

# Programação em Banco de Dados

Faculdade Anhanguera – Unidade Betim/MG

**AULA 05**

# # Aula 05

- Processamento de Consultas
- Processamento de Bloqueios

# # Introdução

- No SGBD *MySQL* os *locks* (bloqueios) são *mecanismos* utilizados para *controlar* o acesso *concorrente* a dados, garantindo a *integridade* e o *isolamento* das *transações*.
- Estes *mecanismos* evitam *problemas* como condições de *corrida*, leituras *sujas* e atualizações *conflitantes*.
- Os principais *locks* são: Nível de Tabela (*Table Locks*), Nível de Linha (*Row Locks*), Nível de Página (*Page Locks*) e Nível de Banco de Dados (*Database*)

# # Sobre MyISAM / InnoDB



- São mecanismos de armazenamento (*Storage Engine*) disponíveis no SGBD MySQL.
- O um *Storage Engine* é aplicado as **tabelas** do SGBD MySQL.
- MyISAM - **My** (referente ao MySQL) e **ISAM** (*Indexed Sequential Access Method*), que é um **método tradicional** de organização de arquivos **indexados**.

# Sobre MYISAM

- O **ISAM** foi um dos primeiros *storage* projetado para permitir acesso **rápido** a registros em **arquivos grandes**.
- O **MyISAM**, por sua vez, é uma **versão aprimorada** desse **conceito**, desenvolvida **especificamente** para o MySQL.
- Ele se destaca pelo uso de **índices** para acelerar buscas e consultas, mas **não suporta transações** nem **integridade referencial**, diferentemente do InnoDB.

# # Características MyISAM



1. **Desempenho Rápido** - É otimizado para consultas de leitura (*SELECT*), tornando-o ideal para aplicações onde a leitura de dados é mais frequente do que a escrita.
2. **Não Suporta Transações** - Diferente do InnoDB, o MyISAM não suporta transações, *ROLLBACK* ou integridade referencial.
3. **Bloqueio de Tabela** - Ao invés de bloqueio em nível de linha (como o InnoDB), ele bloqueia a tabela inteira durante operações de escrita, o que pode causar problemas de concorrência em sistemas com muitas atualizações simultâneas.

# # Características MyISAM

4. **Arquivos:** Cada tabela MyISAM é **armazenada** em três arquivos:

- .frm → **Estrutura da tabela**
- .MYD → **Dados**
- .MYI → **Índices**

# # Prática 1

1. *CREATE DATABASE* Seu\_Nome;
2. *USE* Seu\_Nome;
3. *CREATE TABLE* CLIENTE  
( *CODIGO INT AUTO\_INCREMENT,*  
*NOME VARCHAR(50) NOT NULL,*  
*EMAIL TEXT NOT NULL,*  
*DATA DATETIME NOT NULL,*  
*PRIMARY KEY(CODIGO) );*



# # Prática 1

4. *INSERT INTO* CLIENTE (NOME,EMAIL,DATA) *VALUES* ('CMN',  
'CMN@CMN.BR', NOW() );
5. *INSERT INTO* CLIENTE (NOME,EMAIL,DATA) *VALUES* ('ANA',  
'ANA@ANA.BR', NOW() );
6. *INSERT INTO* CLIENTE (NOME,EMAIL, DATA) *VALUES* ('BETO',  
'BETO@ANA.BR', NOW() );
7. *SELECT* \* *FROM* CLIENTE;
8. *SHOW TABLE STATUS WHERE* NAME = 'CLIENTE';
9. *ALTER TABLE* CLIENTE *ENGINE* = MyISAM;

# # Sobre InnoDB

- É um *storage engine* (*mecanismo de armazenamento*) do MySQL que se destaca pelo **suporte a transações, integridade referencial e concorrência** eficiente.
- *Storage engine* padrão a partir do SGBD MySQL 5.5.
- A palavra/termo **InnoDB** é uma combinação de *innovative* (*inovador*) e *database* (DB) que reflete a nova proposta de *storage engine* para o SGBD MySQL.

