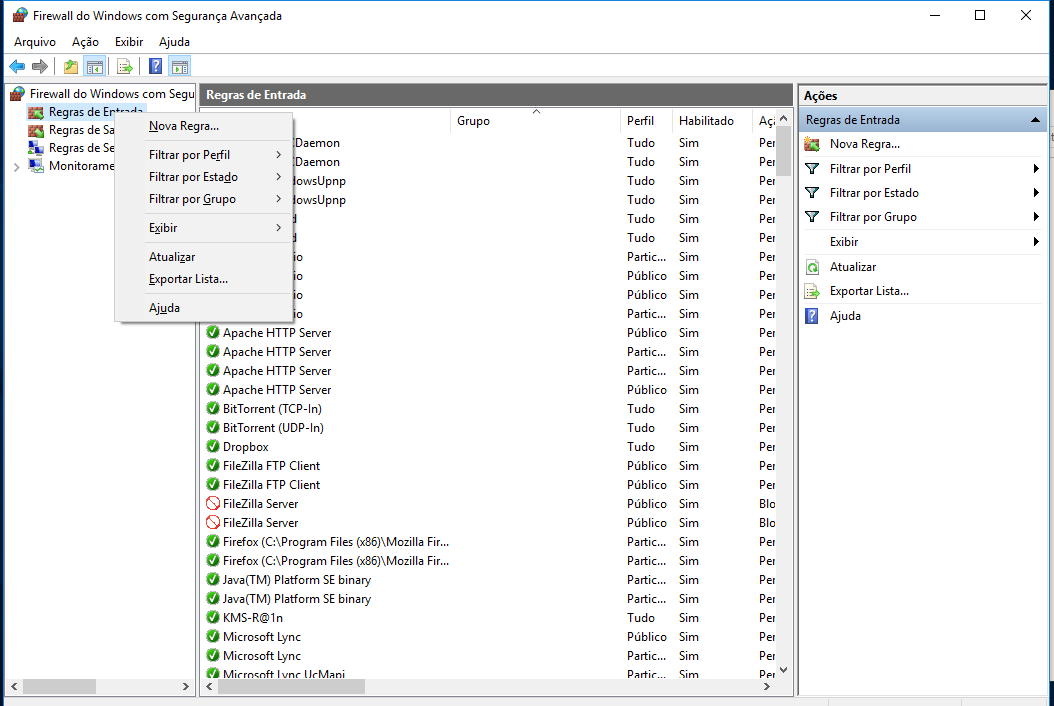
1. Reinicie o Windows para que as configurações sejam aplicadas.
2. No painel de controle do Windows em **Sistema e Segurança**, na opção **firewall do Windows** ir em **Configurações avançadas.**

****

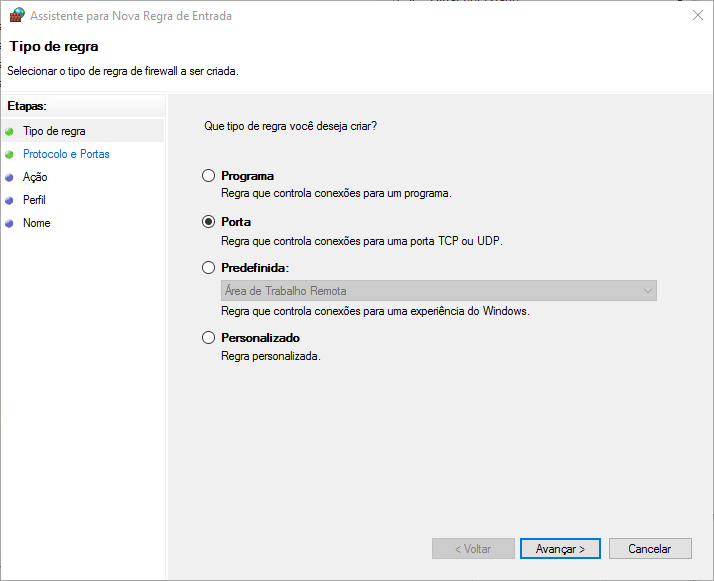
****

****

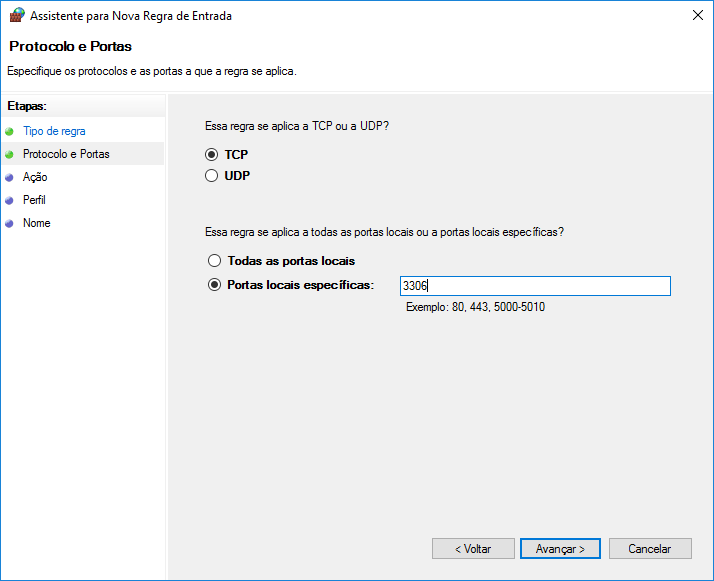
1. No menu esquerdo da janela clicar sobre **Regras de Entrada** e escolher a opção **Nova regra**.



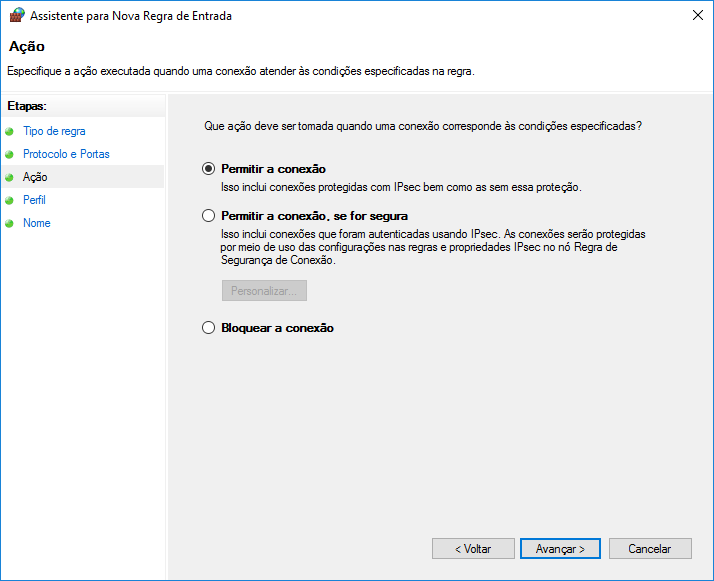
1. Na janela que será aberta pelo procedimento acima na primeira aba **Tipo de regra** escolher a opção **Porta**.



