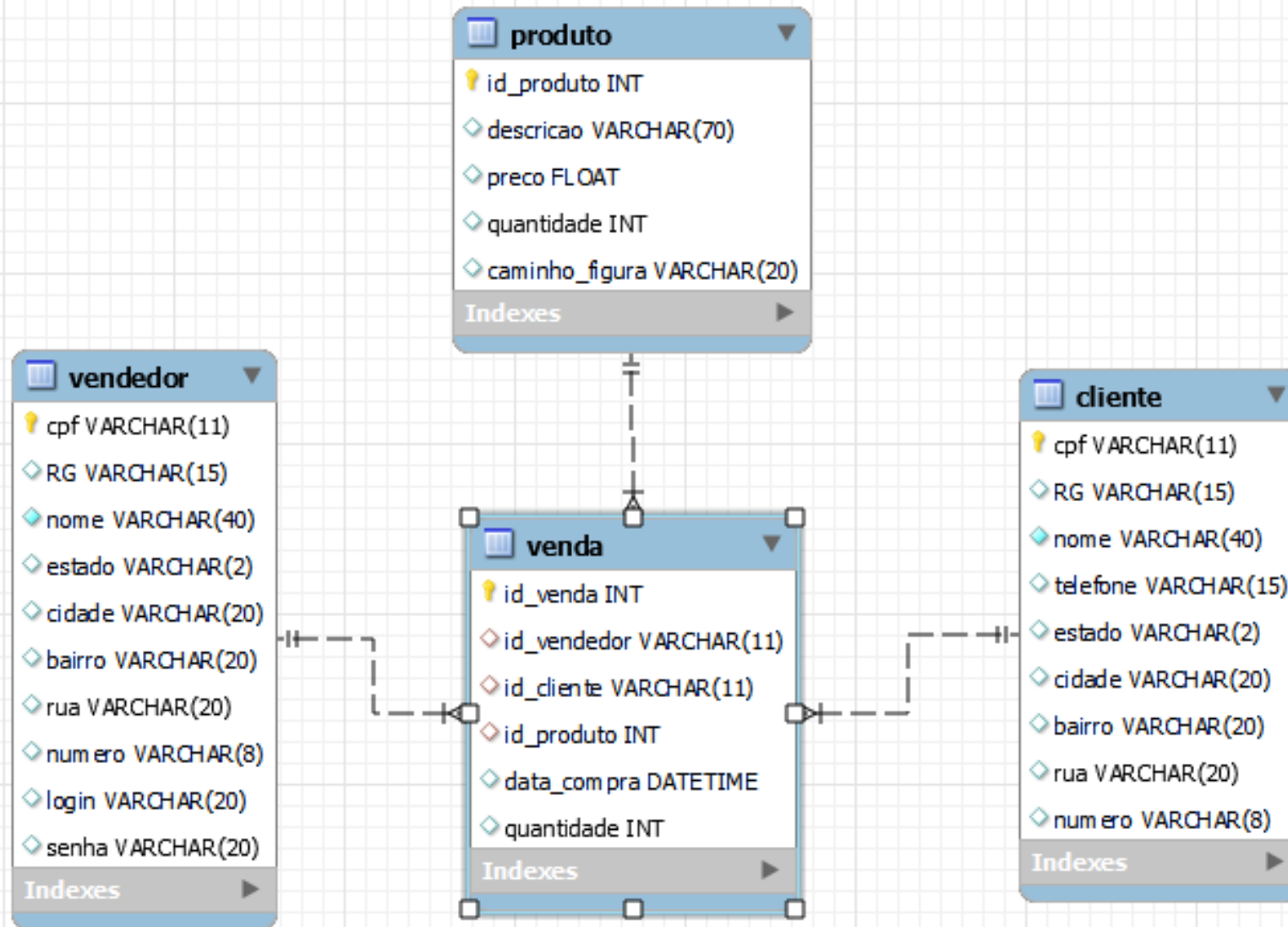


Aula de banco de dados voltada para o Projeto de Lab. De Programação



Criação de Um Banco de Dados

► create database Loja_moveis

Selecionando o banco que será utilizado

► use Loja_moveis

Criando a tabela de Vendedores

```
create table Vendedor (  
    cpf varchar(11) NOT NULL,  
    RG varchar(15),  
    nome varchar(40) NOT NULL,  
    estado varchar(2),  
    cidade varchar(20),  
    bairro varchar(20),  
    rua varchar(20),  
    numero varchar(8),  
    login varchar(20),  
    senha varchar (20),  
    PRIMARY KEY (cpf)  
);
```

