

# Banco de Dados

## Aula 08 - Introdu o   Linguagem SQL e Sistemas Gerenciadores de Banco de Dados

# Apresentação

---

Nas aulas anteriores, estudamos diversos aspectos do projeto e modelagem de bancos de dados. Você aprendeu a construir um modelo entidade-relacionamento (MER) e um modelo relacional (MR) para o seu projeto de banco de dados e aprendeu a organizar melhor suas tabelas usando a normalização.

Agora que você já sabe projetar seu banco de dados, vamos entrar numa segunda parte do curso na qual você aprenderá a criar o banco de dados e a manipular as informações que estão guardadas dentro dele. Para isso, vamos apresentar-lhe a **linguagem SQL**.

Você vai aprender o que é SQL, por que ela foi criada e para que serve. Vai estudar algumas informações sobre seu histórico e suas características. Estudará também sobre Sistemas Gerenciadores de Banco de Dados (SGBD), que serão os softwares que você vai utilizar para criar e manipular seus bancos de dados. E, para finalizar, você vai aprender a instalar e a configurar o MySQL, um SGBD muito utilizado em aplicações para web.



## Vídeo 01 - Apresentação

## Objetivos

- Entender o que é SQL e para que é usado.
- Conhecer os principais Sistemas Gerenciadores de Banco de Dados.
- Identificar as principais diferenças dos Sistemas Gerenciadores de Banco de Dados.
- Instalar e configurar o MySQL.

# Linguagem SQL

---

Todo **Sistema Gerenciador de Banco de Dados (SGBD)** deve oferecer aos seus usuários e administradores meios de criar definições de dados, bem como de manipular esses dados armazenados em suas bases. Desse modo, a linguagem SQL, sigla em inglês para *Structured Query Language*, ou Linguagem de Consulta Estruturada, é uma espécie de padrão adotado por diferentes SGBDs, de modo a facilitar a comunicação e a integração dos sistemas. Essa linguagem tem o objetivo de acessar, de forma estruturada e declarativa, elementos de tabelas de bancos de dados.

Além disso, é uma linguagem de pesquisa para uso em Banco de Dados Relacional. Você deve lembrar que estudou uma de programação estruturada, que permitiu a construção dos seus algoritmos, ou Java, como exemplo de linguagem orientada a objetos. Agora, vamos utilizar SQL como linguagem de construção, modificação e acesso a Banco de Dados. Cabe ressaltar que essa linguagem tem um caráter diferenciado das linguagens de programação, pois é uma linguagem declarativa, ou seja, não é necessário que você programe qualquer algoritmo para acessar o conteúdo das tabelas. Muito bom, não acha?

A linguagem SQL é um grande padrão de banco de dados, resultado da sua simplicidade e facilidade de uso. Ela se diferencia de outras linguagens de consulta a banco de dados, pois uma consulta SQL especifica a forma do resultado e não o caminho para chegar a ele.

Você pode se perguntar: qual a vantagem de existir uma linguagem padrão de acesso a banco de dados? Imagine se você estivesse numa reunião com um italiano, um chinês e um russo e ninguém falasse outra língua a não ser a de origem. Como se daria a comunicação? Provavelmente por mímicas e/ou desenho. Os sistemas computacionais ainda não conseguem utilizar esse tipo de estratégia, desse modo foi instituído um **padrão de comunicação** para Banco de Dados Relacional.

A linguagem SQL foi desenvolvida originalmente no início dos anos 1970 nos laboratórios da IBM em *San Jose*, dentro do projeto *System R*. O nome original da linguagem era **SEQUEL**, acrônimo para *Structured English Query Language* (Linguagem de Consulta Estruturada em Inglês). Embora a linguagem SQL tenha sido

originalmente criada pela IBM, rapidamente surgiram vários "dialetos" produzidos por outros desenvolvedores. Essa expansão levou à necessidade de criar e de adaptar um padrão para a linguagem. Essa tarefa foi realizada pela *American National Standards Institute* (ANSI), em 1986, e pela *International Organization for Standardization* (ISO), em 1987. O SQL foi revisto três vezes, gerando a versão SQL:2003.

Embora SQL seja a linguagem de banco de dados mais influente do mercado, ela não é a única. A seguir, podemos conferir alguns exemplos de outras linguagens.

- **QBE** (*Query-by-Example*) – Integra o sistema de banco de dados QBE, foi desenvolvida pela IBM no início de 1970.
- **Quel** – Linguagem de consulta lançada para o banco de dados Ingres, desenvolvida na Universidade da Califórnia em *Berkeley*.
- **Datalog** – Linguagem de consulta baseada na lógica de programação Prolog.

SQL é caracterizada pela utilização de palavras-chaves, que podem ser classificadas, de acordo com sua função, nos seguintes tipos.

- **DML** – Linguagem de Manipulação de Dados, subconjunto da linguagem SQL, usado para inserir, atualizar e apagar dados. Exemplos: INSERT, UPDATE, DELETE.
- **DDL** – Linguagem de Definição de Dados, permite ao utilizador definir tabelas novas e elementos associados. Exemplos: CREATE, DROP.
- **DCL** – Linguagem de Controle de Dados, controla quem tem acesso para ver ou manipular dados dentro do banco de dados. Exemplos: GRANT, REVOKE.
- **DTL** – Linguagem de Transação de Dados, usada para o controle de transações no banco de dados. Exemplo: START TRANSACTION.

- **DQL** – Linguagem de Consulta de Dados, permite ao usuário especificar uma consulta (query) como uma descrição do resultado desejado. Exemplo: SELECT.

Todos esses termos estranhos, escritos em letra maiúscula, citados como exemplo, fazem parte da linguagem SQL. Nas próximas aulas, você descobrirá para que eles servem e como utilizá-los.

Já utilizamos na aula de hoje, algumas vezes, o termo **consulta**, mas o que seria isso? Quando precisamos de uma informação que está armazenada em um banco de dados, dizemos que vamos fazer uma consulta ao mesmo. Imagine que você precisa ligar para um amigo, mas não lembra o número do telefone dele. O que você faz? Digita o nome dele na agenda do seu telefone, fazendo uma consulta.



**Vídeo 02** - Vantagens da Linguagem SQL

## Atividade 01

---

1. Descreva alguma situação do seu dia a dia em que você precise utilizar um banco de dados. Especifique um exemplo de consulta que você pode fazer por meio dele.

# Sistemas Gerenciadores de Banco de Dados

---

Os Sistemas Gerenciadores de Banco de Dados (SGBDs) são conjuntos de programas de computador (softwares) responsáveis pelo gerenciamento de um banco de dados. O principal objetivo é retirar da aplicação em desenvolvimento a responsabilidade de gerenciar o acesso, manipulação e organização dos dados. O SGBD disponibiliza uma interface para que os seus clientes possam incluir, alterar ou consultar dados.

Todas as organizações possuem grandes quantidades de dados e informações que precisam armazenar. Contudo, nem sempre o armazenamento de dados em papel é confiável e de fácil acesso. Nesse sentido, torna-se mais fácil encontrar a informação num banco de dados que se utilize das tecnologias de informação atuais para seu gerenciamento. Ou seja, os bancos de dados estendem a função do papel ao guardar a informação em computadores.

Qualquer empresa que pretenda garantir um controle efetivo sobre o seu negócio tem obrigatoriamente que recorrer a SGBDs. A planilha eletrônica é uma ferramenta de controle usada para operacionalizar os dados e assim criar informação útil ao planejamento diário das empresas. Contudo, existem outros tipos de ferramentas, mais completas e com funcionalidades extras que elevam a capacidade operacional de gerenciar informação de valor para a organização.

Um SGBD não é nada mais do que um conjunto de programas que permitem armazenar, modificar e extrair informação de um banco de dados. Há diferentes tipos de SGBD, desde pequenos sistemas que funcionam em computadores pessoais, até sistemas complexos associados a servidores de grandes empresas.



**Vídeo 03** - Características de um SGBD

Os SGBDs mais populares que utilizam linguagem SQL estão listados a seguir.