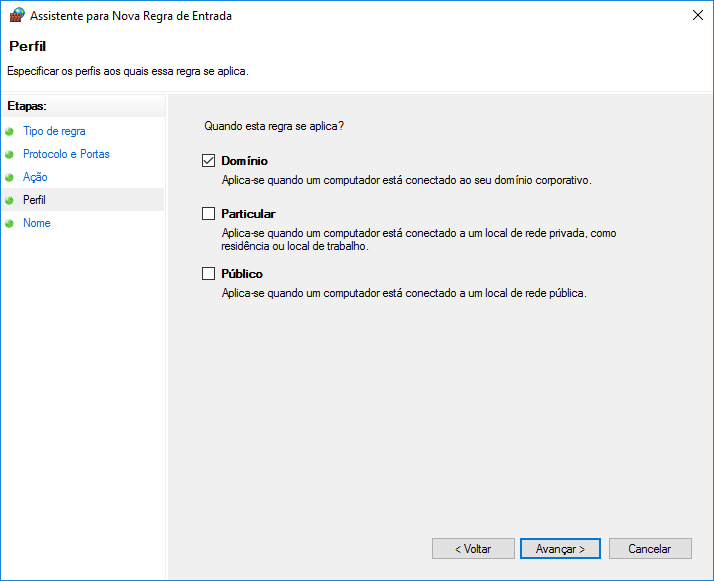
* 1. Na aba **Protocolos e Portas** escolher a opção **TCP** em “**Essa regra se aplica a TCP ou UDP?”** e em “**Essa regra se aplica a todas as portas locais ou portas locais especificas?”** escolher a opção **Portas locais especificas** e preencher no campo de texto o número da porta a ser liberada, **3306.**



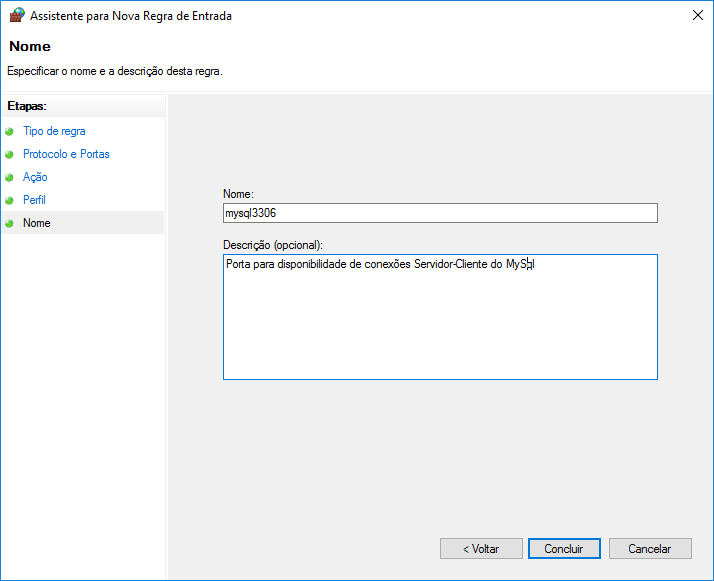
* 1. Na Aba **Ação** definir a opção **Permitir conexão**.



* 1. Na aba **Perfil** manter selecionado apenas a opção **Domínio.**

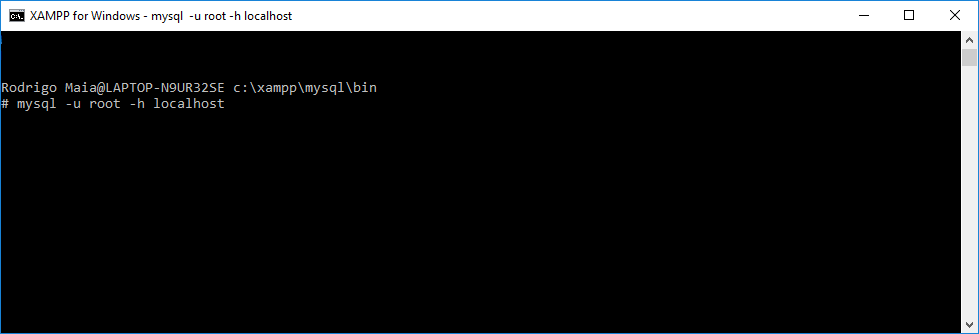


* 1. Na aba **Nome** colocar um nome sugestivo com uma descrição sucinta.

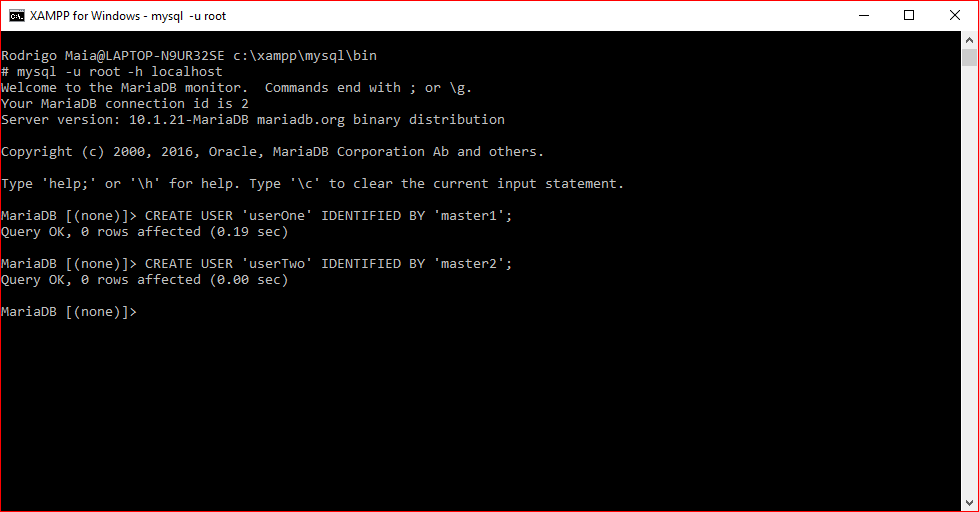


1. Efetuar o mesmo procedimento para **Regras de Saída.**
2. Seguir ao diretório onde os arquivos do MySql estão presentes, no caso desse computador é o caminho **C:\xampp\mysql\bin,** e logar na base dados do MySql presente no próprio servidor através da linha de comando **mysql – u <usuário> –h localhost,** onde **–u** representa o usuário com qual será efetuado o login e o **–h** o local onde está presente a base de dados, mas como esse comando estão sendo efetuado no servidor será colocado como link para a base de dados a frase **localhost** que aponta para o próprio servidor**.**

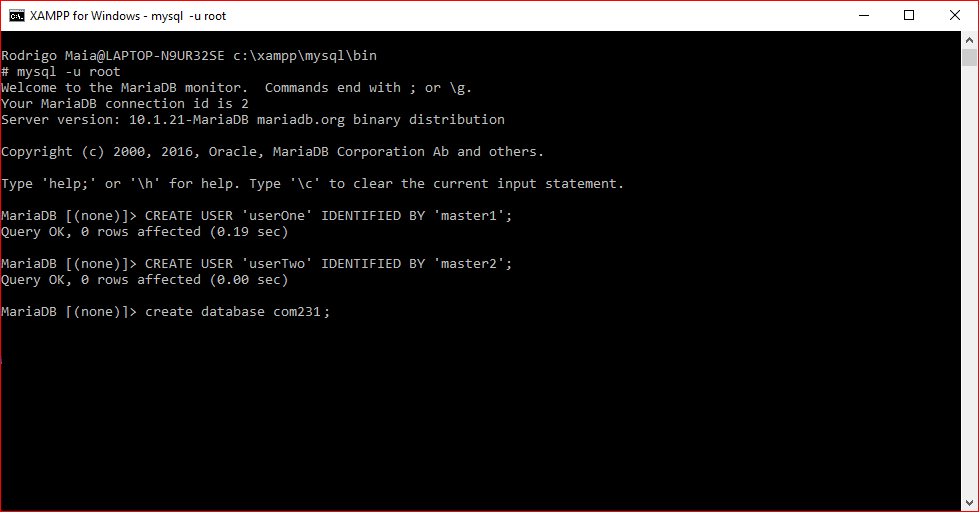
Por padrão do XAMPP o MySql vem com usuário root já cadastrado no SGBD e ele não possui senha.