# # Tabela MyISAM x InnoDB

Característica	MyISAM	InnoDB
Transações	Não suporta	Suporta (COMMIT e ROLLBACK)
Chaves estrangeiras	Não suporta	Suporta (FOREIGN KEY)
Bloqueio	Tabela inteira	Linha específica
Velocidade	Mais rápido para leitura	Melhor para leitura e escrita simultâneas
Recuperação após falhas	Não confiável	Suporta recuperação automática

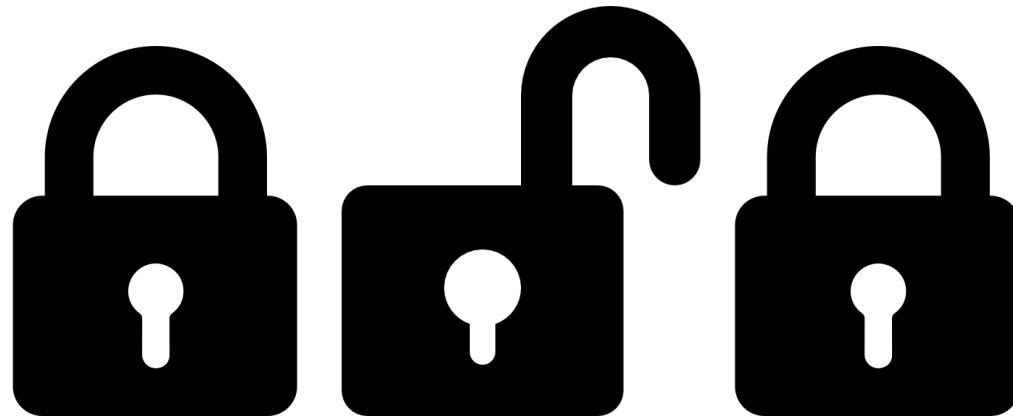
# # Prática 2

1. *CREATE TABLE PEDIDO (*  
    *NUMERO INT AUTO\_INCREMENT PRIMARY KEY,*  
    *CLIENTE INT,*  
    *VALOR DECIMAL(10,2),*  
    *FOREIGN KEY (CLIENTE) REFERENCES CLIENTE (CODIGO))*  
*ENGINE=InnoDB;*
2. *SHOW TABLE STATUS WHERE NAME = 'CLIENTE';*
3. *ALTER TABLE CLIENTE ENGINE = InnoDB;*

# # Prática 2

4. *INSERT INTO PEDIDO VALUES (2025,1,1.99),  
(2026,2,150.90), (2027,3,9.99);*
5. *SELECT \* FROM CLIENTE, PEDIDO  
WHERE CLIENTE.CODIGO = PEDIDO.CLIENTE;*

# *Mecanismos de Locks no SGBD*



# # Nível de Tabela - *table locks*

- Realiza o bloqueio de **toda a tabela** para **leitura** ou **escrita**.
- *Storage engine* MyISAM (**não** suporta **locks** de nível de **linha**).
- **READ LOCK** permite **múltiplas** leituras **simultâneas**, mas **bloqueia** escritas.
- **WRITE LOCK** impede **qualquer** outra **operação** na **tabela** até a liberação do **lock**.

## # Nível de página - *page locks*

- Este mecanismo **bloqueia páginas** de dados em vez de **linhas** ou **tabelas** inteiras.
- **Não** é usado no **MySQL**, mas está **presente** em outros **SGBDs** como o **Oracle** e o **SQL Server**.



## # Nível de BD - *database locks*

- Este mecanismo **bloqueia** o **banco de dados** inteiro para **evitar modificações concorrentes** em tabelas **interligadas**.
- Atenção! Este mecanismo é **pouco utilizado**, pois **impacta** muito a **performance**.