- **Oracle** – surgiu no fim dos anos 70, desenvolvido pela empresa de mesmo nome, que é líder de mercado no segmento de banco de dados, sendo pioneira no lançamento de novas soluções de banco de dados para diversos segmentos, sobretudo, no campo das aplicações corporativas. Atualmente, encontra-se na versão *Oracle 12c*.
- **SQL Server** – criado pela *Microsoft* em 1989, surgiu de uma parceria entre a *Sybase*, *Ashton-Tate* e *Microsoft*. Atualmente, encontra-se na versão *Microsoft SQL Server 2016*, sendo considerada uma plataforma de dados confiável, produtiva e inteligente.
- **PostgreSQL** – SGBD de código aberto do tipo objeto relacional lançado em 1995, resultante de uma evolução do projeto Ingres, desenvolvido na Universidade da Califórnia em *Berkeley*. Encontra-se atualmente na versão 9.4.
- **Firebird** – SGBD de código aberto, criado a partir do *InterBase* da *Borland*, quando da abertura de seu código em julho de 2000. Encontra-se atualmente na versão 3.0.1.
- **MySQL** – SGBD de código aberto, criado na década de 1980 na Suécia, considerado o SGBD *open source* mais popular no mundo atualmente, sendo utilizado por grandes corporações, como NASA, Banco Bradesco, Dataprev, HP, *Nokia*, *Sony*, *Cisco Systems*, *Google* e outros. O MySQL foi comprado em 2008 pela *SUN Microsystems*, que posteriormente foi comprada pela *Oracle*. Ou seja, o MySQL atualmente é um produto da *Oracle*. O seu sucesso é atribuído à sua fácil integração com PHP presente na maioria das ferramentas de desenvolvimento para Internet. Encontra-se atualmente na versão 5.7.



**Vídeo 04** - Arquitetura ANSI/SPARC

## Atividade 02

---

1. Descubra outras características de um dos SGBDs listados anteriormente, consultando sites de busca. Se a página acessada estiver em inglês, será uma ótima oportunidade de você treinar o que aprendeu na disciplina de inglês técnico.

## Instalando o *MySQL*

---

Considerando o resumo apresentado na lista anterior, escolhemos o MySQL como exemplo de SGBD a ser utilizado em nossa disciplina. A partir de agora, vamos ensinar você a instalar e configurar o ambiente. Após isso, você já estará apto a entrar no mundo maravilhoso da linguagem SQL.

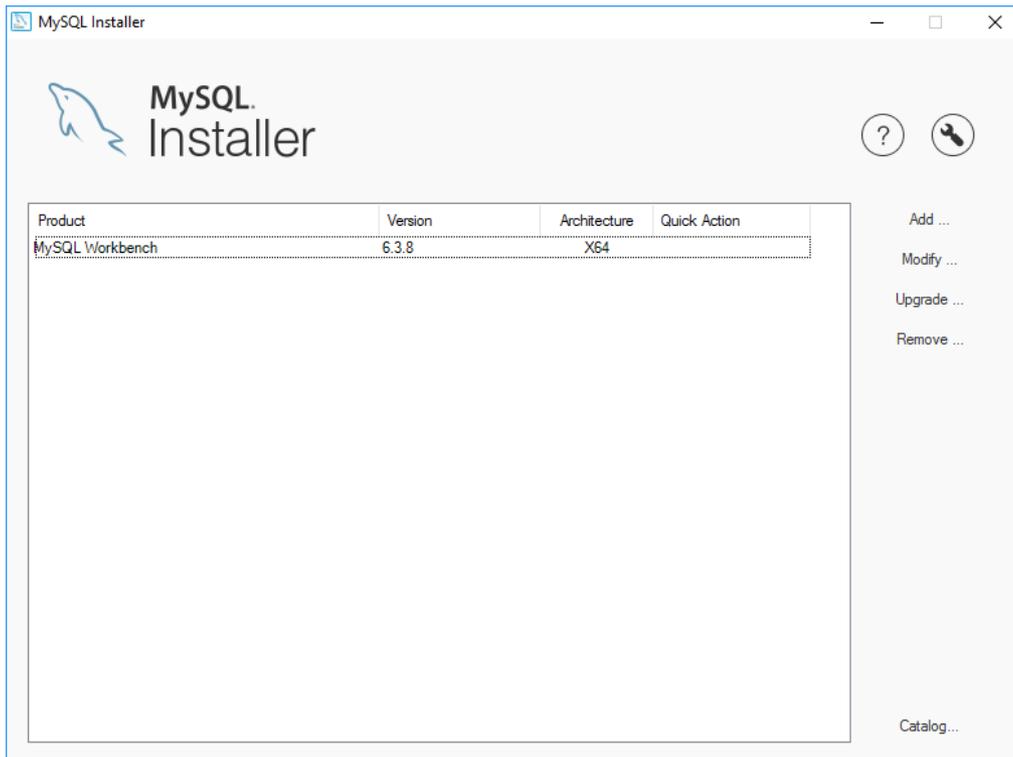
Como vimos anteriormente, o MySQL é o SGBD de código aberto mais popular no mundo atualmente. Por isso, nós o utilizaremos nas nossas aulas. A versão que utilizaremos será a versão MySQL Community Server 5.7.17. Versões mais recentes deverão ser lançadas e possivelmente poderão ser compatíveis com nosso material.

Para instalar o MySQL Server Community será necessário realizarmos o download do MySQL Installer uma ferramenta que permitirá a instalação do MySQL Server. Para obter o MySQL Installer 5.7 for Windows você poderá acessar a página do link <<https://dev.mysql.com/downloads/mysql/>>. Localize o botão Download correspondente e então clique. Você será levado para uma nova página onde são apresentadas duas opções do MySQL Installer: a primeira opção, com menos de 2MB, somente realizará o download dos arquivos do MySQL Server no momento da instalação.

Já a segunda opção do MySQL Installer, com mais de 380MB, os arquivos necessários para a instalação do MySQL Server já estarão disponíveis no momento da instalação. Recomendamos o download dessa segunda opção.

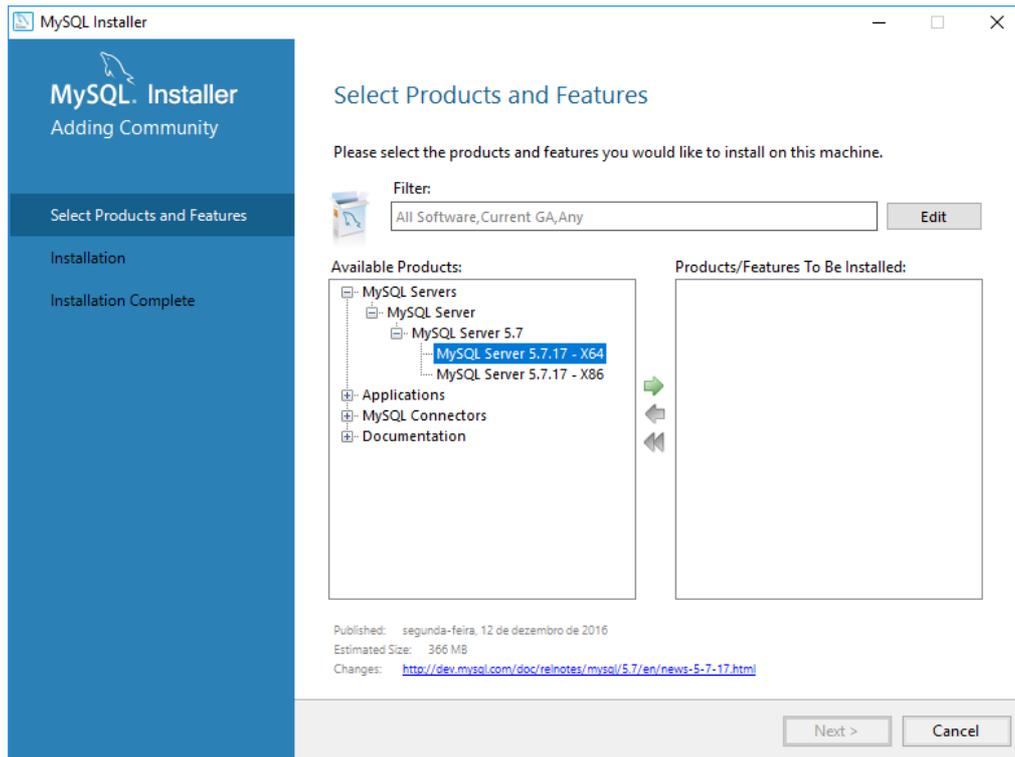
Após instalar e executar o MySQL Installer, você verá a tela apresentada na **Figura 1**. Clique em **"Add ..."** para iniciarmos o processo de instalação do MySQL Server.

