

FIAP



AULA 06

INTRODUÇÃO AO SQL/DML

Welcome to the next evolution in higher education.

I SUMÁRIO

☐ DML

- ☐ Incluir Dados(SQL Worksheet)
- ☐ Atualizar Dados (SQL Worksheet)
- ☐ Remover Dados (SQL Worksheet)
- ☐ Incluir, Atualizar e Remover Dados (Programação Visual)
- ☐ Utilizando as Extensões do ORACLE no VS Code
- ☐ Exercício Prático

| OBJETIVO

Introduzir conceitos iniciais do SQL/DML

Aplicar os conceitos no Oracle SQL Developer

- ***Data Manipulation Language (DML)*** – Linguagem de Manipulação de Dados ,
expressa um conjunto de comandos que especificam a:
 - **INSERIR** 1 ou vários dados (INSERT)
 - **ATUALIZAR** 1 ou vários dados (UPDATE)
 - **REMOVER** 1 ou vários dados (DELETE)
- Os comandos da **DML** viabiliza o acesso ou manipulação dos dados de forma compatível ao modelo de dados projetado.

- A **DML** é categorizado por:
 - **Procedural (ou Específico)**: Exige especificar quais dados são necessários e como obtê-los:
 - No INSERT é apenas mencionado as colunas que irão receber novos dados;
 - No UPDATE é apenas mencionado as colunas que irão receber novos dados E que são impostos a uma condição lógica (WHERE) para atualizar uma ou várias linhas de dados;
 - No DELETE é apenas mencionado uma condição lógica (WHERE) para remover uma ou várias linhas de dados;

- A **DML** é categorizado por:
 - **Não-Procedural (ou Genérico)**: Exige especificar quais dados são necessários, sem especificar como obtê-los (Mais fácil de aprender e poderá gerar códigos não muito eficientes).
 - No INSERT é mencionado todas as colunas que irão receber novos dados;
 - No UPDATE é mencionado as colunas que irão receber novos dados, sem a necessidade de ter uma condição lógica (WHERE), isto é, atualizando todas as linhas da tabela
 - No DELETE não é mencionado uma condição lógica (WHERE), portanto, poderá remover todas as linhas da tabela

DML – INSERT

SQL Worksheet

I COMANDO: **INSERT**

SINTAXE

- O comando **INSERT** é utilizado para incluir 1 ou vários dados de uma tabela específica em um banco de dados.
- Para usar o comando **INSERT**, é necessário especificar o nome de objeto que se deseja inserir, seguido do nome das colunas e dos novos dados respectivamente.
- Veja o exemplo da Sintaxe para Incluir

```
INSERT INTO NOME_TABELA (coluna1, coluna2)
VALUES (valor1, valor2);

-- ou

INSERT INTO NOME_TABELA
VALUES (valor1, valor2)
```

EXEMPLO DE TABELA

ALUNO

FIAP

PROMPT

```
-- EXEMPLO INSERT NÃO-PROCEDURAL (OU GENÉRICO)
INSERT INTO ALUNO VALUES (1, 'Diego Cruz')

-- EXEMPLO INSERT PROCEDURAL (OU ESPECÍFICO)
INSERT INTO ALUNO (RA) (2)
```

RESULTADO

RA	NOME
1	Diego Cruz
2	(null)

Atenção! Para esse exemplo funcionar a sua Tabela Aluno precisar ter apenas as duas colunas (ra, nome)

DML – UPDATE

SQL Worksheet

I COMANDO: UPDATE

FIAP

SINTAXE

- O comando **UPDATE** é utilizado para atualizar 1 ou vários dados de uma tabela específica em um banco de dados.
- Para usar o comando **UPDATE**, é necessário especificar o nome de objeto que se deseja atualizar (nome da tabela), seguido do nome das colunas que irão receber o novo valor E da condição lógica para atualizar os dados (**WHERE** – podendo ser opcional)
- Veja o exemplo da Sintaxe para Atualizar

```
UPDATE NOME_TABELA  
SET coluna1=valor_novo;  
  
-- ou  
  
UPDATE NOME_TABELA  
SET coluna1=valor_novo;  
WHERE <condicao_logica>;
```

EXEMPLO DE TABELA ALUNO

FIAP

COMANDO

RESULTADO

NÃO-PROCEDURAL

RA	NOME
1	João
2	João

Todas as linhas
foram alteradas

PROCEDURAL

RA	NOME
1	Maria
2	João

Apenas uma linha
foi alterada

```
-- EXEMPLO UPDATE NÃO-PROCEDURAL (OU GENÉRICO)
UPDATE ALUNO
SET nome='João';

-- EXEMPLO UPDATE PROCEDURAL (OU ESPECÍFICO)
UPDATE ALUNO
SET nome='Maria';
WHERE ra=1;
```

DML – DELETE

SQL Worksheet

I COMANDO: **DELETE**

SINTAXE

- O comando **DELETE** é utilizado para remover 1 ou vários dados de uma tabela específica em um banco de dados.
- Para usar o comando **DELETE**, é necessário especificar o nome de objeto que se deseja inserir (nome da tabela), seguido da condição lógica para remover os dados (**WHERE** – podendo ser opcional)
- Veja o exemplo da Sintaxe para Remover



```
DELETE FROM NOME_TABELA;
```

```
-- ou
```

```
DELETE FROM NOME_TABELA  
WHERE <condicao_logica>;
```