# # *Locks* Explícitos vs. Implícitos

- *Locks* Explícitos
  - O usuário define **manualmente** os bloqueios (*LOCK TABLES*, *SELECT ... FOR UPDATE*).
- *Locks* Implícitos
  - O MySQL **automaticamente bloqueia** recursos quando necessário, **dependendo** do **nível** de **isolamento** da transação.

# # Prática 03

- Criar uma nova instância do SGBD
- Realizar o bloqueio *WRITE LOCK*
- Testar comandos



# Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

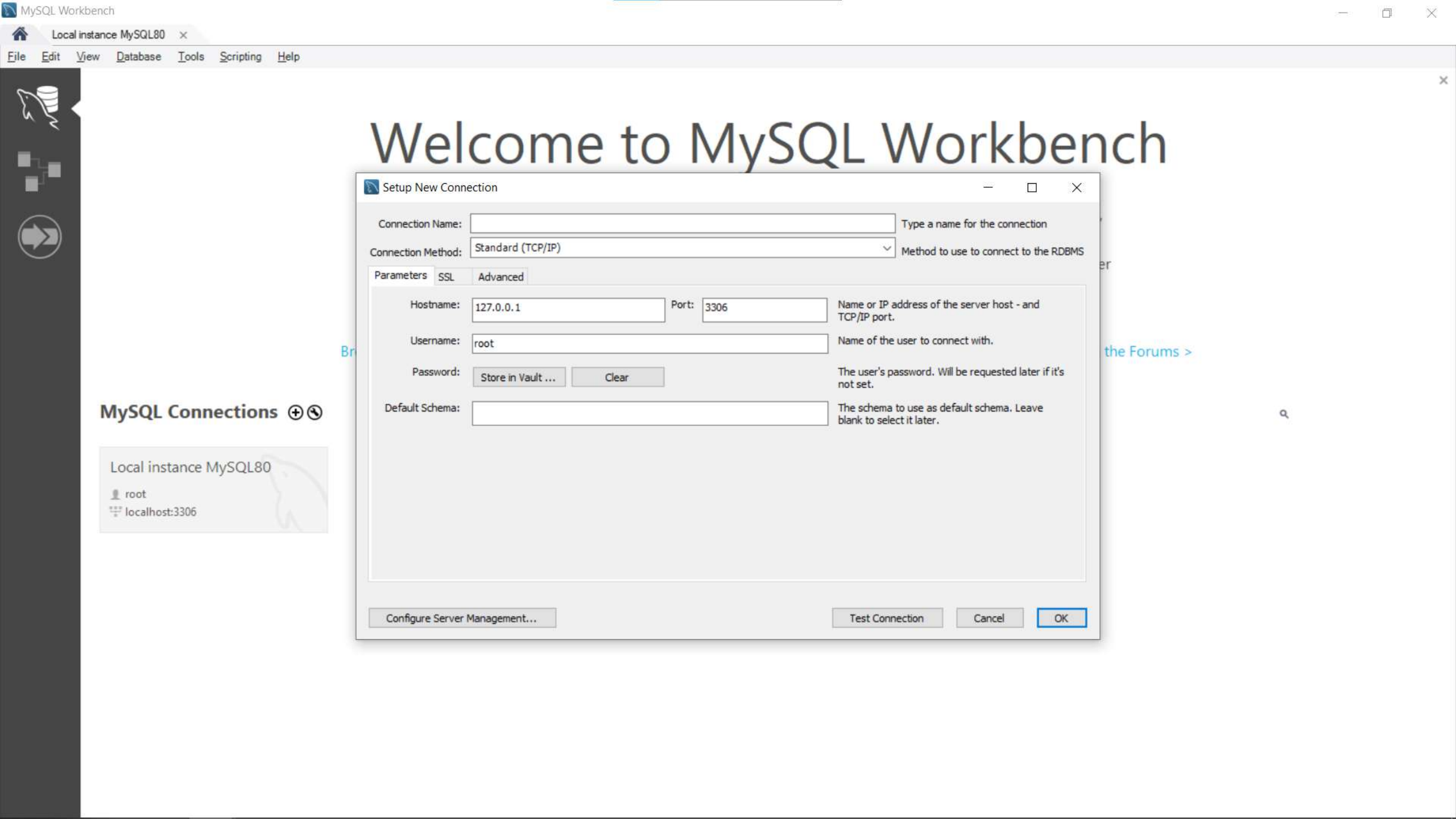
[Browse Documentation >](#)[Read the Blog >](#)[Discuss on the Forums >](#)