**Figura 01** - Tela Inicial do MySQL Installer.



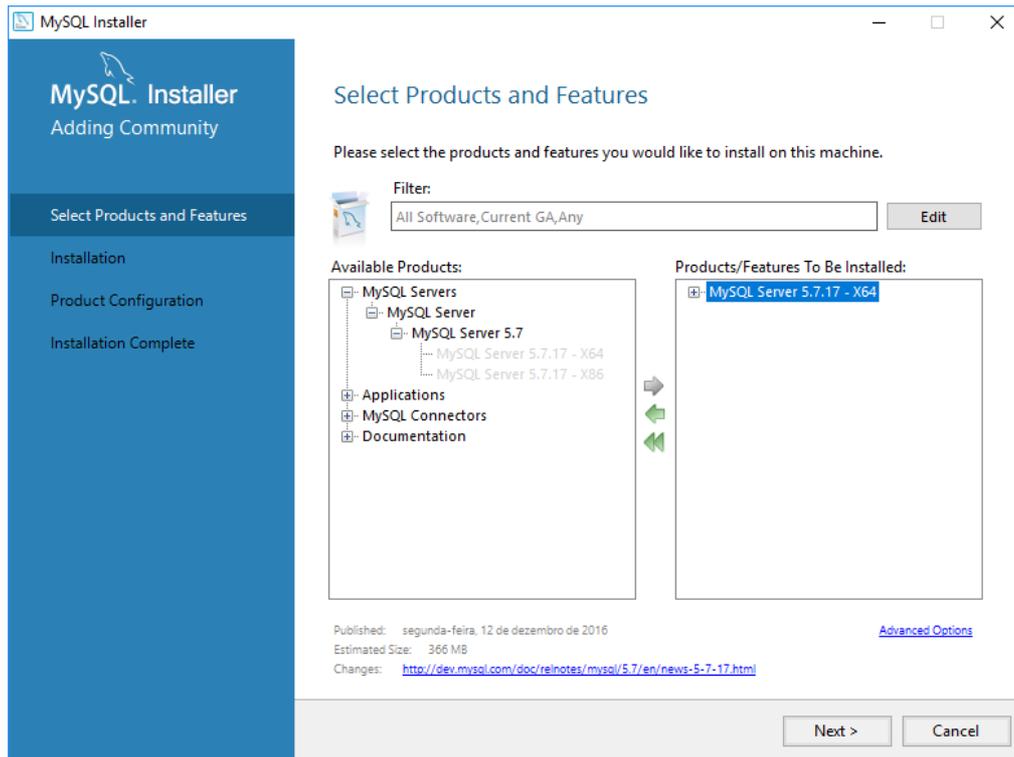
Na tela da **Figura 2** você deverá expandir a as opções referentes ao MySQL Server. Note que há duas verões disponíveis para a instalação: a versão para sistemas operacionais de 64bits e a versão de 32bits. Você deverá selecionar a versão que for adequada ao seu sistema operacional. Em nosso exemplo estamos realizando a instalação no Windows de 64bits.

**Figura 02** - Instalação do MySQL Server.



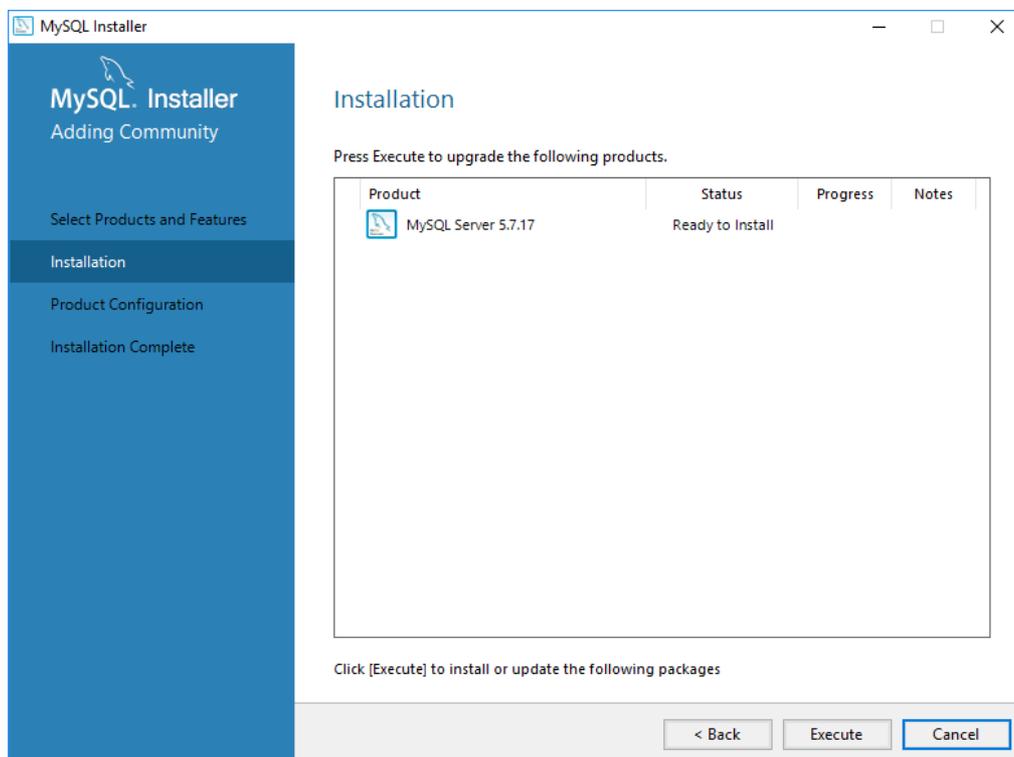
Para selecionar a versão que será instalada, clique sobre o "MySQL Server 5.7.17 - X64" e então clique na seta localizada ao centro da janela e que está apontando para a caixa a direita. Após clicar na seta a tela deverá ser semelhante ao apresentado na **Figura 3**. Clique em *Next >*.

**Figura 03** - Instalação do MySQL Server.



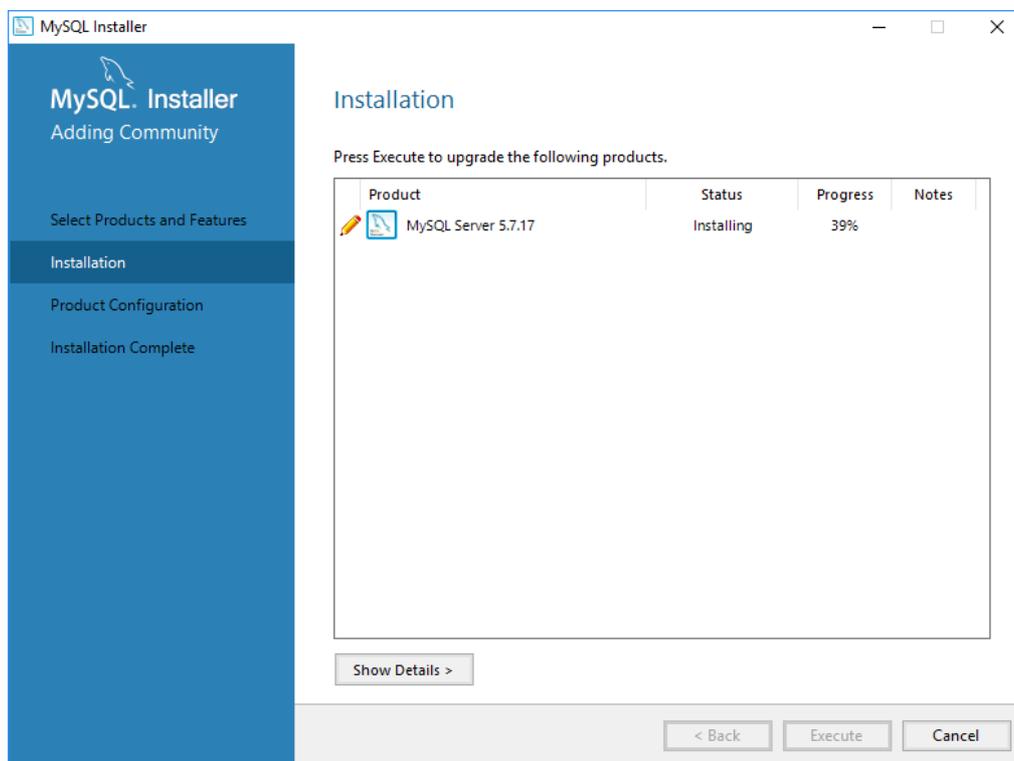
A tela seguinte, **Figura 4**, é apresentada um resumo do que será instalado. Clique então em *Execute* para continuar o processo de instalação.

