Universidade do Trabalho Digital

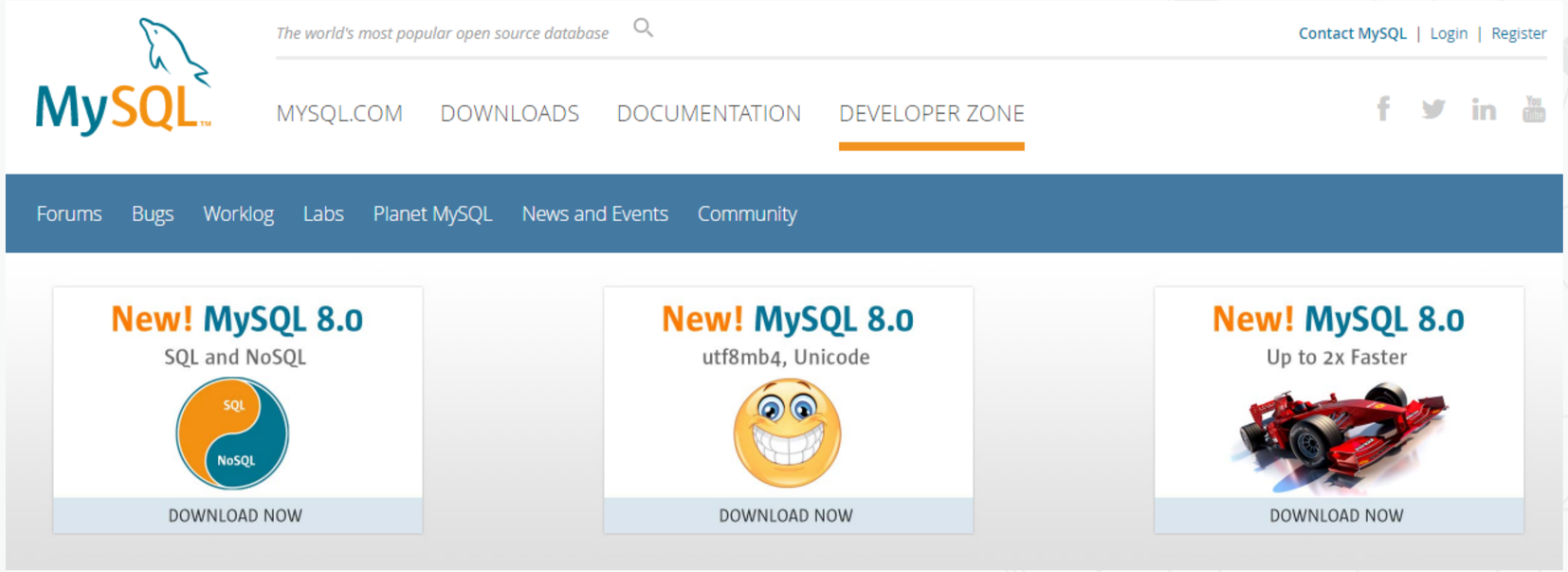
INTEGRAÇÃO JAVA COM BANCO DE DADOS MYSQL

CRUD e conexão através do JDBC

CENTEC
INSTITUTO CENTRO DE ENSINO TECNOLÓGICO

GOVERNO DO ESTADO DO CEARÁ
Secretaria da Ciência, Tecnologia e Educação Superior

Conhecendo o MySQL.



MySQL é um SGBD (sistema gerenciador de banco de dados). Vamos utilizá-lo nesse curso para fazer a persistência dos dados da nossa aplicação.