****

1. Após efetuado o login na base de dados, será criando dois novos usuários para que possam ser utilizados pelos clientes. Através da linha SQL **“CREATE USER '<novousuario>'@'localhost' IDENTIFIED BY '<senha>';”**, serão criados os dois novos usuários.

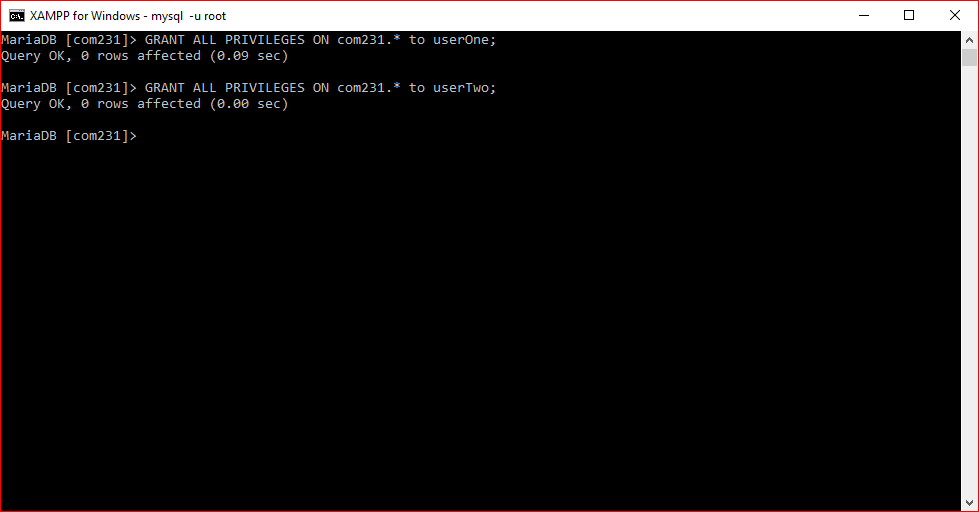


1. Agora será criando um banco de dados com nome **com231** para ser usado pelos dois usuários recém-criado.

****

1. Conceder permissões aos usuários para que possam acessar, consultar e modificar o banco **com231**.

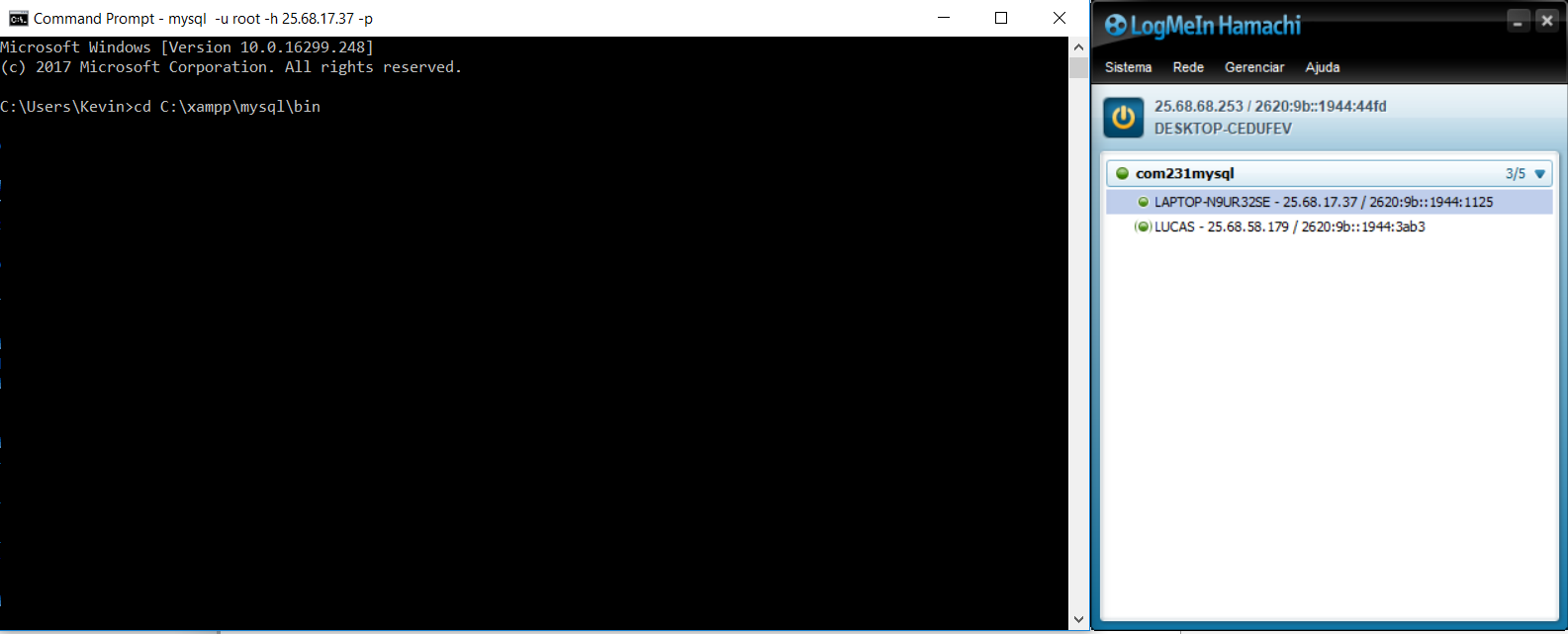
Essa concessão será efetuada pela linha SQL **“GRANT ALL PRIVILEGES ON <NomedoBanco>.<NomedaTabela> to <usuário>; “,** onde o nome do banco será **com231,** o nome da tabela receberá o valor **“\*”** para simbolizar que o usuário obterá permissões para todas as tabelas do banco e no final será especificado o usuário que receberá essas permissões.