**Figura 04** - Instalação do MySQL Server.

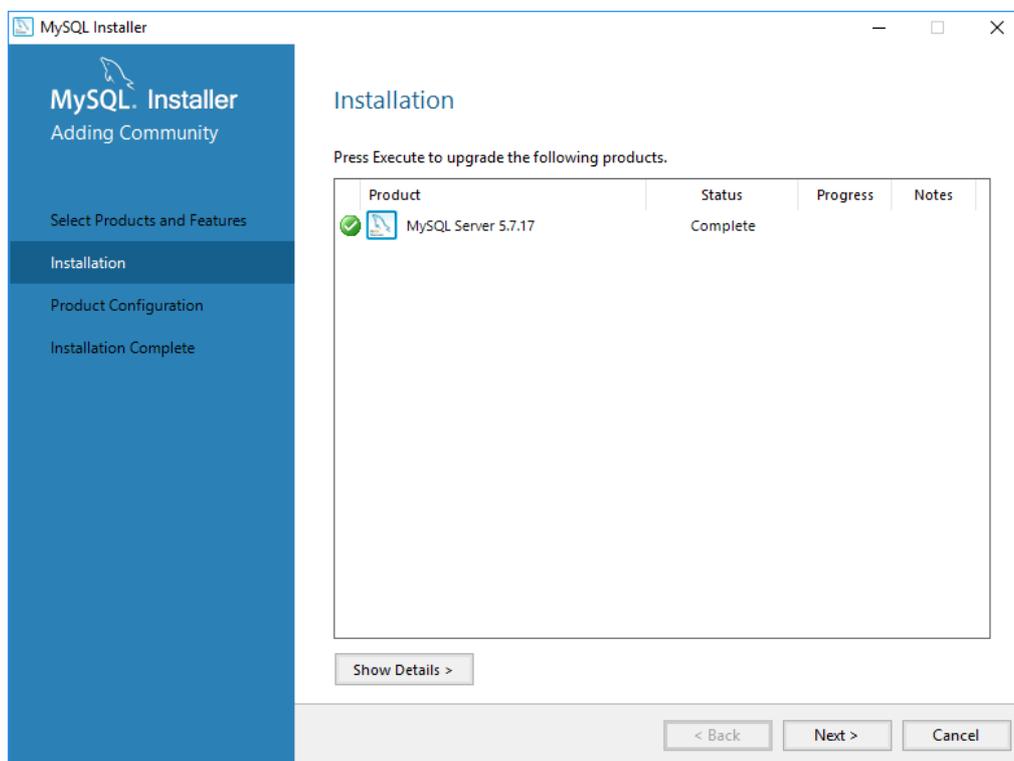


A **Figura 5** mostra o progresso da instalação do MySQL Server, já a **Figura 6** indica que o processo de instalação foi concluído. Clique no botão *Next >*.

**Figura 05** - Instalação do MySQL Server.

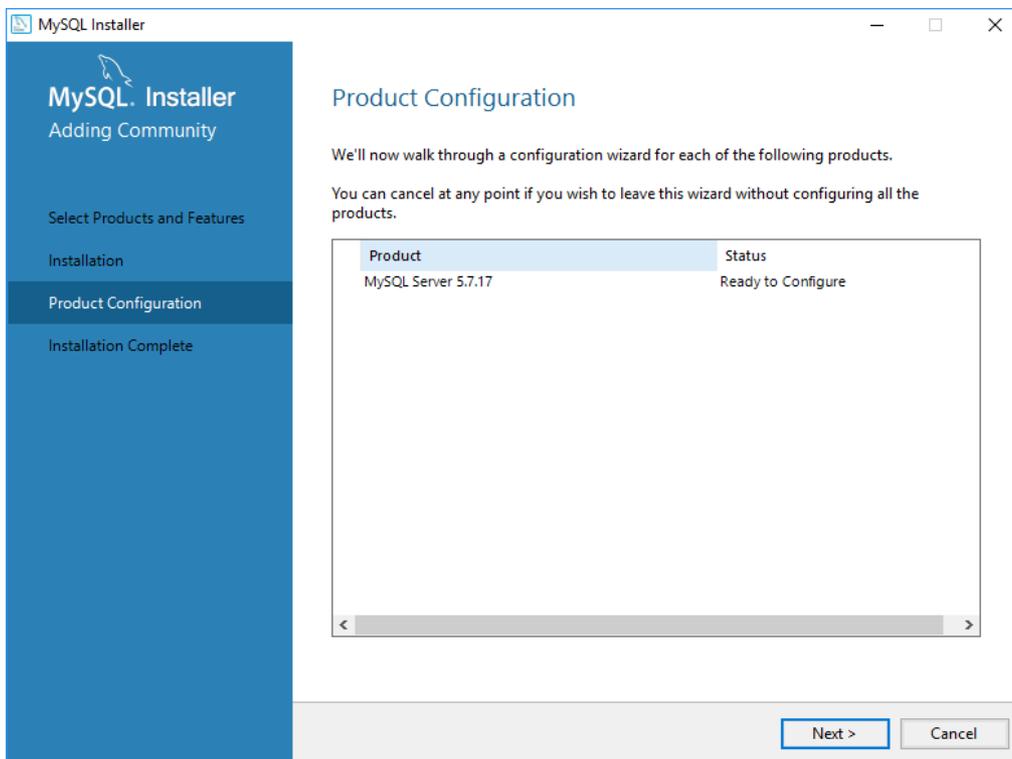


**Figura 06** - Instalação do MySQL Server.



Após a instalação do MySQL Server será apresentada a tela da **Figura 7**. A partir desse momento iremos realizar a configuração inicial do serviço do banco de dados.