I EXEMPLO DE TABELA

ALUNO

PROMPT



```
-- EXEMPLO DELETE NÃO-PROCEDURAL (OU GENÉRICO)
```

```
DELETE FROM ALUNO
```

```
-- EXEMPLO DELETE PROCEDURAL (OU ESPECÍFICO)
```

```
DELETE FROM ALUNO
```

```
WHERE ra=1;
```

FIAP

RESULTADO

NÃO-PROCEDURAL

RA	NOME
----	------

Nenhuma linha
permaneceu

PROCEDURAL

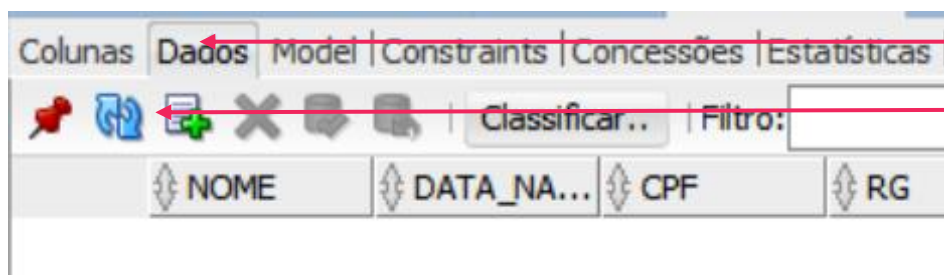
RA	NOME
2	João

Apenas 1 linha
permaneceu

DML – INSERT/DELETE/UPDATE

Programação Visual

1. Clique-esquerdo na Tabela e Procure a Aba “Dados”



Aba “Dados”

Atualize os “Dados”

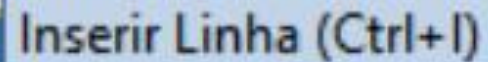
A screenshot of the same database management tool interface, but now showing data in the 'Dados' tab. The tab bar includes additional tabs: 'Triggers', 'Flashback', 'Dependências', 'Detalhes', and 'Partições'. The toolbar and column list remain the same. The main area displays a table with two rows of data.

	RA	NOME	DATA_NASCIMENTO	CPF	RG	CEP_FK	SEXO	ST_ALUNO	IDADE
1	2	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)
2	1	Diego Cruz	(null)	(null)	(null)	(null)	(null)	(null)	(null)

Exibição dos “Dados”

FIAP

- Ou clique no ícone



Confirma a Inclusão

19

ALTERAR DADOS NA TABELA

1. Clique duas vezes na tupla que deseja alterar

	RA	NOME
1	1	Diego Cruz
2	2	(null)

Clique no ícone
para abrir a
Janela de
"Editar Valor"

Editar Valor

Finalizador de Linhas: Plataforma Padrão [Change...](#)

Valor:

Diego Silva

[Ajuda](#) [OK](#) [Cancelar](#)

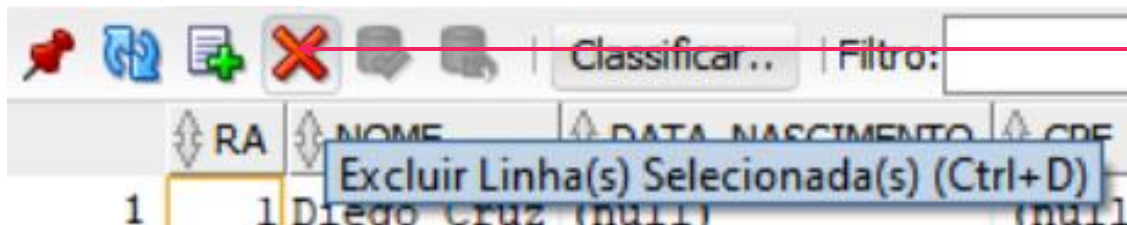
	RA	NOME	DATA_NASCIMENTO	CPF	RG	CEP_FK	SEXO	ST_ALUNO	IDADE
*1	1	Diego Silva	(null)	(null)	(null)	(null)	(null)	(null)	(null)
2	2	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)

Confirma a Atualização

A linha que foi modificada







REMOVER DADOS NA TABELA

1. CTRL+D para Remover uma Linha



Ou clique no ícone

Colunas | Dados | Model | Constraints | Concessões | Estatísticas | Triggers | Flashback | Dependências | Detalhes | Partições |



Classificar.. | Filtro:

	RA	NOME	DATA_NASCIMENTO	CPF	RG	CEP_FK	SEXO	ST_ALUNO	IDADE
-1	1	Diego Cruz	(null)	(null)	(null)	(null)	(null)	(null)	(null)
2	2	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)