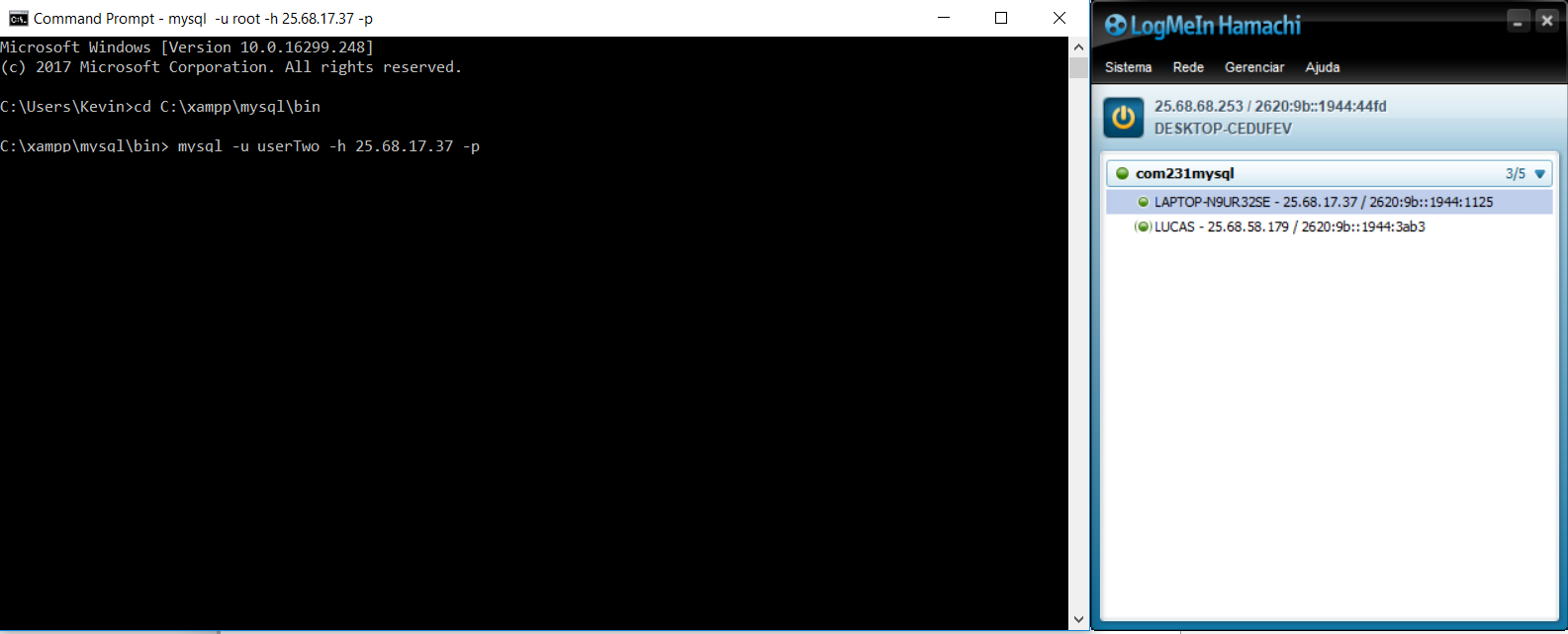
**Configuração do cliente**

**Cliente 1 (Kevin):**

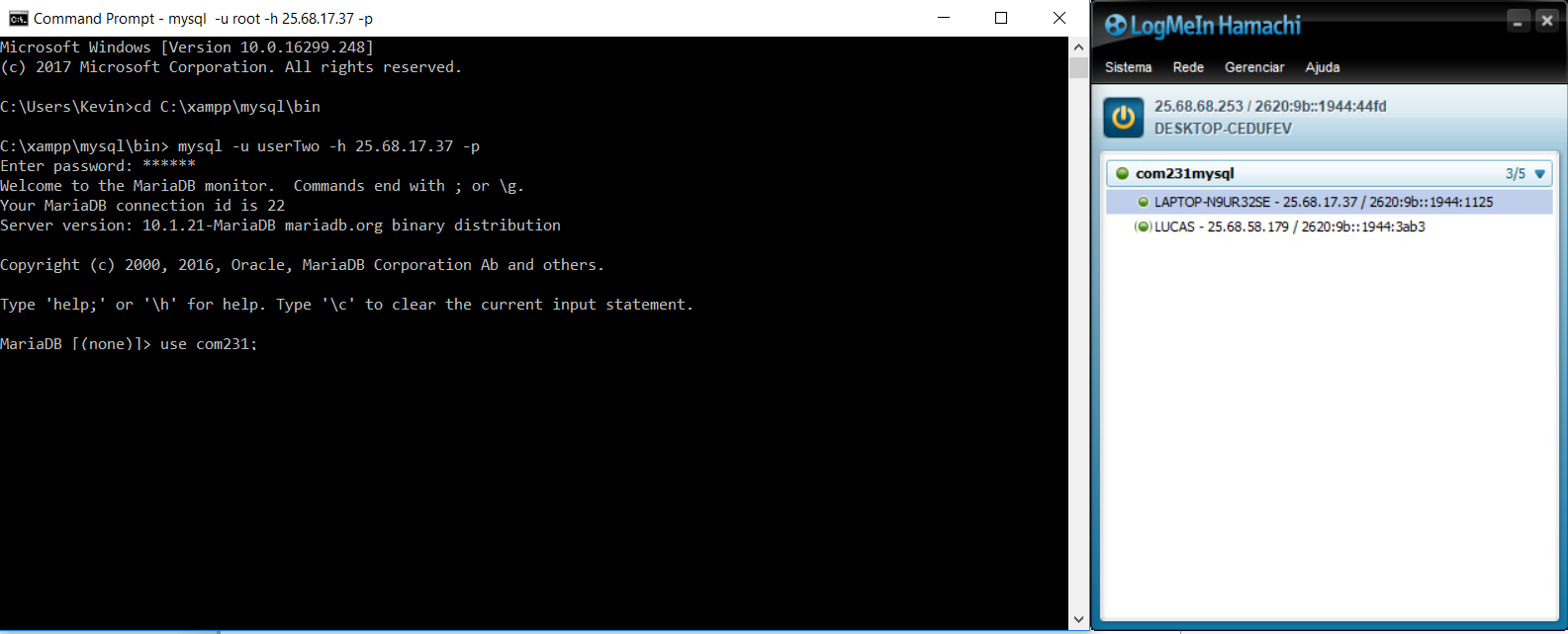
1. Primeiro é necessário abrir o terminal de comando do Xampp, então utiliza-se o comando **cd** mais o caminho necessário para chegar na pasta bin contida em MySql, que se encontra no Xampp, sendo que este caminho pode variar de computador para computador. No caso deste computador o comando é **cd C:\xampp\mysql\bin**, após aplicar este comando será possível continuar o procedimento para ser configurado como cliente. A janela do programa Hamachi encontra-se ao lado para demonstrar que este computador está conectado com o computador que está funcionando como servidor;