**Figura 07** - Configuração do MySQL Server após a instalação.

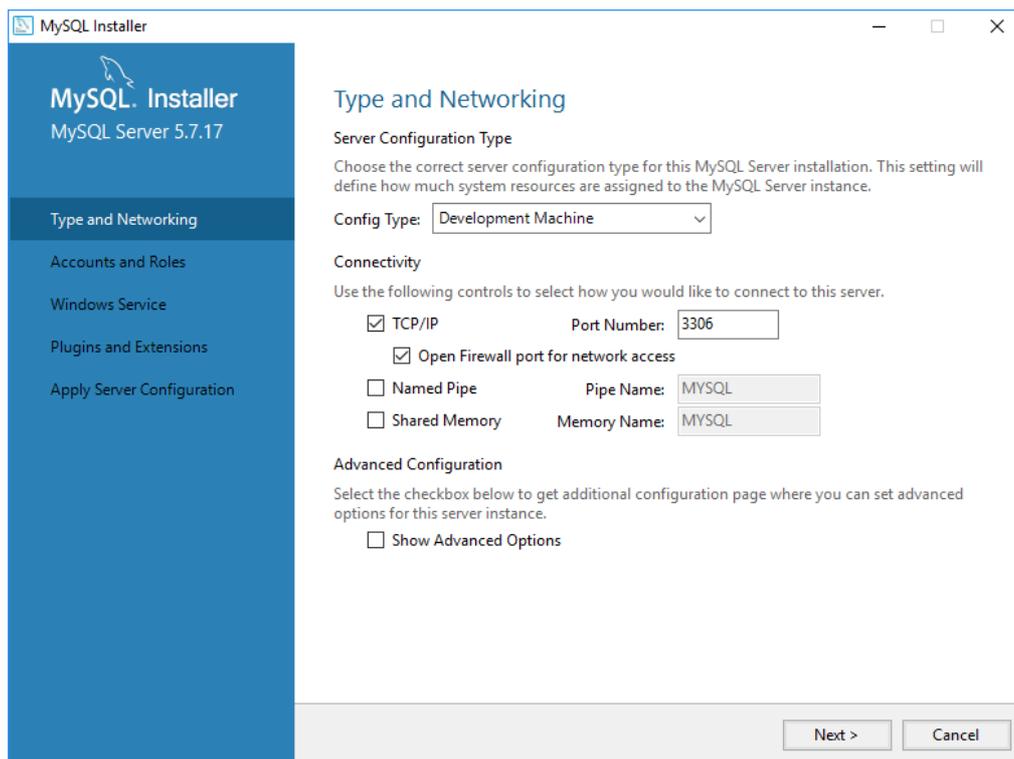


A **Figura 8** apresenta o tipo de configuração que desejamos em *Config Type*. No nosso caso vamos utilizar a opção *“Development Machine”*, já que a nossa instalação será utilizada para o desenvolvimento de nossas atividades e não necessitará do desempenho máximo do MySQL Server.

As outras opções disponíveis são utilizadas quando vamos montar um *“Ambiente de Produção”* para a execução de Softwares e Aplicações que necessitam de todo o desempenho disponível da máquina em que o banco de dados está instalado.

Ainda na tela apresentada na **Figura 8** é possível definir o número da porta que o MySQL Server deverá aceitar as conexões e a porta padrão é a 3306. Defina suas opções conforme o que está apresentado e então clique em *Next >*.

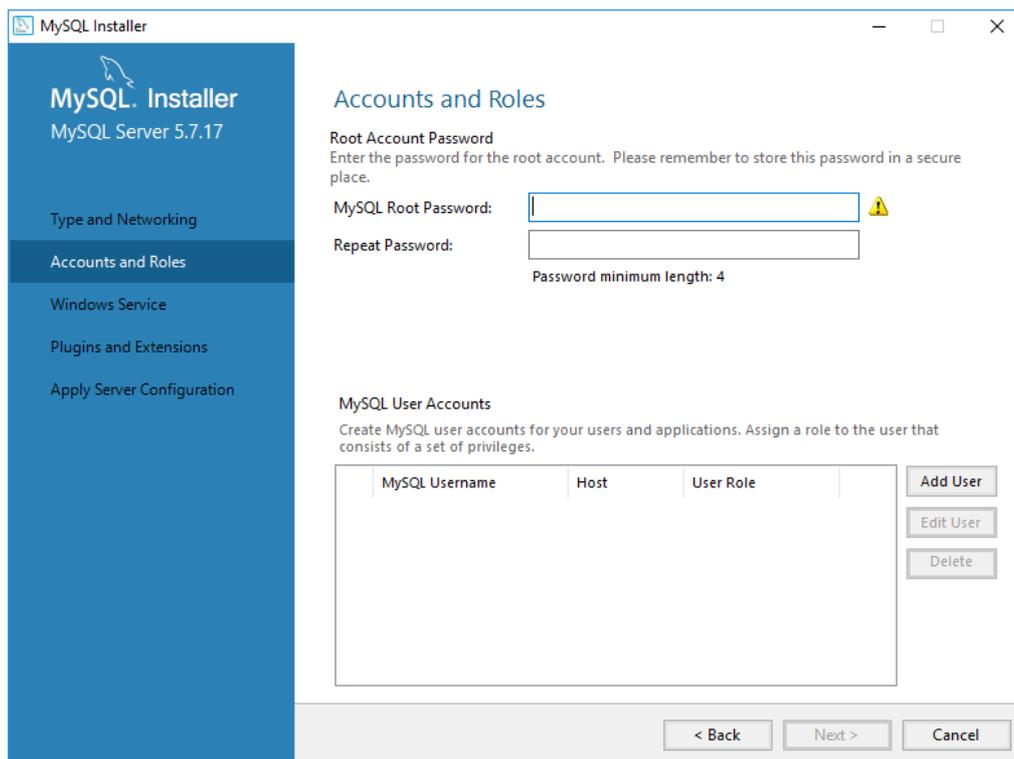
**Figura 08** - Configuração do MySQL Server após a instalação.



Na tela seguinte, **Figura 9**, devemos definir a senha do usuário “root”. Esse usuário possui a permissão para criar, apagar ou modificar relações/tabelas, registros e os usuários cadastrados no banco de dados.

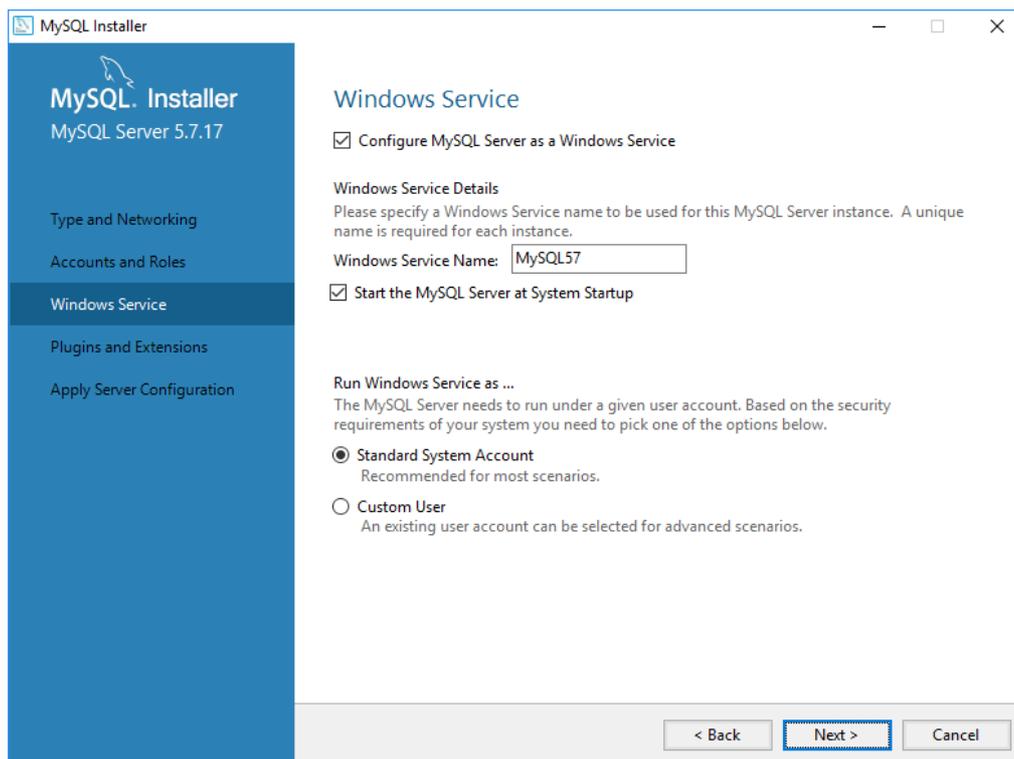
A senha para esse usuário não poderá possuir menos de 4 caracteres. Como estamos instalando um ambiente de desenvolvimento devemos definir uma senha fácil de ser memorizada. Podemos utilizar a palavra “senha” (sem aspas e toda e minúsculas) nessa instalação. Após inserir a senha nas duas caixas de texto, clique em *Next >*.