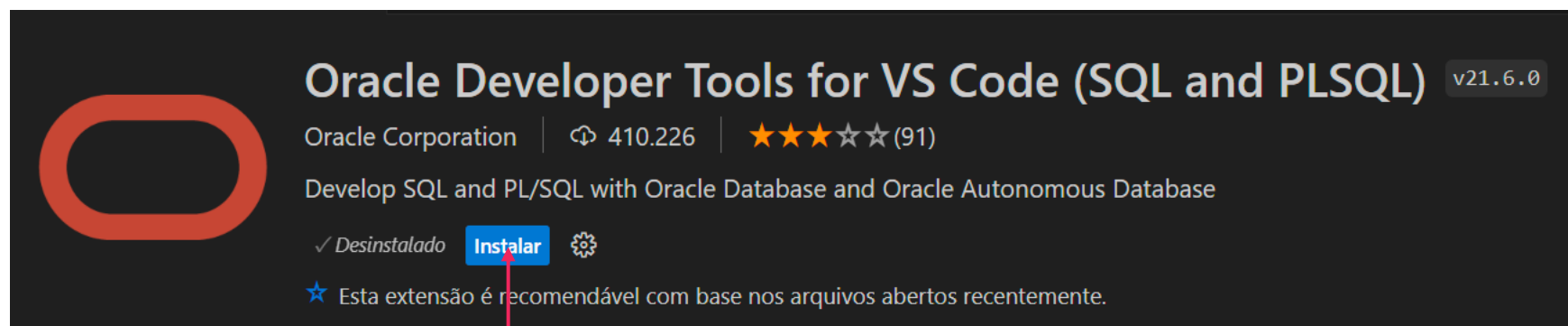
Confirma a Exclusão

Selecione a Linha para Exclusão

DML – INSERT/DELETE/UPDATE

Programação Visual

1. Clique em Extensões ou CTRL+SHIFT+X e Busque por “Oracle”
e Localize essa extensão:



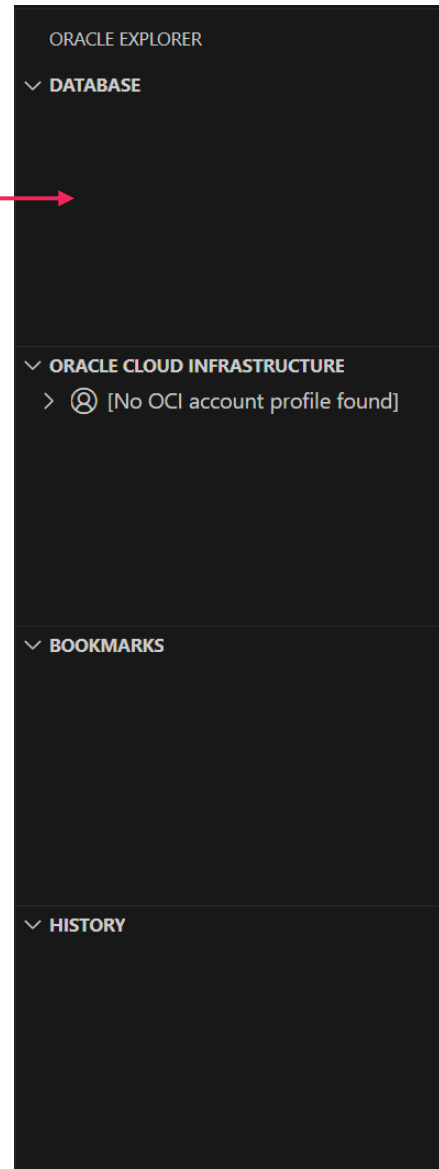
Clique para
Instalar

I CONFIGURANDO A EXTENSÃO

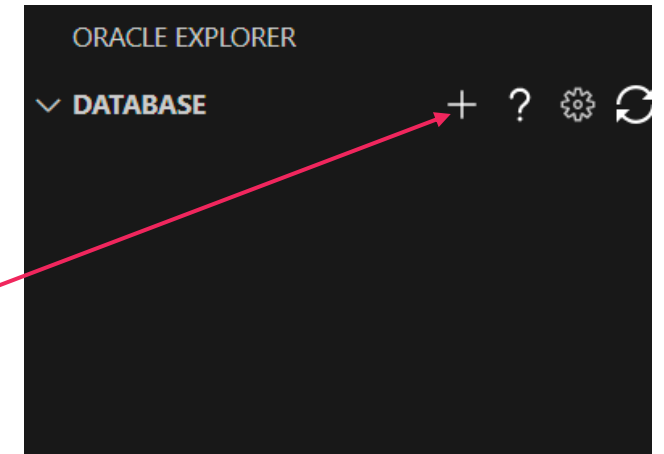
FIAP



2. Clique na Area
DATABASE



3. Clique na Realizar
uma Conexão



Create New Connection ⓘ

User

Connection Type Basic (Host, Port, Service Name) ▼

Database host name * localhost

Port number * 1521

Service name * XEPDB1

Connection string localhost:1521/XEPDB1 Copy

Role Default ▼

User name * ADMIN

Password *

☐ Save password

☐ Set as default connection

Connection name * ADMIN.XEPDB1 ⊗ Color

☐ Show more options

Create Connection

Configure as
propriedades da
conexão

PRIMEIRA CONEXÃO

Informações para conexão na FIAP:

Conexão: FIAP_SEUNOME (sugestão)

Usuário: RM##### (número do seu RM)