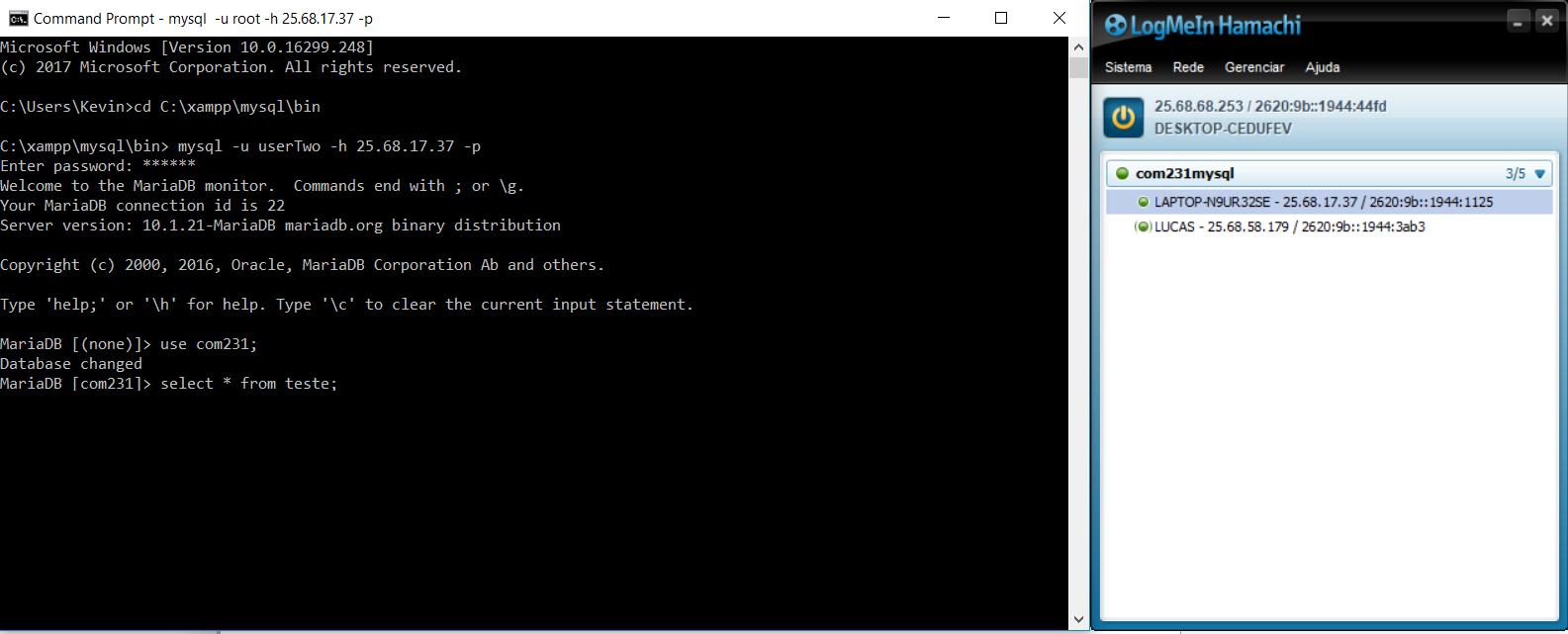
1. Então utiliza-se o comando **mysql -u (nome do usuário) -h (o ip do servidor) -p** para conectar com o computador que está funcionando como servidor;



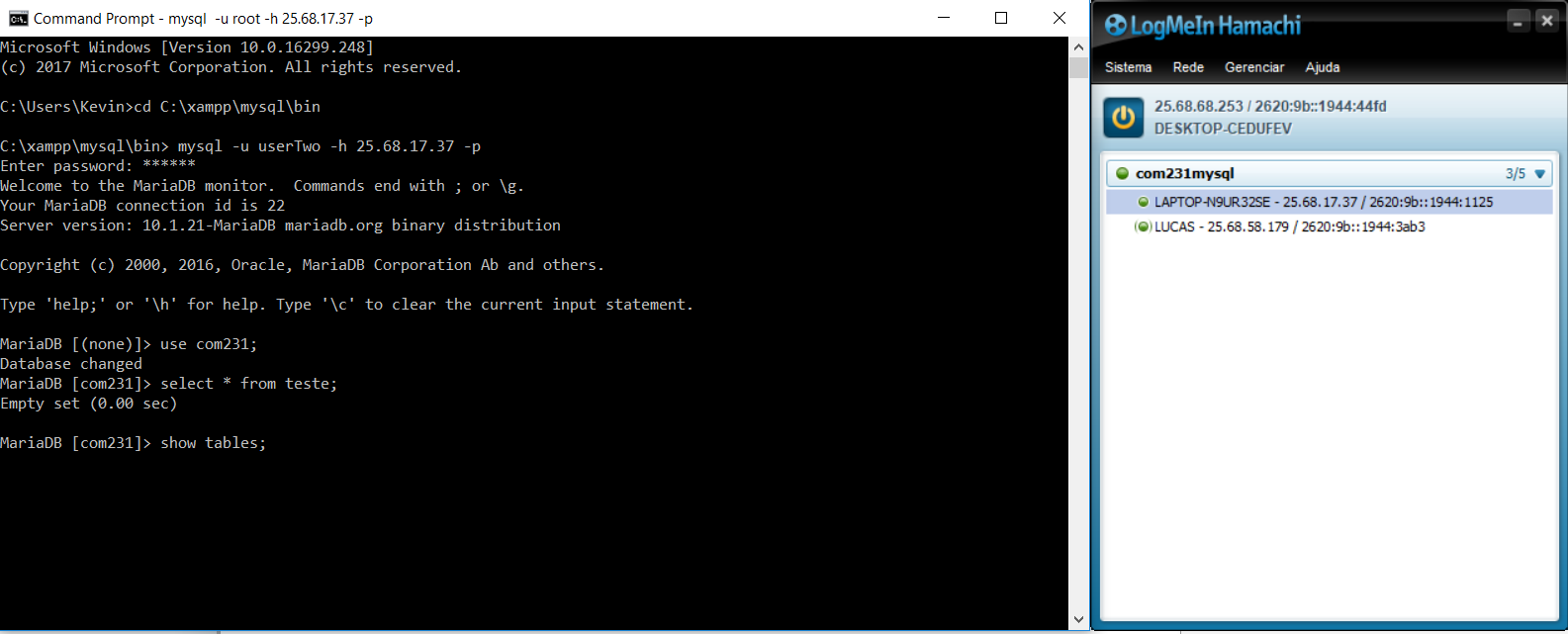
1. Depois é necessário fornecer a senha do servidor;



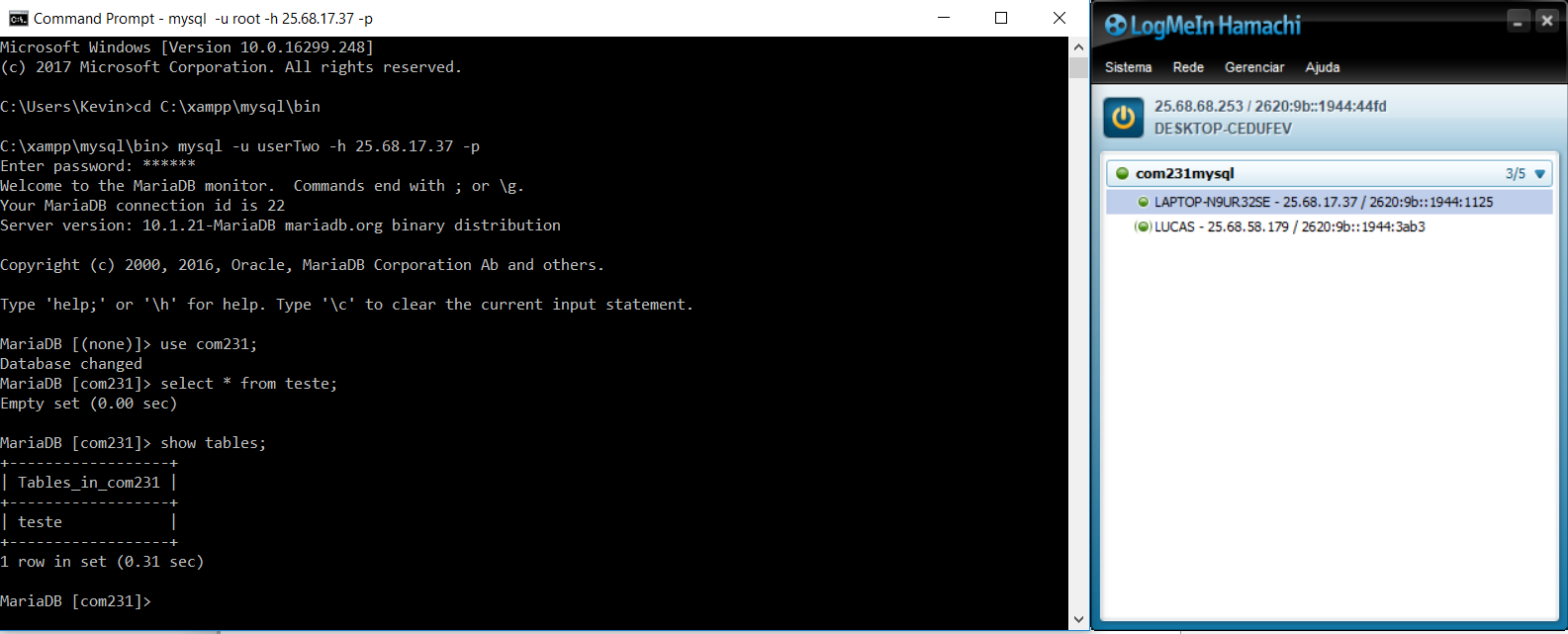
1. Então para utilizar um banco de dados desse servidor é necessário fornecer o nome da base de dados, no caso desta demonstração chama-se com231;