**Figura 09** - Configuração do MySQL Server após a instalação.



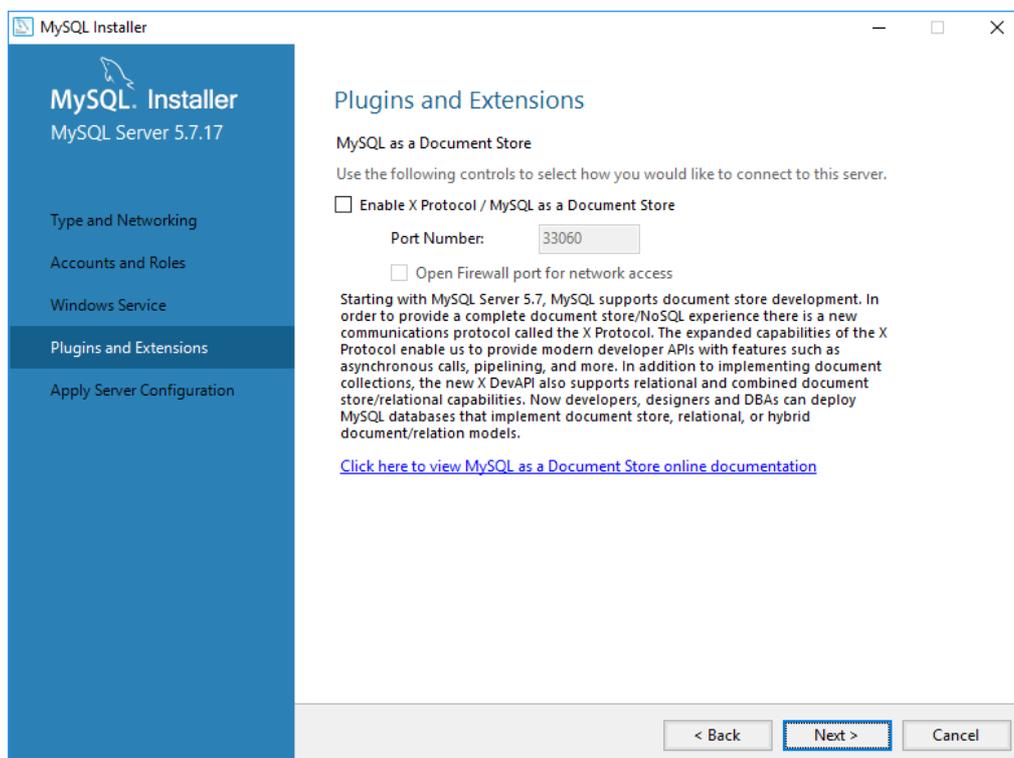
Na tela da **Figura 10** podemos definir se o MySQL Server será executado como um serviço, o nome do serviço (MySQL57) e se ele deverá ser inicializado junto com o sistema operacional. Não iremos realizar modificações nessa etapa, clique em *Next >*.

**Figura 10** - Configuração do MySQL Server após a instalação.



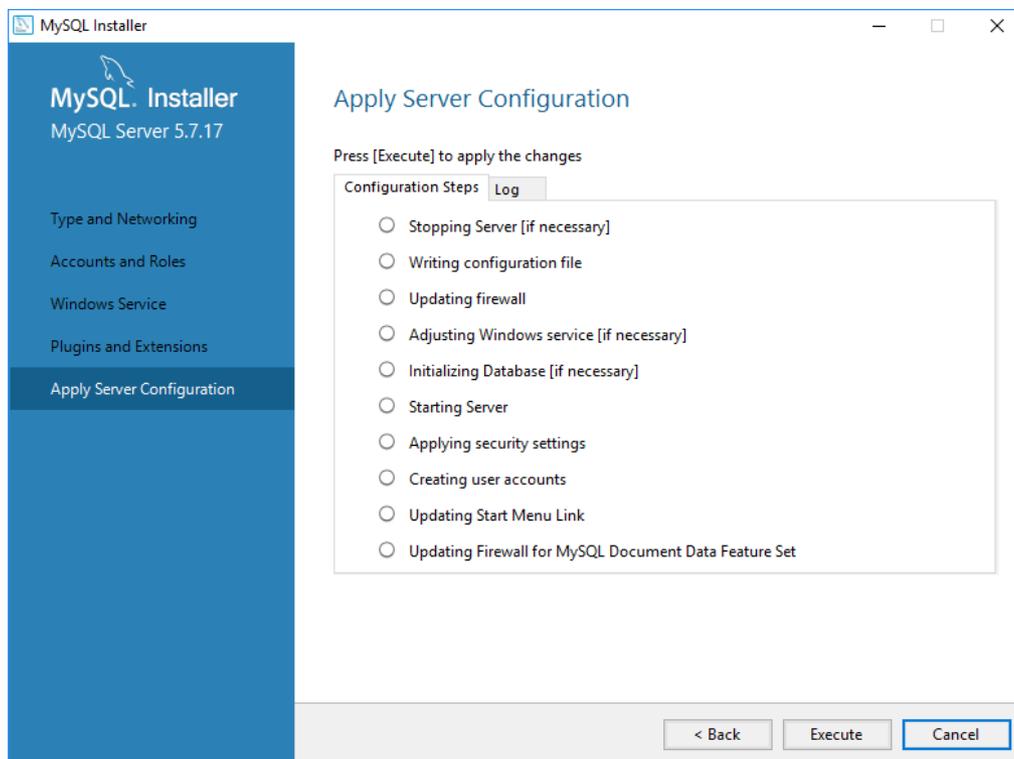
Na tela da **Figura 11** podemos ativar o protocolo adicional. Não vamos necessitar realizar nenhuma configuração nessa tela, devemos então clicar em *Next >*.

**Figura 11** - Configuração do MySQL Server após a instalação.



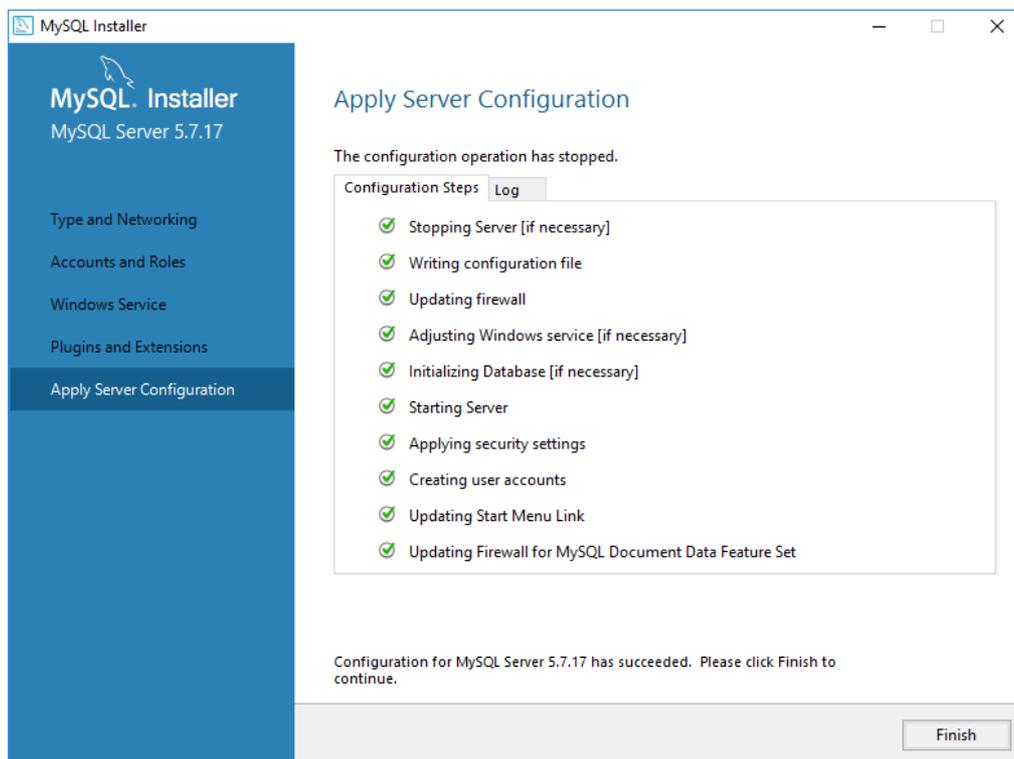
Na próxima tela, **Figura 12**, são apresentados os passos da configuração que realizamos. Para que possamos aplicar as configurações definidas até o momento devemos clicar no botão *Execute*.

