Início Rápido: Instalar o SQL Server e criar um banco de dados no Ubuntu

Artigo • 15/01/2022 •

9 minutos para o fim da leitura •









Neste artigo

Pré-requisitos

Instalar o SQL Server

Instalar as ferramentas de linha de comando do SOL Server

Conectar-se localmente

Criar e consultar dados

Práticas recomendadas de desempenho

Ferramentas de dados multiplataforma

Conectando do Windows

Outros cenários de implantação

Próximas etapas

Aplica-se a: SQL Server (todas as versões compatíveis) – Linux

Neste início rápido, você instalará o SQL Server 2019 no Ubuntu 16.04, 18.04 ou 20.04. Em seguida, você se conecta ao **sqlcmd** parar criar seu primeiro banco de dados e executar consultas.

Observação

Há suporte para o Ubuntu 20.04 do SQL Server 2019 CU10 em diante.

□ Dica

Este tutorial requer a entrada do usuário e uma conexão com a Internet. Se estiver interessado nos procedimentos de instalação autônoma ou offline, confira Diretrizes de instalação para o SQL Server em Linux. Para ver uma lista das plataformas com suporte, confira nossas Notas sobre a versão.

Pré-requisitos

É necessário ter um computador com o Ubuntu 16.04, 18.04 ou 20.04 com **pelo menos 2 GB** de memória.

Para instalar o Ubuntu 20.04 no seu computador, acesse https://releases.ubuntu.com/20.04/ ☑. Também é possível criar máquinas virtuais Ubuntu no Azure. Confira Criar e gerenciar VMs do Linux com a CLI do Azure.

(!) Observação

Neste momento, não há suporte para o **Subsistema do Windows para Linux** para Windows 10 ou Windows 11 como um destino de instalação para cargas de trabalho de produção.

Para obter outros requisitos do sistema, confira Requisitos do sistema do SQL Server em Linux.

Instalar o SQL Server

Observação

Os comandos do SQL Server 2019 a seguir apontam para o repositório do Ubuntu 20.04. Se você estiver usando o Ubuntu 18.04 ou 16.04, altere o caminho abaixo para /ubuntu/18.04/ ou /ubuntu/16.04/ em vez de /ubuntu/20.04/.

Para configurar o SQL Server no Ubuntu, execute os seguintes comandos em um terminal para instalar o pacote **mssql-server**.

1. Importe as chaves GPG do repositório público:

```
wget -q0-
https://packages.microsoft.com/keys/microsoft.
asc | sudo apt-key add microsoft.asc
```

2. Registre o repositório do Ubuntu do Microsoft SQL Server para o SQL Server 2019:

Para o Ubuntu 16.04:

```
Bash

sudo add-apt-repository "$(wget -q0-https://packages.microsoft.com/config/ubuntu/16.04/mssql-server-2019.list)"
```

Para o Ubuntu 18.04:

```
Bash

Sudo add-apt-repository "$(wget -q0-https://packages.microsoft.com/config/ubuntu/18.04/mssql-server-2019.list)"
```

Para o Ubuntu 20.04:

```
Bash

Sudo add-apt-repository "$(wget -q0-https://packages.microsoft.com/config/ubuntu/20.04/mssql-server-2019.list)"
```

3. Execute os comandos a seguir para instalar o SQL Server:

```
Bash

sudo apt-get update
sudo apt-get install -y mssql-server
```

4. Após a conclusão da instalação do pacote, execute a instalação de mssql-conf e siga os prompts para definir a senha SA e escolher sua edição.

```
Bash

sudo /opt/mssql/bin/mssql-conf setup
```

• Observação

Especifique uma senha forte para a conta SA (comprimento mínimo de 8 caracteres, incluindo letras maiúsculas e minúsculas, 10 dígitos básicos e/ou símbolos não alfanuméricos).

5. Após concluir a configuração, verifique se o serviço está em execução:

Bash Copiar systemctl status mssql-server --no-pager

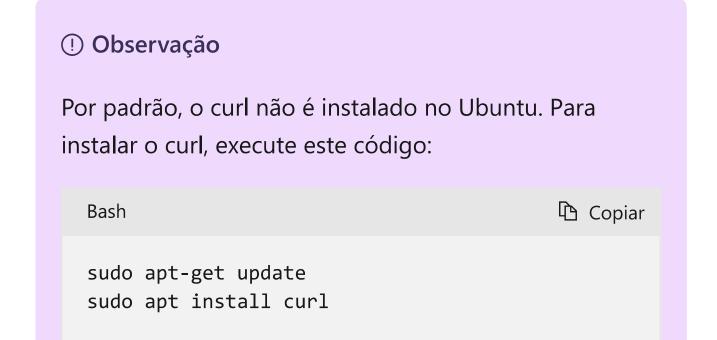
6. Se planeja se conectar remotamente, talvez seja preciso abrir a porta TCP do SQL Server (padrão 1433) em seu firewall.