1. Após selecionado o banco de dados pode-se utilizar todos os comandos que o MySql possibilita, no caso foi utilizado o comando **select** para mostrar que a tabela teste foi criada anteriormente, como resultado do **select** foi mostrado que não possui nenhum registro na tabela teste;

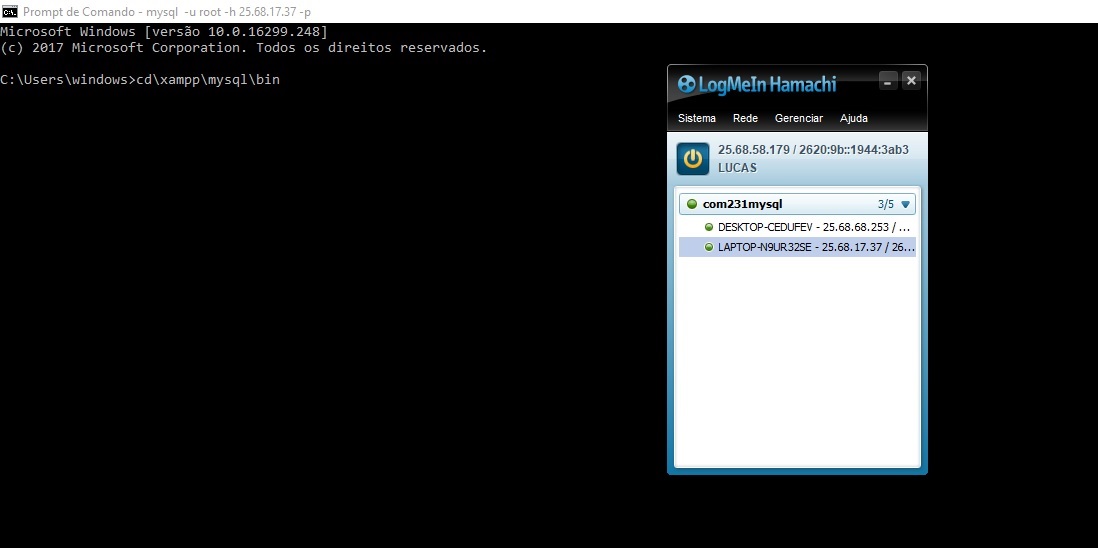


1. Logo após foi utilizado o comando **show tables** para mostrar que a tabela teste existe na base de dados;

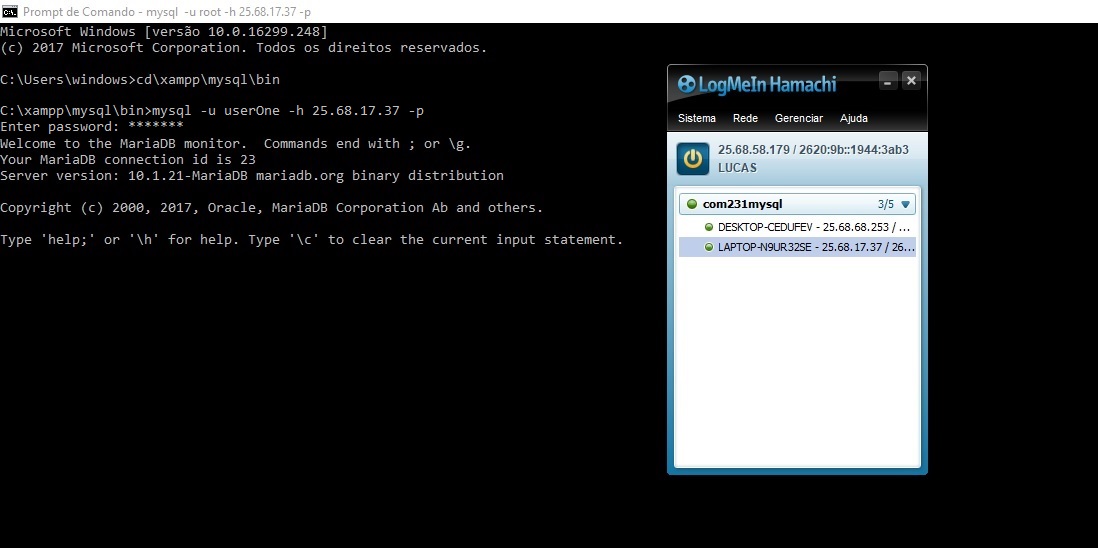


**Cliente 2 (Lucas):**

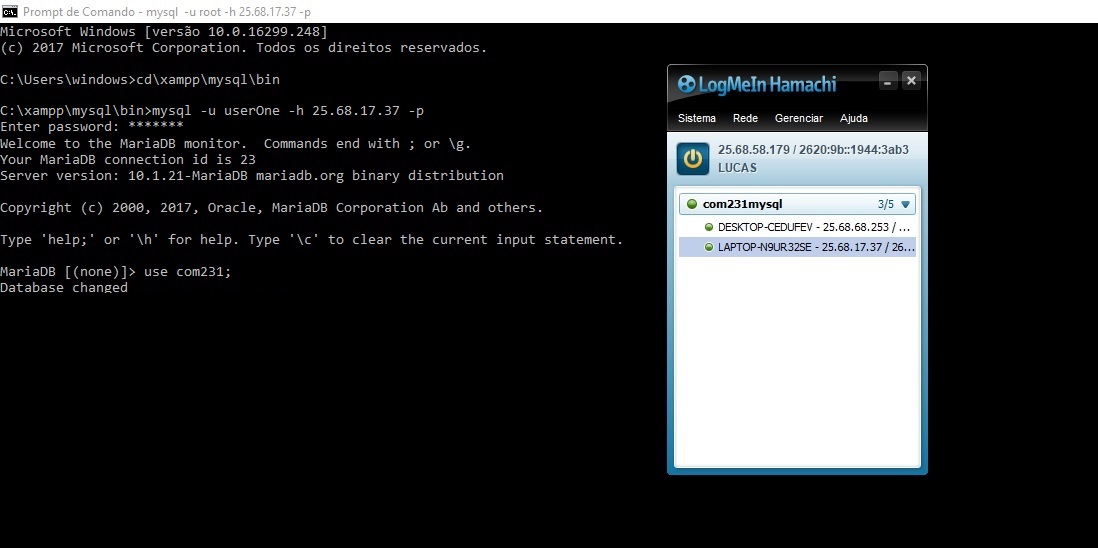
1. Primeiro é necessário abrir o terminal de comando do Xampp, então utiliza-se o comando **cd** mais o caminho necessário para chegar na pasta bin contida em MySql, que se encontra no Xampp, sendo que este caminho pode variar de computador para computador. No caso deste computador o comando é **cd C:\xampp\mysql\bin**, após aplicar este comando será possível continuar o procedimento para ser configurado como cliente. A janela do programa Hamachi encontra-se ao lado para demonstrar que este computador está conectado com o computador que está funcionando como servidor;