**Figura 12** - Configuração do MySQL Server após a instalação.



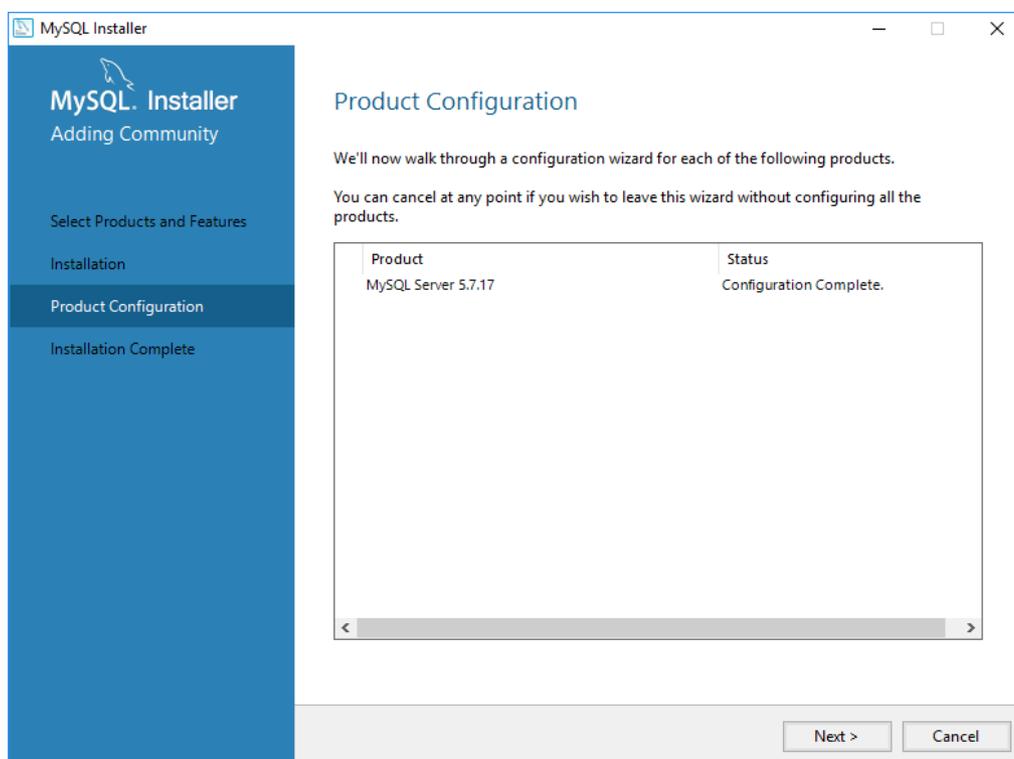
A tela da **Figura 13** apresenta que todos os passos foram concluídos com sucesso. Clique em *Finish* para finalizar o processo de instalação e configuração do MySQL Server.

**Figura 13** - Configuração do MySQL Server após a instalação.

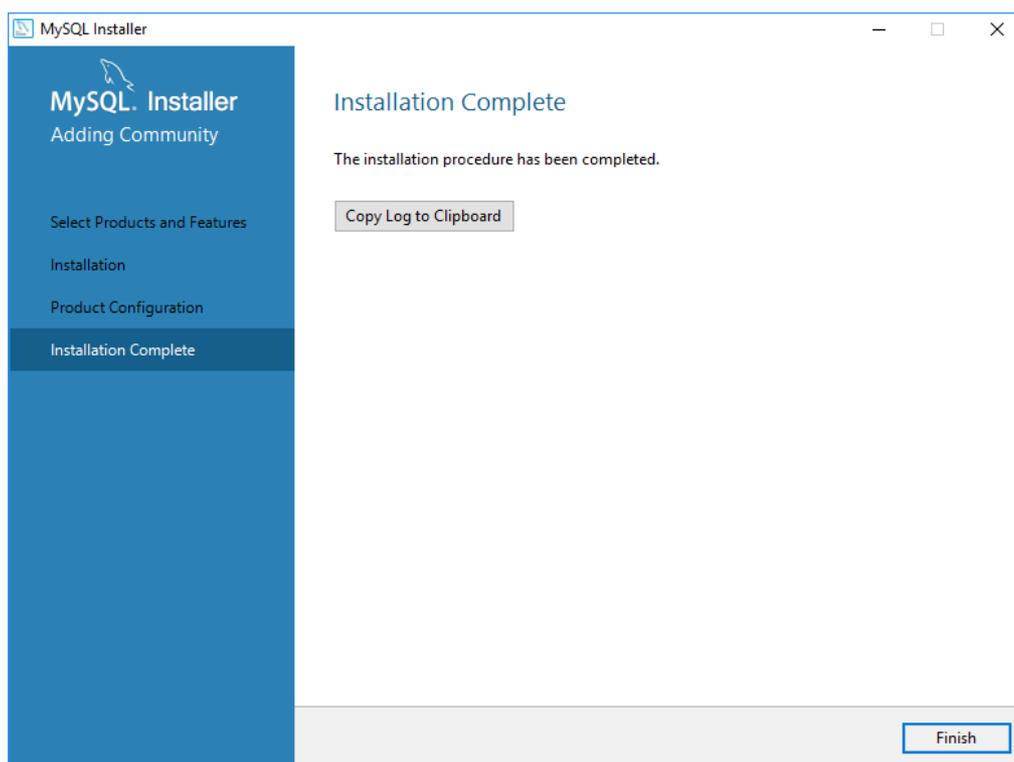


Na tela da **Figura 14** e na da **Figura 15** clique em *Next >* e em *Finish* para concluir todo o processo.

**Figura 14** - Configuração do MySQL Server após a instalação.



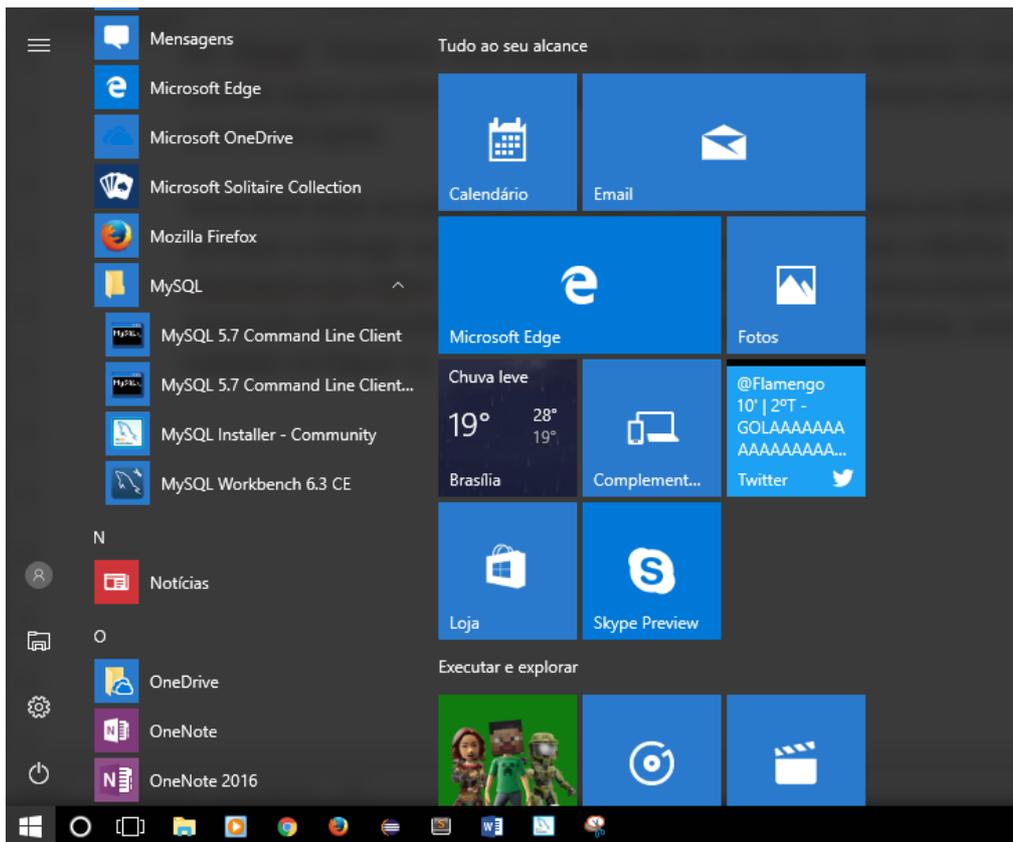
**Figura 15** - Configuração do MySQL Server após a instalação.