Principais comando SQL

- **Mostrar banco de dados**
  - show databases;
- **Selecionar um banco de dados.**
  - use [nome do banco];

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| bookstoredatabase  |
| cb_cadastro        |
| db_agencia         |
| db_agenda          |
| db_consulta        |
| db_ecommerce       |
| db_estudo_java     |
| db_festa           |
| db_hibernate       |
| db_loja            |
| db_projeto         |
| db_store           |
| db_supermercado    |
| db_teste           |
| disciplinadb       |
| mysql              |
| performance_schema |
| trabalho           |
+--------------------+
19 rows in set (0.01 sec)
```

Comandos SQL.

- **Criar tabela.**
  Create table [nome da tabela] ([nome da coluna][tipo de variável], [nome da coluna2][tipo de variável 2], [nome da coluna3][tipo de variável 3]);

- Ex: **create table produtos(id int, nome varchar(40), preco float, quantidade int, descricao varchar(100));**

Tabela produtos.

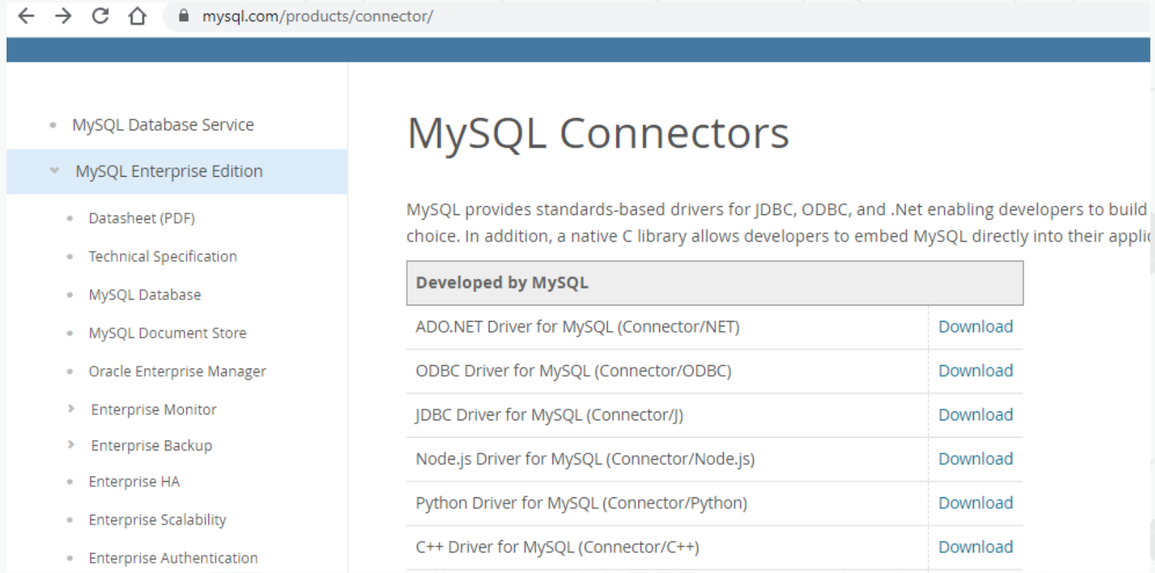
```
mysql> DESC PRODUTOS;
+-------------+---------------+------+-----+---------+----------------+
| Field       | Type          | Null | Key | Default | Extra          |
+-------------+---------------+------+-----+---------+----------------+
| id          | int(11)       | NO   | PRI | NULL    | auto_increment |
| nome        | varchar(40)   | YES  |     | NULL    |                |
| preco       | float         | YES  |     | NULL    |                |
| quantidade  | int(11)       | YES  |     | NULL    |                |
| descricao   | varchar(100)  | YES  |     | NULL    |                |
+-------------+---------------+------+-----+---------+----------------+
5 rows in set (0.81 sec)
```

- O comando desc [nome da tabela] serve para você ter essa visão detalhada da tabela que você criou.

- Com o comando show tables; visualizamos todas as tabelas no nosso banco.

Conexão com JDBC

- Baixar o conector apropriado.
- Copiar para o nosso projeto.
- Adicionar para a biblioteca
- Setar e criar a configuração.



ConnectionFactory.

```java
import java.sql.Connection;
import java.sql.DriverManager;

public class ConnectionFactory {

    private static final String USERNAME = "root";

    private static final String PASSWORD = "";

    private static final String DATABASE_URL =
    "jdbc:mysql://localhost/db_supermercado?useTimezone=true&serverTimezone=UTC";


}
```

Vamos criar a ConnectionFactory. Que vai ser a classe responsável por criar a conexão do banco de dados.Exemplo acima mostra a criação das variáveis de conexão.

ConnectionFactory.