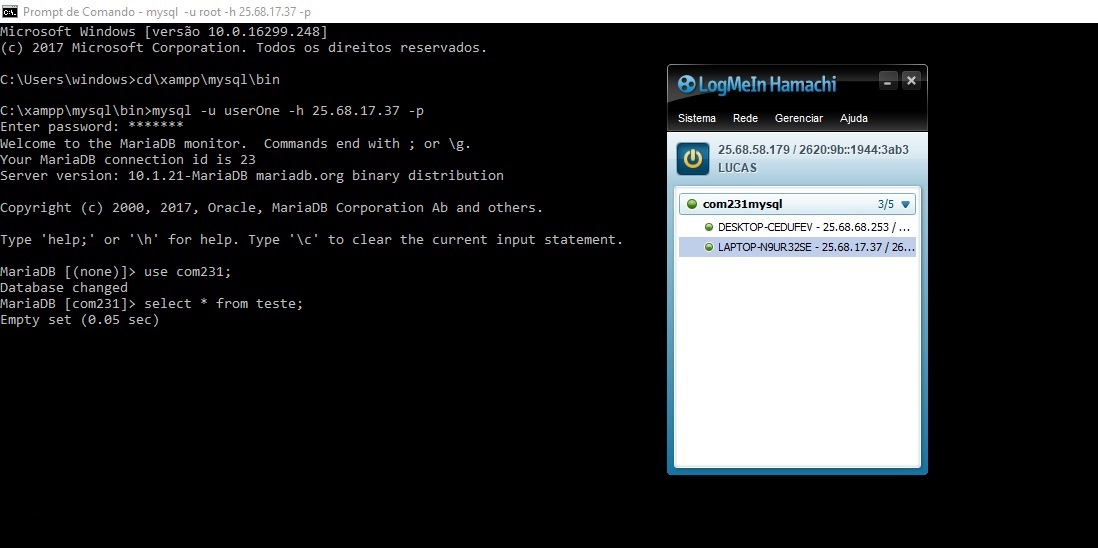
1. Então utiliza-se o comando **mysql -u (nome do usuário) -h (o ip do servidor) -p** para conectar com o computador que está funcionando como servidor e digite a senha predefinida;



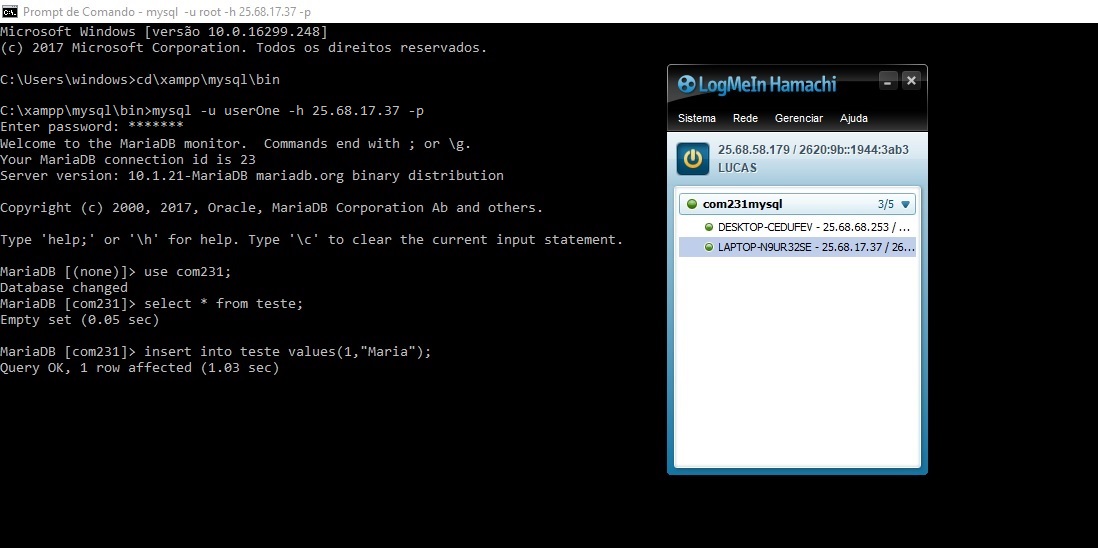
1. Então para utilizar um banco de dados desse servidor é necessário fornecer o nome da base de dados, no caso desta demonstração chama-se com231;



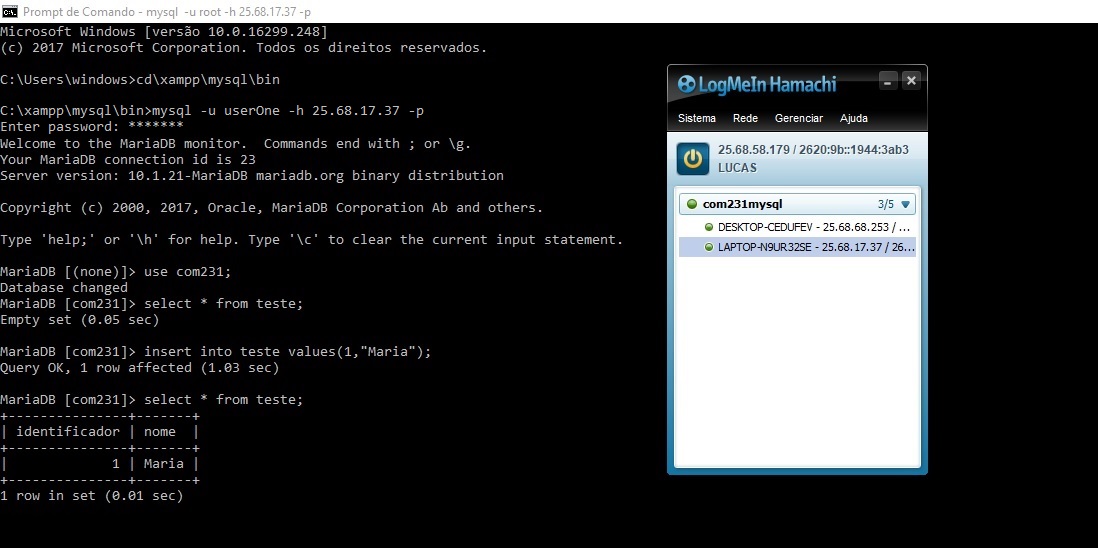
1. Após selecionado o banco de dados pode-se utilizar todos os comandos que o MySql possibilita, no caso foi utilizado o comando **select** para mostrar que a tabela teste foi criada anteriormente, como resultado do **select** foi mostrado que não possui nenhum registro na tabela teste;