Criando a tabela de clientes

```
create table Cliente (  
    cpf varchar(11) NOT NULL,  
    RG varchar(15),  
    nome varchar(40) NOT NULL,  
    estado varchar(2),  
    cidade varchar(20),  
    bairro varchar(20),  
    rua varchar(20),  
    numero varchar(8),  
    PRIMARY KEY (cpf)  
);
```

Criando a tabela de Produtos

```
create table Produto (  
    id_produto int not null auto_increment,  
    descricao varchar(60),  
    preco float,  
    caminho_figura varchar(20),  
    PRIMARY KEY (id_produto)  
);
```

Observação

- ▶ Não é obrigatório utilizar figuras para ilustrar os produtos disponíveis.
- ▶ Mas se o seu grupo optar por fazer, pode-se utilizar o tipo Blob para armazenar figuras no banco de dados, ou pode ser feito como no slide anterior, armazenar apenas o caminho (ou parte dele).

Criando a tabela de vendas

```
create table Venda(  
  id_venda int not null auto_increment,  
  id_vendedor varchar(11),  
  id_cliente varchar(11),  
  id_produto int,  
  primary key(id_venda),  
  foreign key (id_vendedor) references Vendedor(cpf),  
  foreign key (id_cliente) references Cliente(cpf),  
  foreign key (id_produto) references Produto(id_produto)  
);
```

Testando inserções na tabela vendedor

```
insert into Vendedor values('11111111111',  
                             'MG-11.100.111',  
                             'Aline Souza Lima',  
                             'MG',  
                             'Belo Horizonte',  
                             'União',  
                             'Rua do Campo',  
                             125,  
                             'aline_sl',  
                             'vendedora_01');
```

Testando inserções na tabela vendedor

```
insert into Vendedor values('22222222222',  
                             'MG-22.200.222',  
                             'Augusto Gomes Alves',  
                             'MG',  
                             'Belo Horizonte',  
                             'Nova Gameleira',  
                             'Rua Eldorado',  
                             741,  
                             'augusto_ga',  
                             'vendedor_02');
```

Consultando na tabela Vendedor

▶ `select * from Vendedor`

Resultado da Consulta

[illegible]

Consultando apenas o nome de um vendedor específico

```
1 • select nome from Vendedor  
2   where cpf = '11111111111'  
3
```

<	
Result Grid	
	Filter Rows:
	nome
▶	Aline Souza Lima

Atualizando o endereço de um vendedor

```
update Vendedor  
set rua = 'Palmares', bairro = 'Santa Maria'  
where cpf = '222222222222'
```

Consultando se a atualização ocorreu corretamente

```
1 • select * from vendedor
2   where cpf = '22222222222'
3
4
5
```

Result Grid							
		Filter Rows:		Edit:		Export/Import:	
	cpf	RG	nome	estado	cidade	bairro	rua
▶	22222222222	MG-22.200.222	Augusto Gomes Alves	MG	Belo Horizonte	Santa Maria	Palmares
•	NULL	NULL	NULL	NULL	NULL		

Alterando a estrutura de uma tabela

```
ALTER TABLE Cliente
```

```
ADD COLUMN telefone VARCHAR(15) AFTER nome;
```

Select * from Cliente, para ver se a alteração foi feita com sucesso

[illegible]

Utilizando o Update

```
update cliente  
set telefone = '(31)94444-4444'  
where cpf = '444444444444'
```

Consulta com like

```
1  ●  SELECT *
2      FROM Cliente
3      WHERE nome LIKE '%Flavia%'
4
5
```

[illegible]

Acrescentando a coluna com a
quantidade de Produtos

```
ALTER TABLE Produto  
ADD COLUMN quantidade int AFTER preco;
```

Inserindo um produto

```
insert into Produto (descricao, preco)  
values('sofá azul de 3 lugares', '1500')
```


Inserindo um novo produto

```
insert into Produto (descricao, preco)  
values('Mesa de vidro com 6 cadeiras', '1100.50')
```

Inserindo um novo produto

```
insert into Produto (descricao, preco)  
values('Guarda-roupa de casal', '3399.99')
```

Inserindo um novo produto

```
insert into Produto (descricao, preco)  
values('Guarda-roupa de solteiro', '2599.99')
```

Atualizando todos os itens para 10 quantidades

```
update Produto  
set quantidade = 10  
where id_produto = 1  
or id_produto =2  
or id_produto =3  
or id_produto =4
```