```
    private static final String DATABASE_URL =
    "jdbc:mysql://localhost/db_supermercado?useTimezone=true&serverTimezone=UTC";

    public static Connection createConnectionToMySQL() throws Exception{
        Class.forName("com.mysql.cj.jdbc.Driver");

        Connection connection = DriverManager.getConnection(DATABASE_URL,USERNAME,PASSWORD);

        return connection;
    }

    }
}
```

Agora criamos um método que vai ser responsável por estabelecer a conexão com o banco MySQL

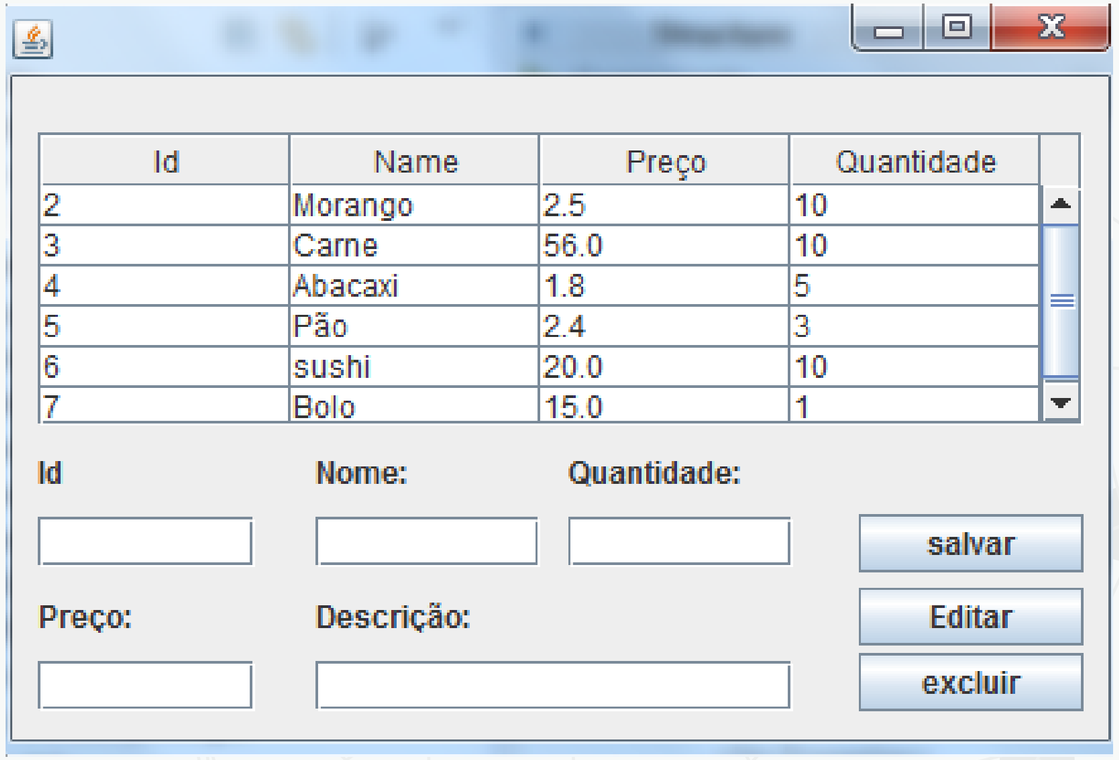
ConnectionFactory.

```
public static void main(String[]args) throws Exception {

    Connection con = createConnectionToMySQL();

    if(con!=null) {
        System.out.println("Conexão obtida com sucesso");
        con.close();
    }

}
```

Esta função serve para testar se a conexão esta funcionando.

CRUD

- Criar
- Ler
- Atualizar
- Deletar



Essas são as quatro funções básicas da persistência de dados. Vamos elaborar esse pequeno projeto para colocar em prática os que foi ensinado nesse módulo.