## MySQL Connections



Local instance MySQL80

 root  
 localhost:3306



# Welcome to MySQL Workbench

## Setup New Connection

Connection Name:  Type a name for the connection

Connection Method:  Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname:  Port:  Name or IP address of the server host - and TCP/IP port.

Username:  Name of the user to connect with.

Password:   The user's password. Will be requested later if it's not set.

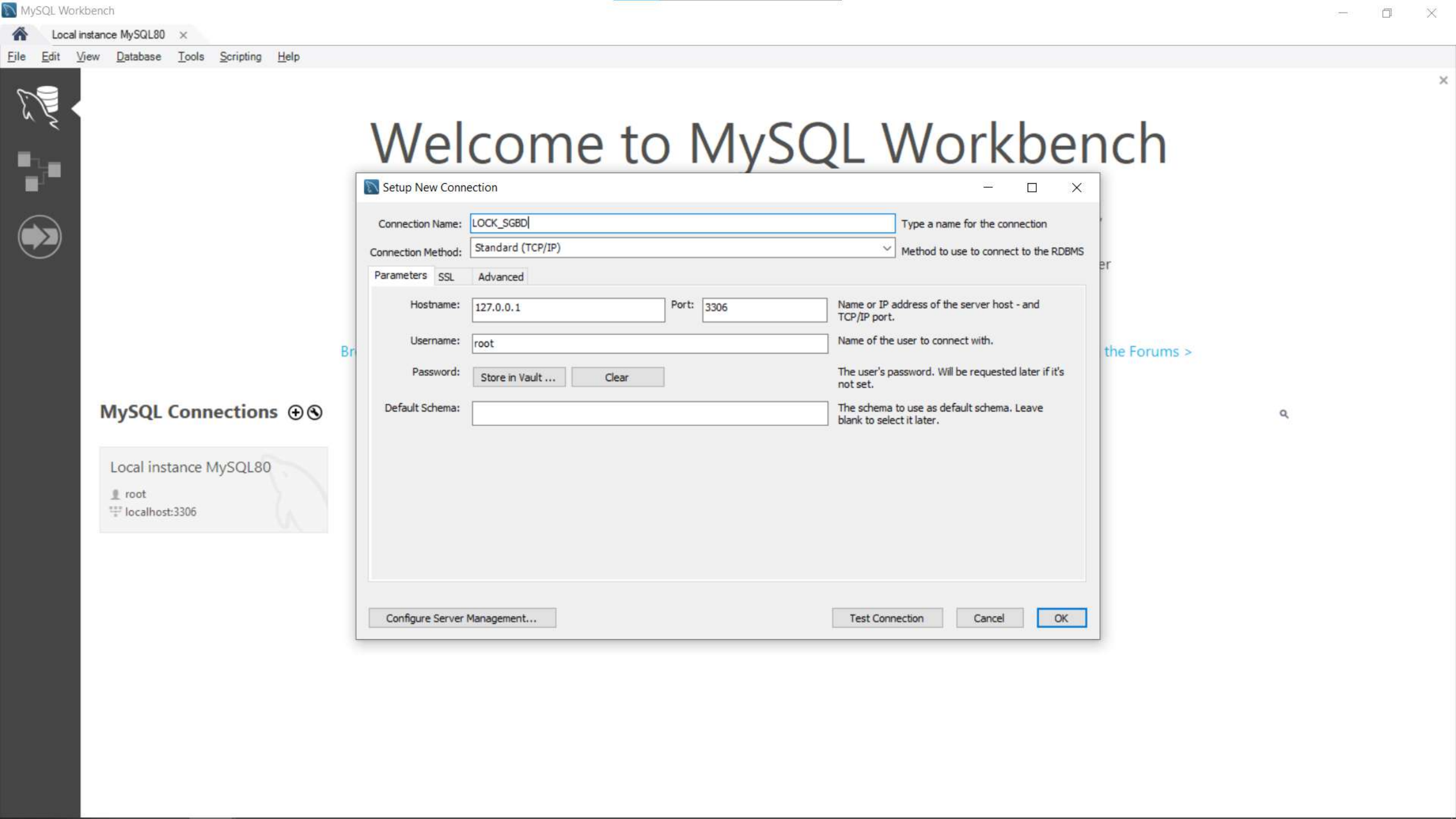
Default Schema:  The schema to use as default schema. Leave blank to select it later.