Senha: A senha é sua data de nascimento (DDMMAA) – Exemplo 220399

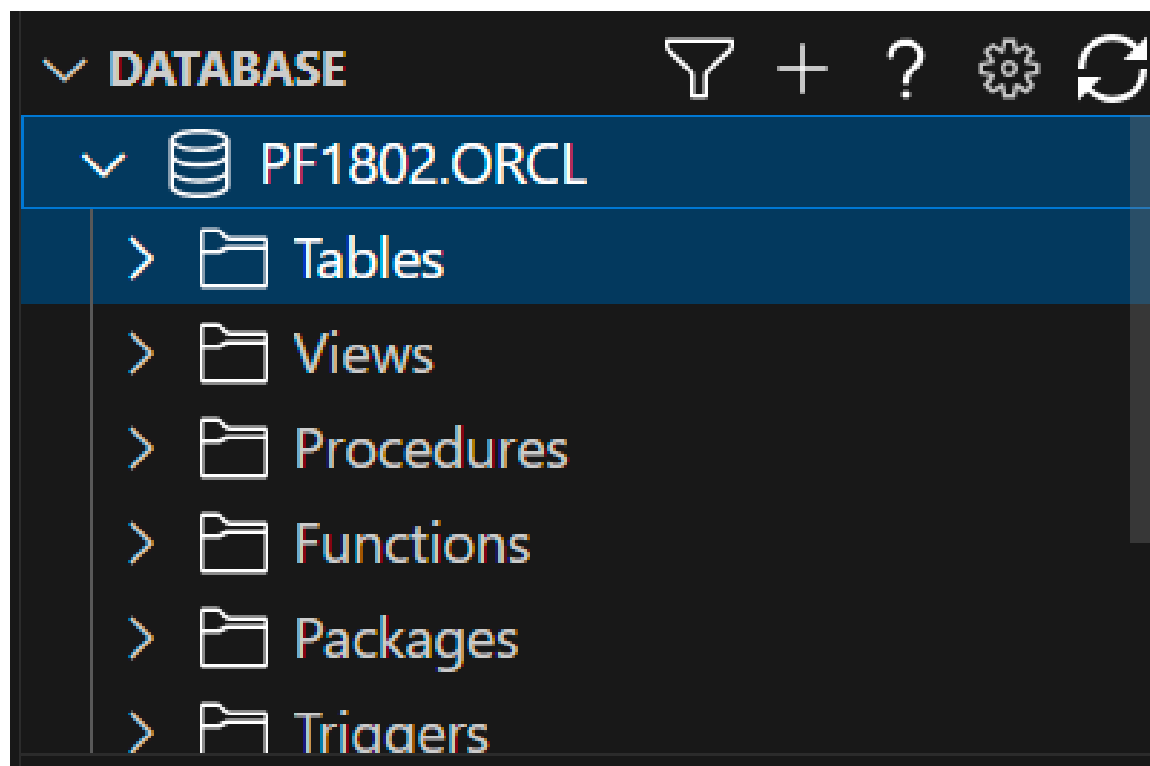
Para acessar o Oracle na FIAP, utilizamos a seguinte configuração:

HOST: **oracle.fiap.com.br**

PORTA: **1521**

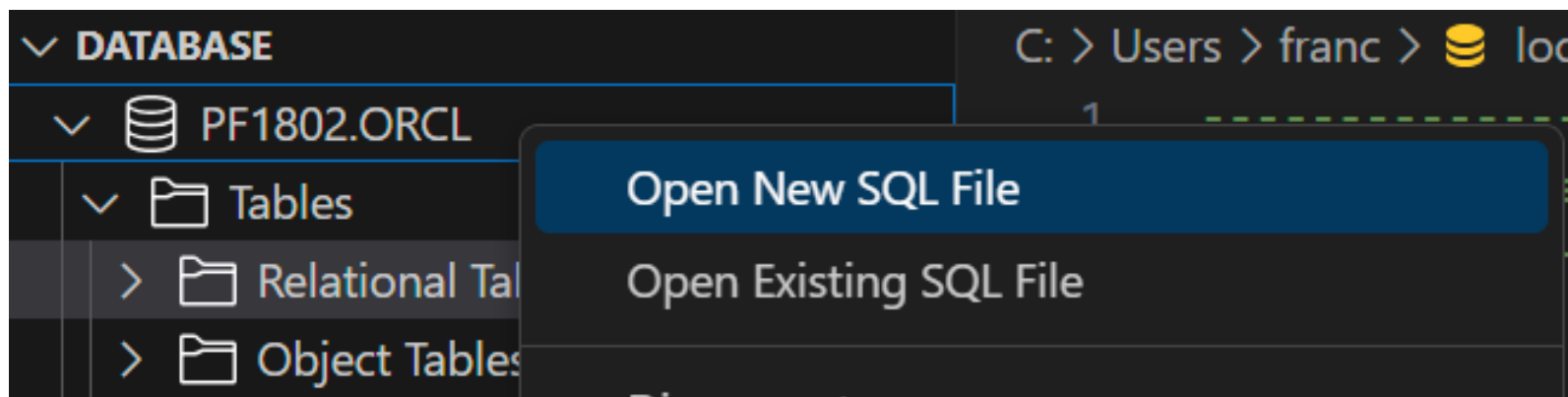
SID: **ORCL**

1. Após sua conexão for estabelecida, a extensão demonstrará os objetos disponíveis no seu usuário



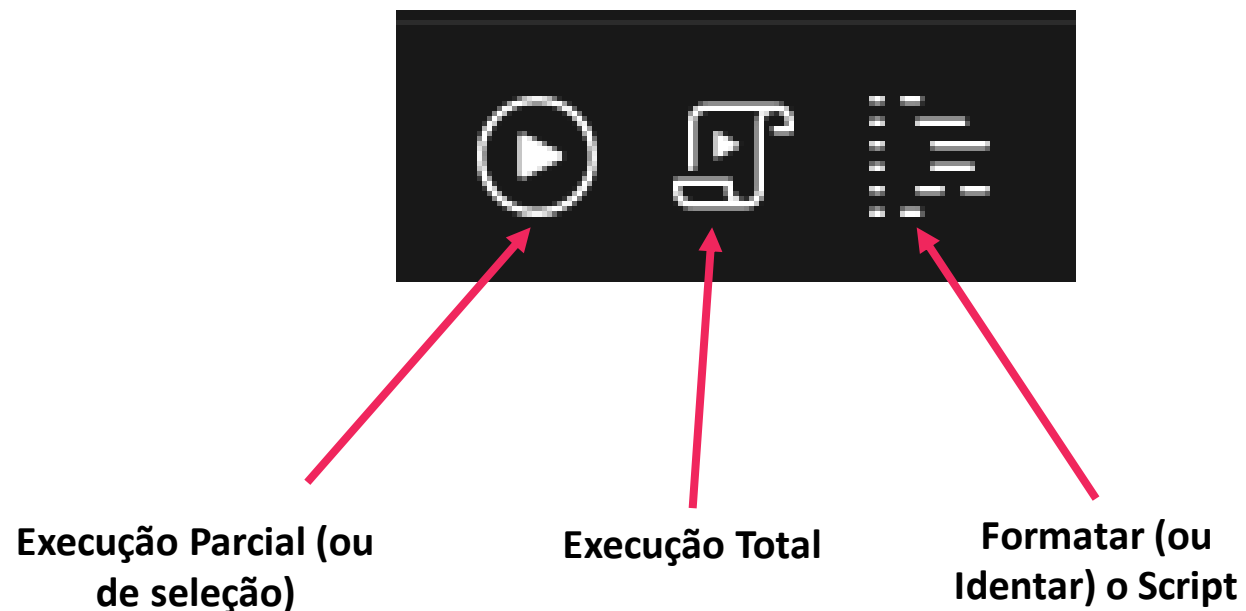
ABRIR UM ARQUIVO SQL

1. Para abrir um arquivo (novo ou existente) É necessário clicar no botão direito do mouse sobre o nome da conexão e escolha opção que desejar

