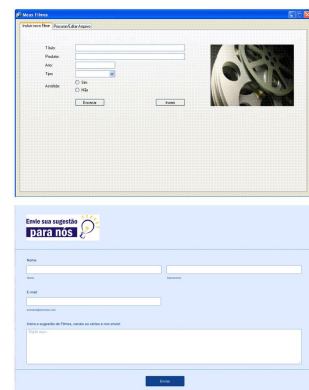
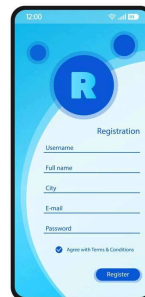


Aula 05 - INSERT, UPDATE e DELETE

Alimentando o banco de dados.

- Para alimentar (preencher com dados) as tabelas usamos formulários (front end).
- Estes podem ser construídos em diferentes linguagens e com diversas aparências.
- É através dele que o usuário irá interagir com o Banco de Dados.
- Durante o desenvolvimento podemos usar a aba "insert" no Workbench.
- O mais utilizado é usar o comando INSERT para inserir os dados na tabela para fins de testes.



Inserindo Dados de Outra Tabela

```
# criando uma tabela igual à outra
CREATE TABLE IF NOT EXISTS banco.tabela LIKE banco.tabela;

# Renomeando os campos da nova tabela - ALTER TABLE
# CHANGE COLUMN, ADD COLUMN nome AFTER, DROP COLUMN e MODIFY COLUMN
ALTER TABLE nova_tabela
    CHANGE COLUMN id novo_id INT,
    CHANGE COLUMN nome novo_nome tipo(comprimento)

# inserindo os dados de um select
INSERT INTO banco.tabela ( campo01, campo02, campo03 )
    SELECT id, nome, idade, email FROM banco.usuarios;
```

INSERT

```
# O comando INSERT no SQL é usado para adicionar novos registros (linhas) em uma tabela do banco de dados.
# Todos os campos (colunas) obrigatórios devem receber valores
# Podemos inserir várias linhas ao mesmo tempo, separando os valores por vírgula
INSERT INTO nome_da_tabela (coluna1, coluna2, coluna3) VALUES
    ( valor1, valor2, valor3 ),
    ( valor1, valor2, valor3 );

# Inserindo sem especificar as colunas ( modo arriscado)
# Passamos apenas os valores na ordem dos campos na tabela
INSERT INTO nome_da_tabela VALUES
    ( null, valor1, valor2, valor3,... );
```

INSERT

```
# Usando IGNORE impedimos que o MySQL interrompa a inserção por causa de erros
# um valor duplicado em uma coluna UNIQUE por exemplo

INSERT IGNORE INTO nome_da_tabela (coluna1, coluna2, coluna3) VALUES

    ( valor1, valor2, valor3 ),
    ( valor1, valor2, valor3 );

# Podemos também atualizar os dados se eles já existirem já existirem
INSERT INTO usuarios (id, nome, idade, email)
VALUES (1, "Carlos Silva", 26, "[email protected]")
ON DUPLICATE KEY UPDATE
    idade = VALUES(idade),
    email = VALUES(email);
```

UPDATE

```
# Muitas vezes temos que alterar os dados de um registro do banco.

# Usamos o Update SEMPRE com a cláusula WHERE e a PK.

# Se não usarmos a PK podemos alterar mais registros que o necessário.

UPDATE nome.tabela SET coluna = valor, coluna, valor
WHERE id = numeroId;
```

Delete

É o comando usado para apagar um registro da tabela.

É apagado o REGISTRO completo, ou seja, a linha toda com todos os campos.

É obrigatório usar a Chave Primária (PK) para não excluir mais que o desejado.

```
DELETE FROM nome.tabela
```

```
WHERE id = numeroId;
```