[Configure Server Management...](#)[Test Connection](#)[Cancel](#)[OK](#)

## MySQL Connections + ⓘ

Local instance MySQL80

root  
localhost:3306



## MySQL Connections + ⓘ

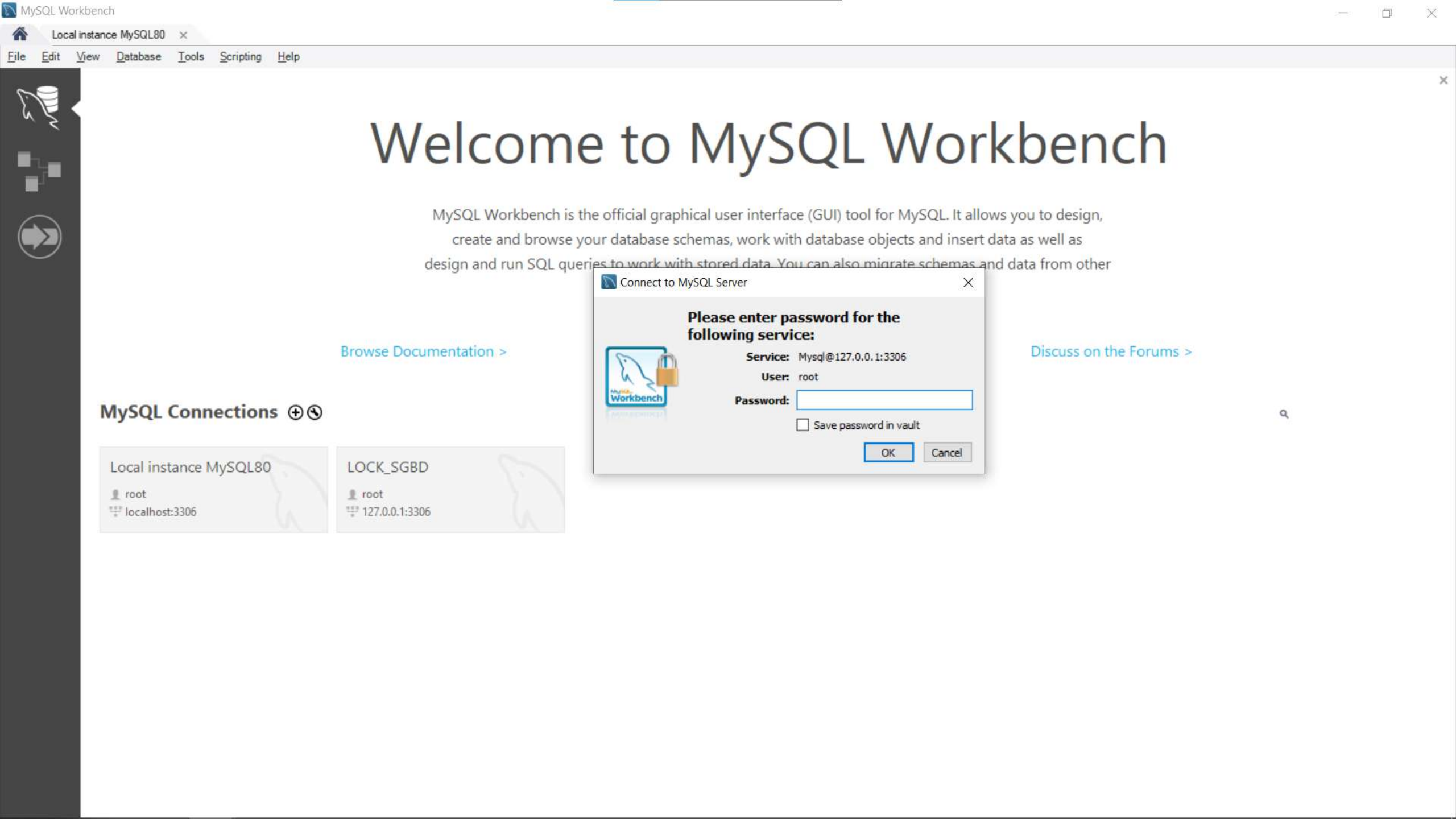
Local instance MySQL80

root  
localhost:3306

## Welcome to MySQL Workbench

## Setup New Connection

Connection Name:	<input type="text" value="LOCK_SGBD"/>	Type a name for the connection
Connection Method:	<input type="text" value="Standard (TCP/IP)"/>	Method to use to connect to the RDBMS
<div>Parameters SSL Advanced</div>		
Hostname:	<input type="text" value="127.0.0.1"/>	Port: <input type="text" value="3306"/>
		Name or IP address of the server host - and TCP/IP port.
Username:	<input type="text" value="root"/>	
		Name of the user to connect with.
Password:	<input type="button" value="Store in Vault ..."/> <input type="button" value="Clear"/>	
		The user's password. Will be requested later if it's not set.
Default Schema:	<input type="text"/>	
		The schema to use as default schema. Leave blank to select it later.



# Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other