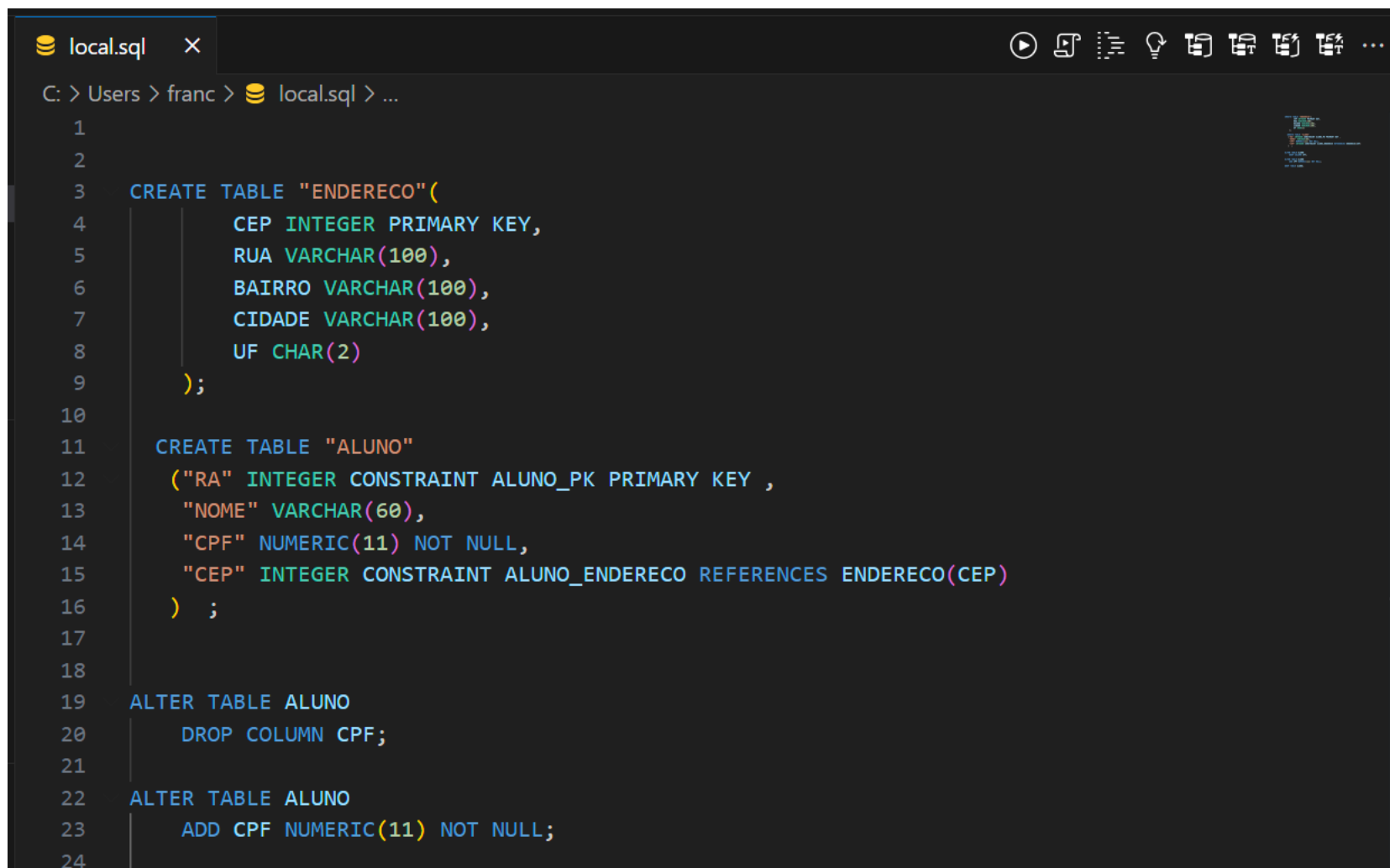


EXECUTAR UM ARQUIVO SQL

1. Ao abrir um arquivo SQL, no canto superior direito será habilitado os botões de execução.



EXEMPLO UM ARQUIVO SQL

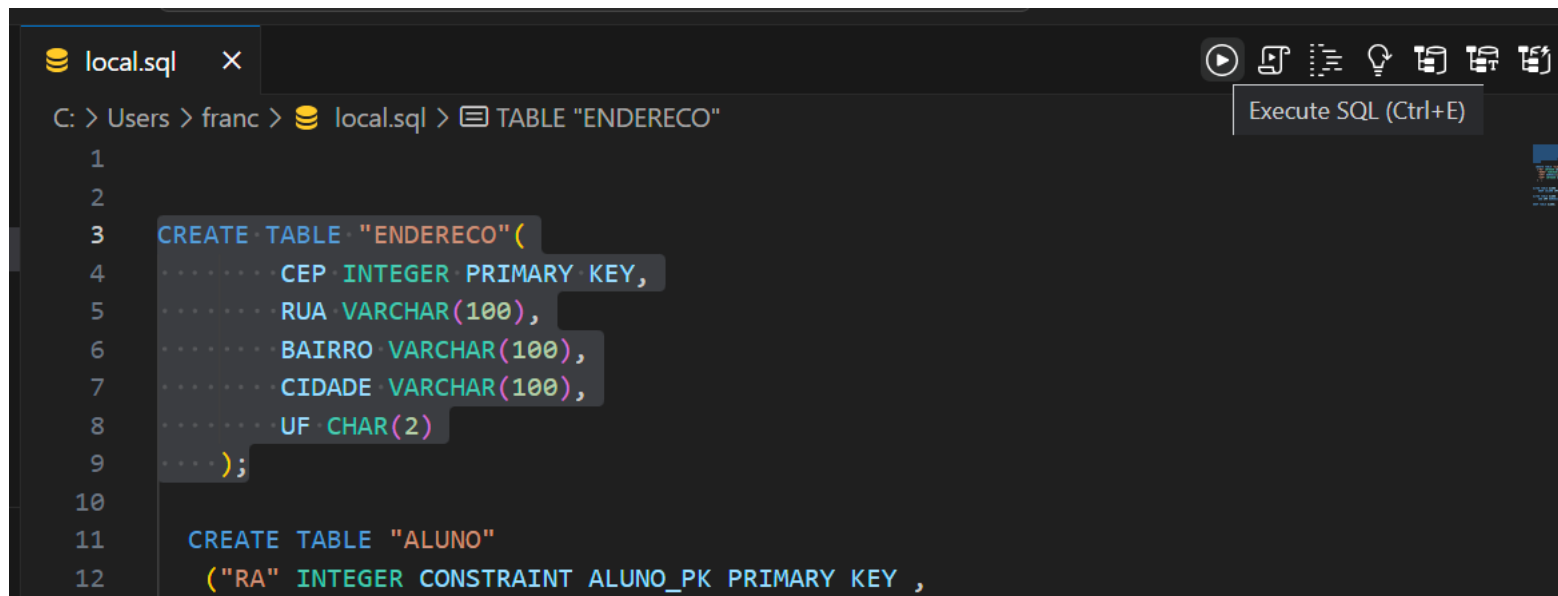


```
local.sql ×
C: > Users > franc > local.sql > ...

1
2
3 CREATE TABLE "ENDereco"(
4     CEP INTEGER PRIMARY KEY,
5     RUA VARCHAR(100),
6     BAIRRO VARCHAR(100),
7     CIDADE VARCHAR(100),
8     UF CHAR(2)
9 );
10
11 CREATE TABLE "ALUNO"
12     ("RA" INTEGER CONSTRAINT ALUNO_PK PRIMARY KEY ,
13     "NOME" VARCHAR(60),
14     "CPF" NUMERIC(11) NOT NULL,
15     "CEP" INTEGER CONSTRAINT ALUNO_ENDereco REFERENCES Endereco(CEP)
16 );
17
18
19 ALTER TABLE ALUNO
20     DROP COLUMN CPF;
21
22 ALTER TABLE ALUNO
23     ADD CPF NUMERIC(11) NOT NULL;
24
```