Aumentando o tamanho de um campo




```
alter table Produto  
modify descricao varchar(70);
```

select * from Produto

id_produto	descricao	preco	quantidade	caminho_figura
1	sofá azul de 3 lugares	1500	10	NULL
2	Mesa de vidro com 6 cadeiras	1100.5	10	NULL
3	Guarda-roupa de casal	3399.99	10	NULL
4	Guarda-roupa de solteiro	2599.99	10	NULL
NULL	NULL	NULL	NULL	NULL

Consulta mais específica

```
1 • select * from produto
2   where descricao like '%guarda-roupa%'
3   and preco between 500 and 3000
4
5
```

Result Grid				
Filter Rows: <input type="text"/>				
Edit:    Export/Import				
id_produto	descricao	preco	quantidade	caminho_figura
4	Guarda-roupa de solteiro	2599.99	10	NULL
NULL	NULL	NULL	NULL	NULL

Adicionando uma coluna para armazenar a data e hora das vendas

```
alter table Venda
```

```
add column data_compra datetime
```


Adicionando uma nova coluna para armazenar a quantidade de um item comprado

```
alter table Venda  
add column quantidade int
```

Inserindo uma venda

```
insert into Venda (id_vendedor,  
id_cliente,  
id_produto,  
quantidade,  
data_compra)  
values('11111111111', '333333333333', 1, 2, now())
```

Atualizando a quantidade em estoque

```
update produto  
set quantidade = quantidade - 2  
where id_produto = 1
```

Conferindo





1 ● `select * from produto`

2

3

4

5

result Grid			
  Filter Rows: <input type="text"/> Edit:  			
id_produto	descricao	preco	quantidade
1	sofá azul de 3 lugares	1500	8
2	Mesa de vidro com 6 cadeiras	1100.5	10
3	Guarda-roupa de casal	3399.99	10
4	Guarda-roupa de solteiro	2599.99	10
NULL	NULL	NULL	NULL

Inserindo uma nova venda

```
insert into Venda (id_vendedor,  
id_cliente,  
id_produto,  
quantidade,  
data_compra)  
values('222222222222','333333333333',2,1,now())
```

Atualizando a quantidade em estoque

```
update produto  
set quantidade = quantidade - 1  
where id_produto = 2
```

Conferindo

1 • `select * from produto`

2

3

4

5

Result Grid



Filter Rows:

Edit:



id_produto	descricao	preco	quantidade
1	sofá azul de 3 lugares	1500	8
2	Mesa de vidro com 6 cadeiras	1100.5	9
3	Guarda-roupa de casal	3399.99	10
4	Guarda-roupa de solteiro	2599.99	10
NULL	NULL	NULL	NULL

Inserindo uma nova venda

```
insert into Venda (id_vendedor,  
id_cliente,  
id_produto,  
quantidade,  
data_compra)  
values('222222222222','333333333333',4,1,now())
```


Atualizando a quantidade em estoque

```
update produto  
set quantidade = quantidade - 1  
where id_produto = 4
```

Conferindo

```
1 • select * from produto
```

```
2
```

```
3
```

```
4
```

```
5
```

<

Result Grid

Filter Rows:

Edit:

	id_produto	descricao	preco	quantidade
▶	1	sofá azul de 3 lugares	1500	8
	2	Mesa de vidro com 6 cadeiras	1100.5	9
	3	Guarda-roupa de casal	3399.99	10
	4	Guarda-roupa de solteiro	2599.99	9
●	NULL	NULL	NULL	NULL

Conferindo as vendas

```
1 • select * from venda
```

```
2
```

```
3
```

```
4
```

```
5
```

<

Result Grid |   Filter Rows: | Edit:    | Export/Import: 

	id_venda	id_vendedor	id_cliente	id_produto	data_compra	quantidade
▶	1	1111111111	3333333333	1	2023-06-15 00:53:32	2
	2	2222222222	4444444444	2	2023-06-15 00:57:00	1
	4	2222222222	3333333333	4	2023-06-15 00:58:30	1
✱	NULL	NULL	NULL	NULL	NULL	NULL

Soma das vendas

```
1 • select sum(p.preco * v.quantidade)
2   from produto as p, venda as v, vendedor as ve
3   where ve.cpf = v.id_vendedor
4   and p.id_produto = v.id_produto
5
```



Result Grid



Filter Rows:

Export:






Wrap


	sum(p.preco * v.quantidade)
▶	6700.489990234375

Vendas de um único vendedor

```
1 • select sum(p.preco * v.quantidade)
2   from produto as p, venda as v, vendedor as ve
3   where ve.cpf = v.id_vendedor
4   and p.id_produto = v.id_produto
5   and ve.cpf = '11111111111'
```

result Grid				Filter Rows: <input type="text"/>	Export: 	Wrap
sum(p.preco * v.quantidade)						
3000						

Conectando o Java com o MySQL

[General Availability \(GA\) Releases](#)[Archives](#)

Connector/J 8.0.33

Select Operating System:

Microsoft Windows ▼


Recommended Windows Download:

MySQL Installer for Windows

**All MySQL Products. For All Windows Platforms.
In One Package.**

Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI



[Go to Download Page >](#)

General Availability (GA) Releases

Archives



MySQL Installer 8.0.33

Select Operating System:

Microsoft Windows



Looking for previous GA versions?

Windows (x86, 32-bit), MSI Installer

(mysql-installer-web-community-8.0.33.0.msi)

8.0.33

2.4M

[Download](#)

MD5: 2a330cf24915964cca87e04dbb34e5d3 | [Signature](#)

Windows (x86, 32-bit), MSI Installer

(mysql-installer-community-8.0.33.0.msi)

8.0.33

428.3M

[Download](#)

MD5: 9b4ce33ab05ae7e0aa30a6c4f1a4d1c2 | [Signature](#)



We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

⬇ MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »

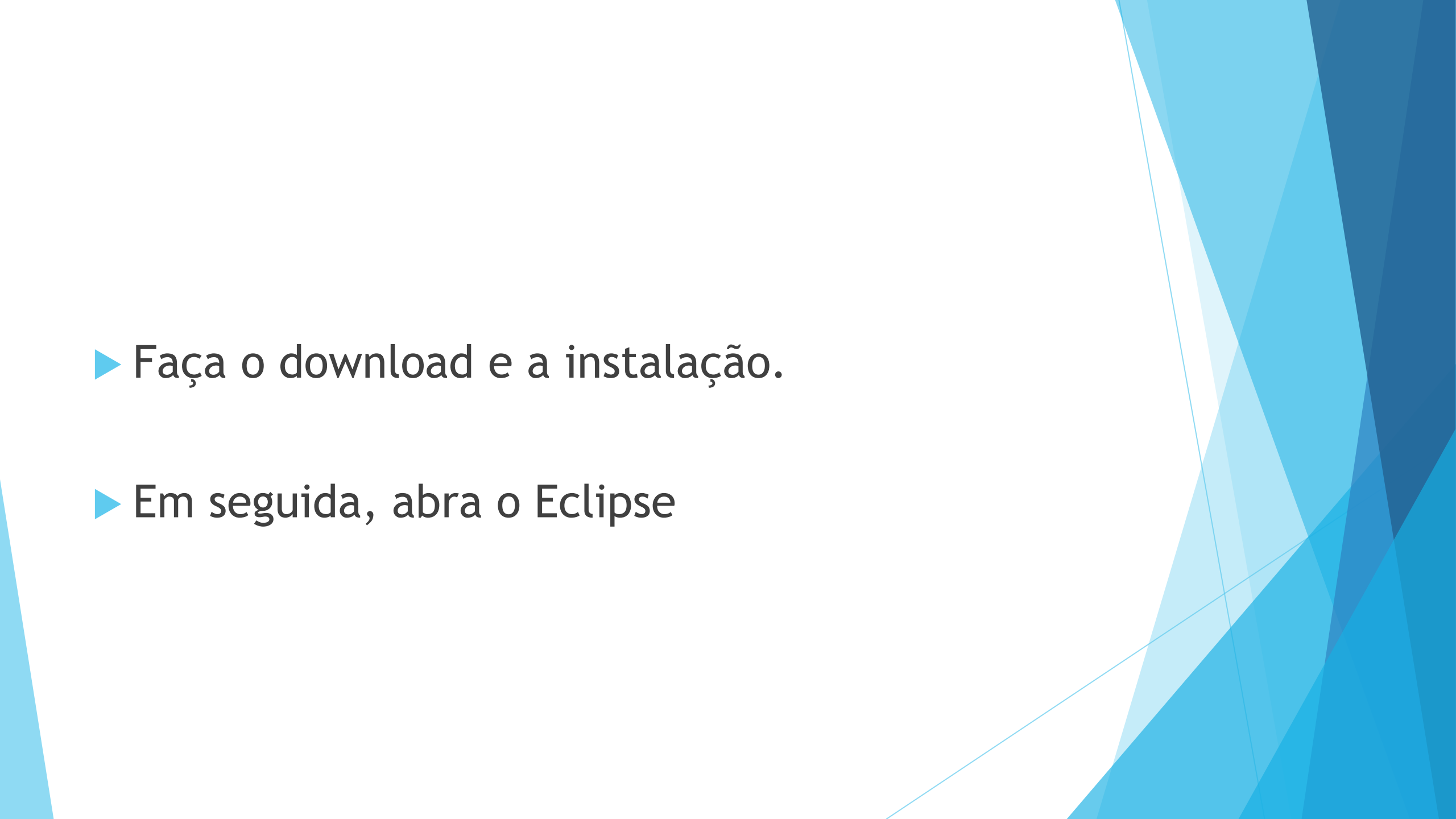
using my Oracle Web account

Sign Up »








for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can sign up for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

- 
- The background of the slide features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side of the slide, creating a modern, dynamic aesthetic.
- ▶ Faça o download e a instalação.
 - ▶ Em seguida, abra o Eclipse







- > Aluno_Ar
- > Atividade
- > Atividade
- > Atividade
- > Excecoes
- > Interface
- > LojaMove
- > Prefeito
- > Testes
- > TestesArq

New	>
Go Into	
Open in New Window	
Open Type Hierarchy	F4
Show In	Alt+Shift+W >
 Copy	Ctrl+C
 Copy Qualified Name	
 Paste	Ctrl+V
 Delete	Delete
Build Path	>
Source	Alt+Shift+S >
Refactor	Alt+Shift+T >
 Import...	
 Export...	
 Refresh	F5
Close Project	

```
"Conexao realizada com sucesso  
os();
```

```
x) {  
"SQL Exception: " + ex.getMess  
"Código do erro: " + ex.getErr
```

```
TodosProdutos() throws SQLExce  
iverManager.getConnection("jdb  
word=senha_acesso_banco");  
from produto";
```

	Link Source...
	New Source Folder...
	Use as Source Folder
	Add External Archives...
	Add Libraries...
	Configure Build Path...

- ▶ Em seguida, selecionar “Modulepath” e clicar em “Adicionar external Jar”.
- ▶ Selecionar o mysql-connector-j e clicar em abrir.
- ▶ Após esses passos será possível visualizar o arquivo .jar no projeto.

Properties for Lojalmoveis

type filter text

- > Resource
- Builders
- Coverage
- Java Build Path
- > Java Code Style
- > Java Compiler
- Javadoc Location
- > Java Editor
- Project Natures
- Project References
- Refactoring History
- Run/Debug Settings
- WikiText

Java Build Path

Source Projects Libraries Order and Export Module Dependencies

JARs and class folders on the build path:

- Modulepath
 - > mysql-connector-j-8.0.33.jar - C:\Program Files (x86)\MySQL\Connector J 8.0
 - > JRE System Library [JavaSE-19]
- Classpath

Add JARs...

Add External JARs...

Add Variable...

Add Library...

Add Class Folder...

Add External Class Folder...

```
package database;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import com.mysql.cj.xdevapi.Statement;
public class Conexao {

    public static void main(String[] args) {
        try {
            DriverManager.getConnection("jdbc:mysql://localhost/Loja_moveis?" +
                                       "user=root&password=senha_acesso_banco");

            System.out.println("Conexao realizada com sucesso.");
            consultaTodosProdutos();

        } catch (SQLException ex) {
            System.out.println("SQL Exception: " + ex.getMessage());
            System.out.println("Código do erro: " + ex.getErrorCode());
        }
    }
}
```

```
24 public static void consultaTodosProdutos() throws SQLException {  
25  
26     Connection conexao = DriverManager.getConnection("jdbc:mysql://localhost/Loja_moveis?" +  
27         "user=root&password=senha_acesso_banco");  
28     String sql = "SELECT * from produto";  
29     PreparedStatement comando = conexao.prepareStatement(sql);  
30     ResultSet retorno = ((java.sql.Statement) comando).executeQuery(sql);  
31  
32     while(retorno.next()){  
33         System.out.println(retorno.getString("descricao"));  
34     }  
35 }  
36  
37 }
```

Problems @ Javadoc Declaration Console X

<terminated> Conexao [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (15 de jun. de 2023 00:18:53 – 00:18:54) [pid: 91164]

```
Conexao realizada com sucesso.  
sof@ azul de 3 lugares  
Mesa de vidro com 6 cadeiras  
Guarda-roupa de casal  
Guarda-roupa de solteiro
```