1. Usando agora o comando **insert** para saber se o cliente está realmente inserindo valores na tabela.

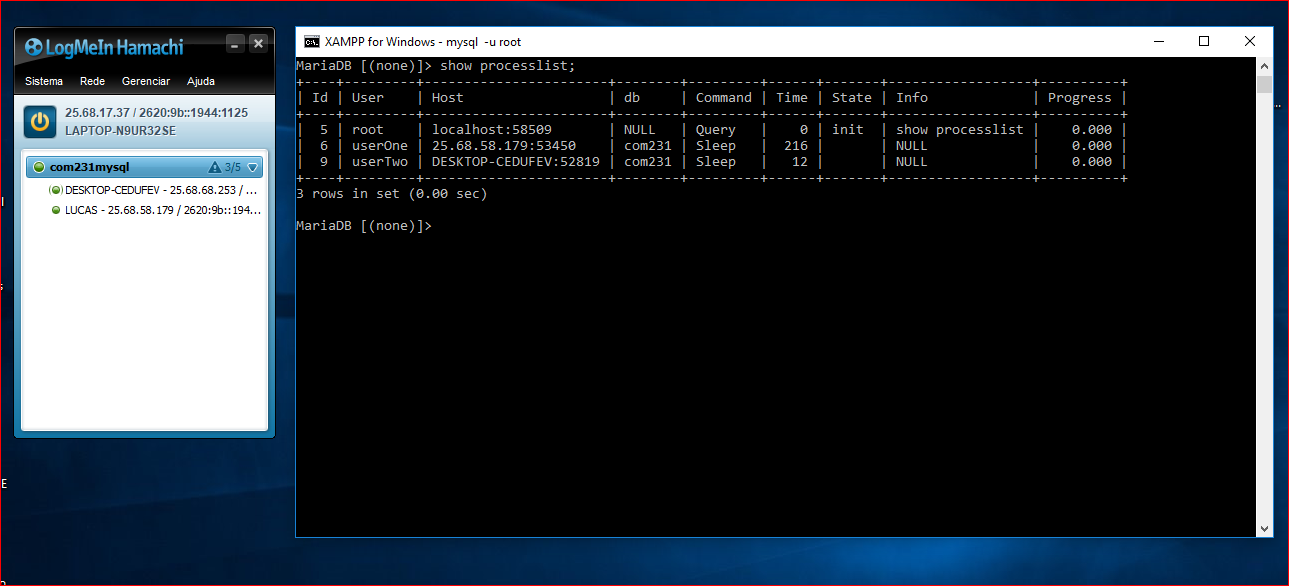


1. E agora repetindo o comando **select** para analisar se a linha foi inserida com sucesso no banco.



**Status do Servidor**

Para as demonstrações acima foi retirado do servidor um screenshot que exibi os dois clientes conectados a base de dados.



**Questão 2:** Estudar os tipos de dados e seus domínios no SGBD. Existe algum tipo específico, que os integrantes do grupo desconheciam? Para quê ele serve?

A seguir encontra-se em forma de tabela os tipos de dados que o MySQL dá suporte, separados em categorias tipo String, Numérico, Data/Tempo, LOB (Large Object) e JSOL (JavaScript Object Notation).

|  |  |  |
| --- | --- | --- |
| TIPO STRING | | |
| SINTAXE DO TIPO DE DADO: | TAMANHO MÁXIMO: | ARMAZENAMENTO(BYTES): |
| \*CHAR (Tamanho) | 255 caracteres | Tamanho em bytes |
| VARCHAR (Tamanho) | 255 caracteres | Tamanho da string + 1 |
| TINYTEXT (Tamanho) | 255 caracteres | Tamanho da string + 1 |
| TEXT (Tamanho) | 65.535 caracteres | Tamanho da string + 2 |
| MEDIUMTEXT (Tamanho) | 16.777.215 caracteres | Tamanho da string + 3 |
| LONGTEXT (Tamanho) | 4.294.967.295 caracteres | Tamanho da string + 4 |
| \*BINARY (Tamanho) | 255 caracteres | Tamanho em bytes |
| VARBINARY (Tamanho) | 255 caracteres | Tamanho da string + 1 |

\* - O tipo não varia conforme o tamanho da string digitada, ou seja, todo o tamanho definido será separado para guardar a string mesmo que não ocupe todo o espaço.

|  |  |  |
| --- | --- | --- |
| TIPO NUMERÍCO | | |
| SINTAXE DO TIPO DE DADO: | TAMANHO MÁXIMO(números): | DESCRIÇÃO |
| BIT(N) | 64 ((N+7)/8Bytes) | Foi realizado um agrupamento por características e tamanho, sendo então os tipos que estão juntos ao  Decimal por tratarem os dados com enfoque na precisão, enquanto no caso do grupo Float utilizam de ponto flutuante, e pôr fim ao Bool e Boolean por tratarem de forma lógica os dados;  N é o número total de dígitos, e D é a quantidade de dígitos decimais; |
| TINYINT | 255 (1 Byte) |
| SMALLINT | 65.535(2Bytes) |
| MEDIUMINT | 16.777.215(3Bytes) |
| INT e INTEGER | 4.294.967.295(4Bytes) |
| BIGINT | 2^64-1(8Bytes) |
| DECIMAL (N, D)  DEC (N, D)  NUMERIC (N, D)  FIXED (N, D) | 4.294.967.295(4Bytes) |
| FLOAT (N, D)  FLOAT (N)  DOUBLE (N, D)  DOUBLE PRECISION (N, D)  REAL (N, D) | 2^64-1(8Bytes) |
| BOOL  BOOLEAN | 0 e 1, sendo 0 igual a Falso e qualquer outro valor é igual a Verdadeiro (1Byte) |

|  |  |  |
| --- | --- | --- |
| TIPO DATA/TEMPO | | |
| SINTAXE DO TIPO DE DADO: | TAMANHO MÁXIMO: | EXPLICAÇÃO: |
| DATE |  |  |
| DATETIME |  |  |
| TIMESTAMP (N) |  |  |
| TIME |  |  |
| YEAR[(2|4)] |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | TIPO BLOB (Large Object) | | | | SINTAXE DO TIPO DE DADO: | TAMANHO MÁXIMO: | DESCRIÇÃO | | TINYBLOB | 255 bytes | O tipo BLOB (Binary Large Object) é utilizado para armazenamento de qualquer tipo de dados em formato binário, como por exemplo uma imagem. No MySql os campos BLOBs são implementados através de campos de texto (TEXT) não case-sensitive e não podem ser usados como chave primária exceto o TINYBLOB. | | BLOB (Size) | 65.535 bytes | | MEDIUMBLOB | 16.777.215 bytes | | LONGBLOB | 4.294.967.295 bytes (4GB) | |

|  |  |  |
| --- | --- | --- |
| TIPO JSON (JavaScript Object Notation) | | |
| SINTAXE DO TIPO DE DADO: | TAMANHO MÁXIMO: | DESCRIÇÃO: |
| JSON | 4.294.967.295 bytes (4GB) | Tipo JSON (JavaScript Object Notation) permite o armazenamento de forma eficiente de objetos, funcionando praticamente como um array, o que permite uma busca pela chave ou pelo índice, que torna a busca nesse tipo mais eficiente por não ter que ler todos os valores dentro do documento |

\*Tipos desconhecidos pelo grupo: JSOL, Espaciais, e BLOB, eram desconhecidos por todos os membros do grupo, dentro dos tipos numéricos e textuais os tipos precedidos por tiny foram uma novidade para os membros também por estarem acostumados a linguagens como Java que não os havia;

Referências:

<https://dev.mysql.com/doc/refman/5.7/en/data-types.html>