Concluída a configuração, você verá a última tela (**Figura 15**), que apresenta uma mensagem de informação sobre a conclusão da instalação e configuração. Clique em *“Finish”*. Parabéns, você acaba de instalar e configurar o MySQL! Caso tenha ocorrido algum problema na instalação e/ou configuração, procure seu tutor para que ele lhe ajude.

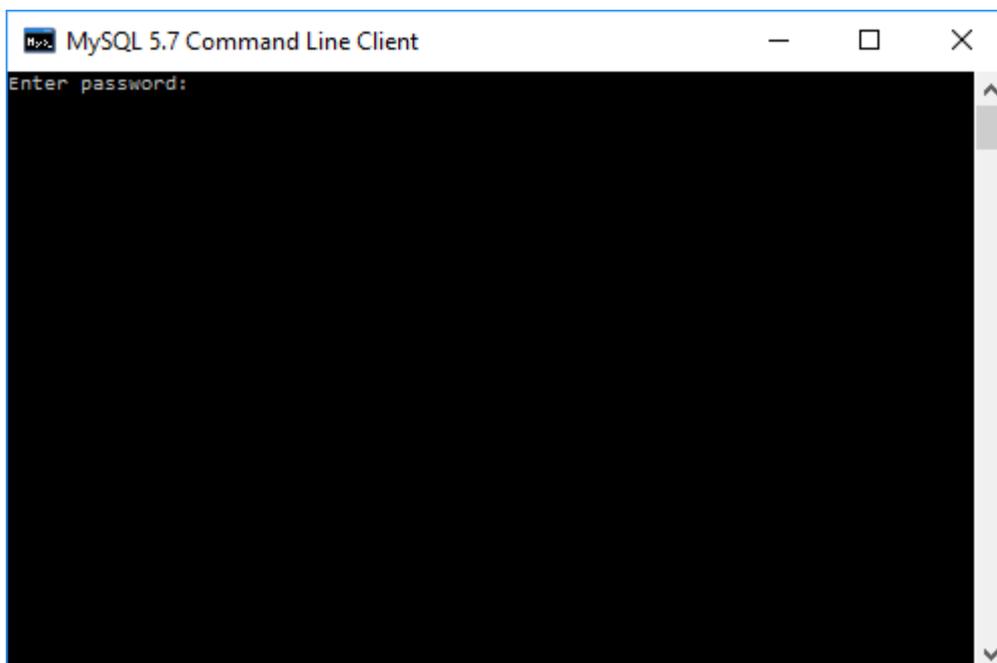
Você deve estar se perguntando: e agora? Bom, para ter acesso ao MySQL e começar a interagir com seu banco de dados, você irá procurar o *MySQL Command Line Client* no menu MySQL dentro dos seus programas, acessíveis diretamente pelo menu Iniciar do seu ambiente Windows, como ilustrado na **Figura 16**.

**Figura 16** - Acesso ao MySQL Command Line Client pelo ambiente Windows.



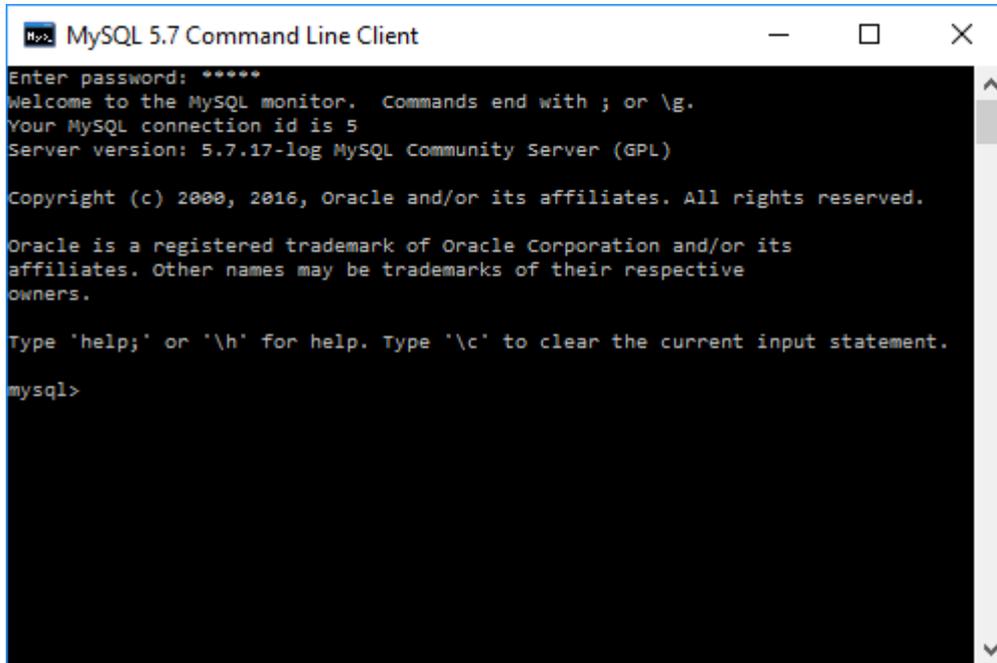
Ao clicar no ícone, aparecerá a tela de entrada do MySQL (**Figura 17**).

**Figura 17**



Na tela inicial do MySQL, é solicitado a você que digite a senha de *root* que foi configurada durante a configuração do ambiente. Digite a senha e tecele *enter*. Uma nova tela aparecerá dando-lhe as boas-vindas ao ambiente MySQL (**Figura 18**).

**Figura 18**



Agora, já podemos começar a interagir com a linguagem SQL. Que tal criar seu primeiro banco de dados? Digite a sequência de comandos apresentada no quadro a seguir. Não esqueça de teclar *enter* após o sinal de ponto e vírgula.

```
1 mysql> CREATE DATABASE metropole;  
2 mysql> SHOW DATABASES;
```

O primeiro comando (CREATE DATABASE) é usado para criar bancos de dados. No exemplo, criamos um banco de dados intitulado de *metropole*. O segundo comando (SHOW DATABASES) é usado para mostrar os bancos de dados existentes no servidor. Você viu que, além do banco de dados que você criou, existem outros dois denominados de *information\_schema* e *mysql*. Eles foram criados automaticamente durante a instalação do programa.

Tente executar os comandos acima, escrevendo os comandos SQL de forma errada, por exemplo, SHOW DATABASE, ou sem digitar o ponto e vírgula no final. Veja o que acontece.

Se você tiver alguma dúvida sobre a escrita ou a função de algum comando, pode digitar "*help* nome\_do\_comando" e teclar *enter* que você terá uma breve descrição do comando. Tudo certo? Pegue sua agenda e anote ao lado da data de hoje: "criei meu primeiro banco de dados em SQL!".

## Atividade 03

---

1. Você deve ter percebido que o acesso ao MySQL é feito diretamente pela linha de comando. Será que existe alguma interface gráfica que facilite essa interação? Faça uma pesquisa na Internet e verifique se existe alguma ferramenta visual para acesso ao MySQL.

# Resumo

---

Nesta aula, você verificou que existe uma linguagem padrão de acesso aos bancos de dados, denominada de SQL, e conheceu um pouco de sua história. Em seguida, aprendeu que, para usar um banco de dados, é necessário instalar um Sistema Gerenciador de Banco de Dados (SGBD) e que existem diversas opções com características variadas. Aprendeu, também, a instalar e configurar a ferramenta MySQL. Agora que você já tem o SGBD instalado, você está pronto para explorar a linguagem SQL em toda a sua profundidade. Será isso que faremos a partir da próxima aula.

## Autoavaliação

---

1. O que é e para que serve a SQL?
2. SQL é a única linguagem utilizada em banco de dados? Justifique.
3. O que é uma consulta em banco de dados?
4. O que é e para que serve um SGBD?
5. Cite duas vantagens existentes na escolha do MySQL como SGBD para nossas aulas.

## Referências

---

BEIGHLEY, L. **Use a cabeça SQL**. Rio de Janeiro: Editora AltaBooks, 2008.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de banco de dados** 8. ed. Rio de Janeiro: Editora Campus/Elsevier, 2006.