EXEMPLO UMA EXECUÇÃO PARCIAL

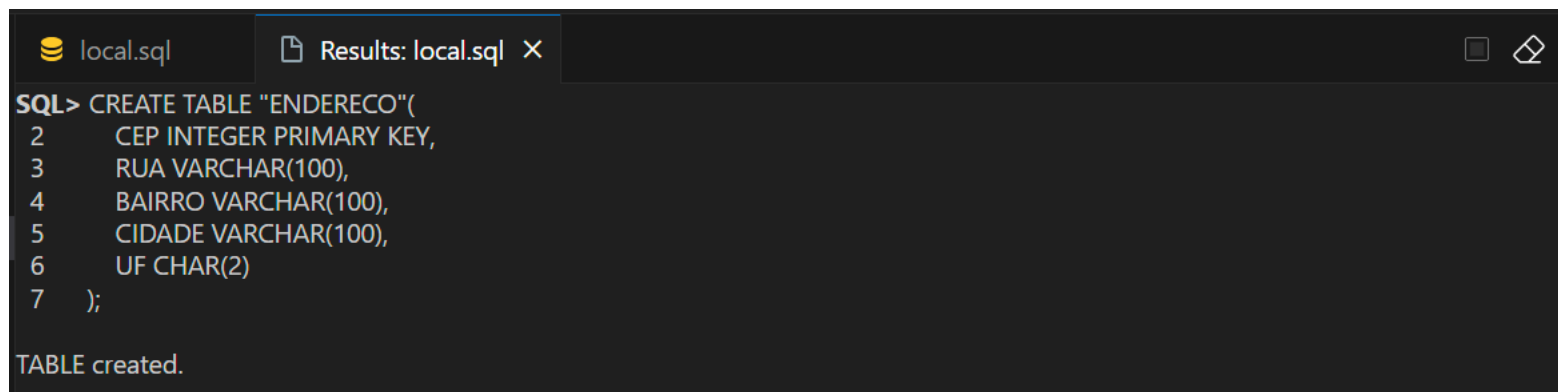
SELECIONE E APERTE CTRL+E



A screenshot of a SQL editor window titled 'local.sql'. The editor shows a SQL script with line numbers 1 through 12. The script defines two tables: 'ENDERECO' and 'ALUNO'. The 'ENDERECO' table has columns: CEP (INTEGER PRIMARY KEY), RUA (VARCHAR(100)), BAIRRO (VARCHAR(100)), CIDADE (VARCHAR(100)), and UF (CHAR(2)). The 'ALUNO' table has a column 'RA' (INTEGER) with a constraint 'ALUNO_PK' and is marked as a PRIMARY KEY. A tooltip 'Execute SQL (Ctrl+E)' is visible over the toolbar.

```
1  
2  
3 CREATE TABLE "ENDERECO"(  
4     CEP INTEGER PRIMARY KEY,  
5     RUA VARCHAR(100),  
6     BAIRRO VARCHAR(100),  
7     CIDADE VARCHAR(100),  
8     UF CHAR(2)  
9 );  
10  
11 CREATE TABLE "ALUNO"  
12 ("RA" INTEGER CONSTRAINT ALUNO_PK PRIMARY KEY ,
```

RESULTADO DA EXECUÇÃO DO SCRIPT

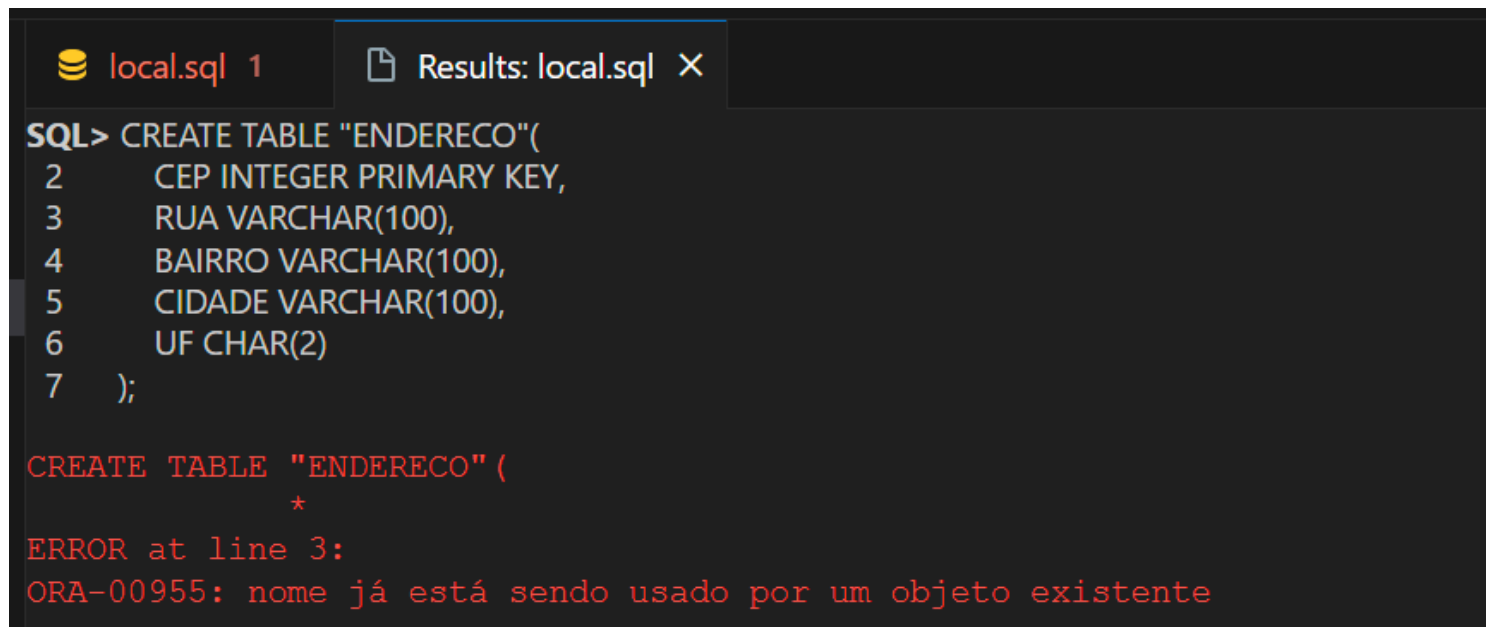


A screenshot of a SQL editor window showing the result of the script execution. The window has two tabs: 'local.sql' and 'Results: local.sql'. The 'Results' tab is active, displaying the output of the SQL script. The output shows the 'CREATE TABLE' statements for 'ENDERECO' and 'ALUNO', followed by the message 'TABLE created.'.

```
SQL> CREATE TABLE "ENDERECO"(  
2     CEP INTEGER PRIMARY KEY,  
3     RUA VARCHAR(100),  
4     BAIRRO VARCHAR(100),  
5     CIDADE VARCHAR(100),  
6     UF CHAR(2)  
7 );  
  
TABLE created.
```