[Browse Documentation >](#)[Discuss on the Forums >](#)

## MySQL Connections + ⓘ

Local instance MySQL80


root  
localhost:3306

LOCK\_SGBD

root  
127.0.0.1:3306

Connect to MySQL Server

Please enter password for the following service:



Service: Mysql@127.0.0.1:3306

User: root

Password:

☐ Save password in vault

OK Cancel

# # Prática 03

## 1. Sessão Local Host

*USE* Seu\_Nome;

*LOCK TABLES* CLIENTE *WRITE*;

## 2. Sessão LOCK\_SGBD

1. *SELECT* \* *FROM* CLIENTE;

2. *UPDATE* CLIENTE *SET* NOME = 'AULA 5' *WHERE* CODIGO = 1;

3. *INSERT INTO* CLIENTE *VALUES* (999, 'Meu Teste', 'aula@aula',  
NOW());

4. *DELETE FROM* CLIENTE;



# # Prática 03

## 3. Sessão Local Host

*UNLOCK TABLES;*

## 4. Sessão LOCK\_SGBD

1. *SELECT \* FROM CLIENTE;*

2. *UPDATE CLIENTE SET NOME = 'AULA 5' WHERE CODIGO = 1;*

3. *INSERT INTO CLIENTE VALUES (999, 'Betim MG', 'aula@aula', NOW());*

4. *DELETE FROM CLIENTE;*

# # Prática 04 – Agora é c/vocês...

1. Como realizar o bloqueio de linha?
2. Como realizar o bloqueio de leitura?
3. UNLOCK TABLES; (Não desligar s/verificar)