Neste momento, o SQL Server 2019 está em execução no seu computador Ubuntu e está pronto para uso!

Instalar as ferramentas de linha de comando do SQL Server

Para criar um banco de dados, é necessário conectar-se a uma ferramenta que pode executar instruções Transact-SQL no SQL Server. As seguintes etapas instalam as ferramentas de linha de comando do SQL Server: sqlcmd e bcp.

Use as etapas a seguir para instalar o mssql-tools no Ubuntu.



1. Importe as chaves GPG do repositório público.

F . .

```
Bash

curl
https://packages.microsoft.com/keys/microsoft.
asc | sudo apt-key add -
```

2. Registre o repositório do Microsoft Ubuntu.

Para o Ubuntu 16.04:

```
Curl
https://packages.microsoft.com/config/ubuntu/1
6.04/prod.list | sudo tee
/etc/apt/sources.list.d/msprod.list
```

Para o Ubuntu 18.04:

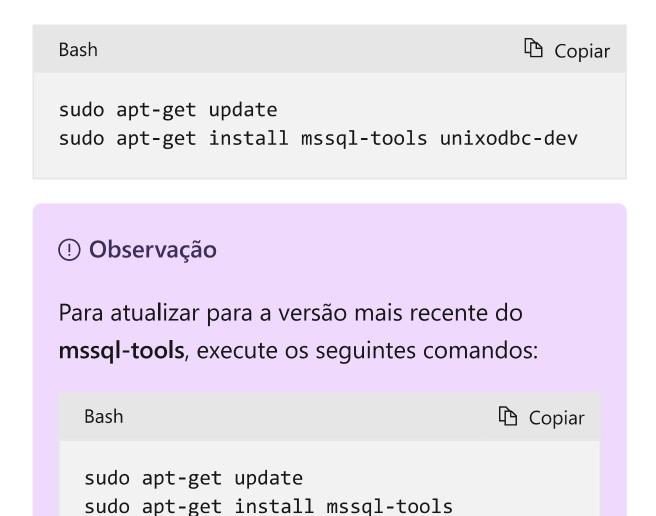
```
curl
https://packages.microsoft.com/config/ubuntu/1
8.04/prod.list | sudo tee
/etc/apt/sources.list.d/msprod.list
```

Para o Ubuntu 20.04:

```
Bash

curl
https://packages.microsoft.com/config/ubuntu/2
0.04/prod.list | sudo tee
/etc/apt/sources.list.d/msprod.list
```

3. Atualize a lista de fontes e execute o comando de instalação com o pacote do desenvolvedor do unixODBC. Para obter mais informações, confira Instalar o Microsoft ODBC Driver for SQL Server (Linux).



4. **Opcional**: Adicione /opt/mssql-tools/bin/ à sua variável de ambiente **PATH** em um shell de Bash.

Para tornar o **sqlcmd/bcp** acessível do shell de Bash para sessões de logon, modifique o **PATH** no arquivo **~/.bash_profile** com o seguinte comando:



```
echo 'export PATH="$PATH:/opt/mssql-
tools/bin"' >> ~/.bash_profile
```

Para tornar o **sqlcmd/bcp** acessível do shell de Bash para sessões interativas/que não são de logon, modifique o **PATH** no arquivo **~/.bashrc** com o seguinte comando:

```
Bash

echo 'export PATH="$PATH:/opt/mssql-
tools/bin"' >> ~/.bashrc
source ~/.bashrc
```

Conectar-se localmente

As etapas a seguir usam o **sqlcmd** para conectar-se localmente à nova instância do SQL Server.

1. Execute o **sqlcmd** com parâmetros para o nome do SQL Server (-S), o nome de usuário (-U) e a senha (-P). Neste tutorial, você está se conectando localmente, portanto, o nome do servidor é localhost. O nome de usuário é SA e a senha é a mesma fornecida para a conta SA durante a instalação.



É possível omitir a senha na linha de comando para receber uma solicitação para inseri-la.

□ Dica

Se depois você decidir se conectar remotamente, especifique o nome do computador ou endereço IP do parâmetro **-S** e verifique se a porta 1433 está aberta no firewall.

- 2. Se isso funcionar, você será levado a um prompt de comando **sqlcmd**: 1>.
- 3. Se houver uma falha de conexão, primeiro, tente diagnosticar o problema da mensagem de erro. Em seguida, examine as recomendações de solução de problemas de conexão.

Criar e consultar dados

As seções a seguir descrevem como usar o **sqlcmd** para criar um novo banco de dados, adicionar dados e executar uma consulta simples.

Criar um novo banco de dados

As etapas a seguir criam um novo banco de dados denominado TestDB.

 No prompt de comando sqlcmd, cole o seguinte comando Transact-SQL para criar um banco de dados de teste:

```
SQL Copiar

CREATE DATABASE TestDB
```

2. Na próxima linha, grave uma consulta para retornar o nome de todos os bancos de dados do servidor:

```
SQL Copiar

SELECT Name from sys.Databases
```

3. Os dois comandos anteriores não foram executados imediatamente. Digite 60 em uma nova linha para executar os comandos anteriores:



Para saber mais sobre como escrever consultas e instruções em Transact-SQL, veja **Tutorial: Escrever**

instruções de Transact-SQL.