EXEMPLO DE UM ERRO

RESULTADO DA EXECUÇÃO DO SCRIPT



The screenshot shows a SQL IDE window with two tabs: 'local.sql 1' and 'Results: local.sql X'. The main editor area contains the following SQL code:

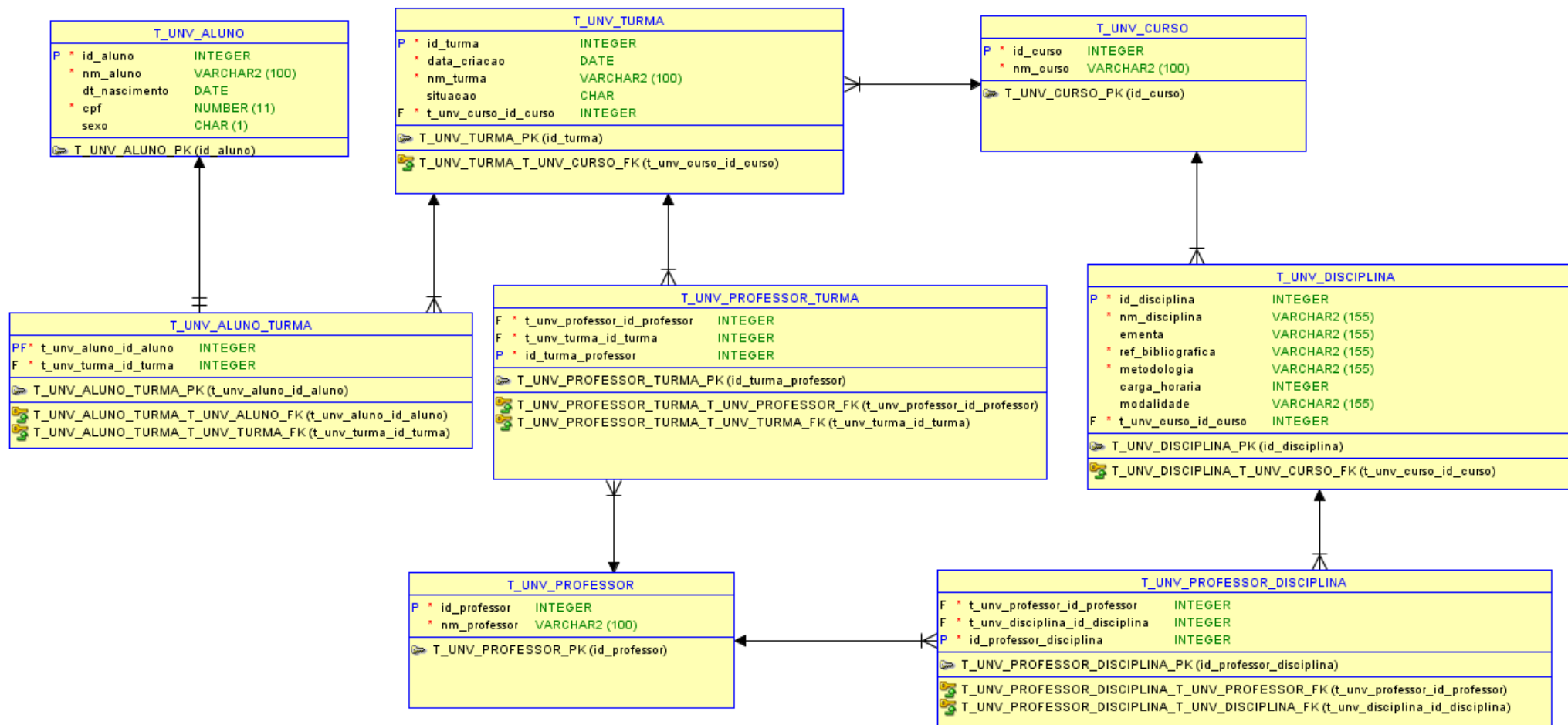
```
SQL> CREATE TABLE "ENDERECO"(  
2     CEP INTEGER PRIMARY KEY,  
3     RUA VARCHAR(100),  
4     BAIRRO VARCHAR(100),  
5     CIDADE VARCHAR(100),  
6     UF CHAR(2)  
7 );
```

Below the code, the execution results are displayed in red text:

```
CREATE TABLE "ENDERECO" (  
      *  
ERROR at line 3:  
ORA-00955: nome já está sendo usado por um objeto existente
```


EXERCÍCIO PRÁTICO

EXERCÍCIO PRÁTICO



- Altere as estruturas das tabelas no SQL Developer:
 - Adicione o campo SITUACAO (CHAR(1)) na tabela ALUNO e PROFESSOR;
 - Altere o nome do campo CARGA HORARIA para CH da tabela DISCIPLINA ;
 - Altere o nome da tabela ALUNO para DISCENTE e PROFESSOR para DOCENTE;
 - Altere o tipo das colunas EMENTA, REF BIBLIOGRAFICA E METODOLOGIA para LONG VARCHAR;
 - Remova todas as tabelas existentes (Observe que há uma ordem correta de exclusão);

Copyright © 2023 Profº Drº Francisco Douglas Lima Abreu

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito ao autor

FIAP

THE WAY WE ARE