Inserir dados

Em seguida, crie uma nova tabela, Inventory, e insira duas novas linhas.

1. No prompt de comando **sqlcmd**, altere o contexto para o novo banco de dados TestDB:

```
SQL Copiar

USE TestDB
```

2. Criar nova tabela denominada Inventory:

```
CREATE TABLE Inventory (id INT, name NVARCHAR(50), quantity INT)
```

3. Inserir dados na nova tabela:

```
INSERT INTO Inventory VALUES (1, 'banana', 150); INSERT INTO Inventory VALUES (2, 'orange', 154);
```

4. Digite 60 para executar os comandos anteriores:

```
SQL Copiar

GO
```

Selecionar dados

Agora, execute uma consulta para retornar da tabela Inventory.

1. No prompt de comando **sqlcmd**, digite uma consulta que retorna linhas de tabela Inventory em que a quantidade é maior que 152:

```
SQL Copiar

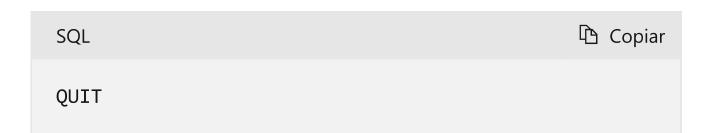
SELECT * FROM Inventory WHERE quantity > 152;
```

2. Execute o comando:



Saia do prompt de comando sqlcmd

Para encerrar a sessão sqlcmd, digite QUIT:



Práticas recomendadas de desempenho

Após instalar o SQL Server em Linux, examine as práticas recomendadas para configuração do Linux e do SQL Server a fim de melhorar o desempenho para cenários de produção. Para obter mais informações, confira Práticas recomendadas de desempenho e diretrizes de configuração do SQL Server em Linux.

Ferramentas de dados multiplataforma

Além de **sqlcmd**, você pode usar as seguintes ferramentas multiplataforma para gerenciar o SQL Server:

Ferramenta	Descrição
Azure Data Studio	Um utilitário de gerenciamento de banco de dados GUI multiplataforma.
Visual Studio Code	Um editor de código GUI multiplataforma que executa instruções Transact-SQL com a extensão mssql.
PowerShell Core	Uma ferramenta de automação e configuração multiplataforma baseada em cmdlets.

Ferramenta	Descrição
mssql-cli ☑	Uma interface de linha de comando multiplataforma para executar comandos Transact-SQL.

Conectando do Windows

As ferramentas do SQL Server no Windows se conectam às instâncias do SQL Server no Linux da mesma forma que se conectam a qualquer instância remota do SQL Server.

Se você tiver um computador Windows que pode se conectar a um computador Linux, tente as mesmas etapas deste tópico de um prompt de comando do Windows executando o sqlcmd. Apenas certifique-se de que está utilizando o nome de destino do computador Linux ou o endereço IP em vez do localhost e de que a porta TCP 1433 está aberta. Se houver problemas ao se conectar do Windows, consulte recomendações de solução de problemas de conexão.

Para outras ferramentas que executam o Windows, mas se conectam ao SQL Server no Linux, consulte:

- SQL Server Management Studio (SSMS)
- Windows PowerShell
- SSDT (SQL Server Data Tools)

Outros cenários de implantação

Para outros cenários de instalação, veja os seguintes recursos:

- Atualização: Saiba como atualizar uma instalação existente do SQL Server no Linux
- Desinstalação: Desinstalar o SQL Server no Linux
- Instalação autônoma: Saiba como gerar o script da instalação sem prompts
- Instalação offline: Saiba como baixar manualmente os pacotes para instalação offline

□ Dica

Para obter respostas a perguntas frequentes, confira as **Perguntas frequentes sobre o SQL Server em Linux**.

Próximas etapas

Explorar os tutoriais do SQL Server em Linux

Conteúdo recomendado

Diretrizes de instalação para SQL Server em Linux - SQL Server

Instale, atualize e desinstale o SQL Server em Linux. Este artigo aborda cenários online, offline e autônomos.

Usar o SSMS para gerenciar o SQL Server em Linux - SQL Server

Este artigo apresenta o SQL Server Management Studio, um ambiente integrado para acessar, configurar, gerenciar, administrar e desenvolver componentes do SQL Server.

Solução de problemas do SQL Server em Linux - SQL Server

Solucione problemas no Microsoft SQL Server em execução no Linux ou em um contêiner do Docker. Saiba onde encontrar informações sobre os recursos com suporte e as...

Migrar um banco de dados do SQL Server do Windows para o Linux - SQL Server

Este tutorial mostra como tomar um backup de banco de dados do SQL Server em Windows e restaurá-lo em um computador Linux que executa o SQL Server.

Mostrar mais ∨