

# Template de Mapa Mental

Aluno: João Victor Ferrareis Ribeiro

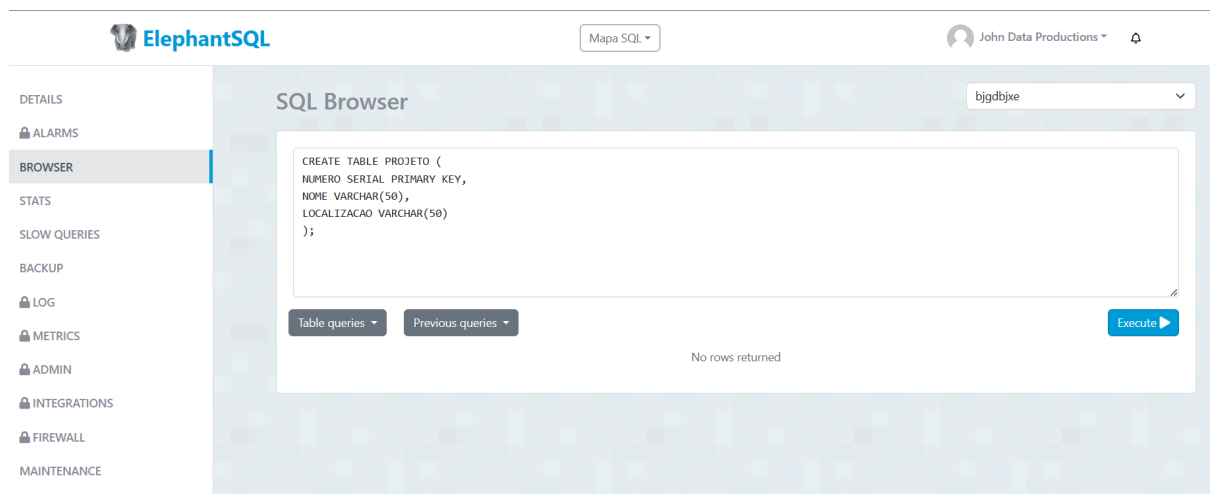
RAMO: DDL

INSTRUÇÃO: CREATE TABLE

CÓDIGO:

```
CREATE TABLE PROJETO (  
    NUMERO SERIAL PRIMARY KEY,  
    NOME VARCHAR(50),  
    LOCALIZACAO VARCHAR(50)  
);
```

IMAGEM:



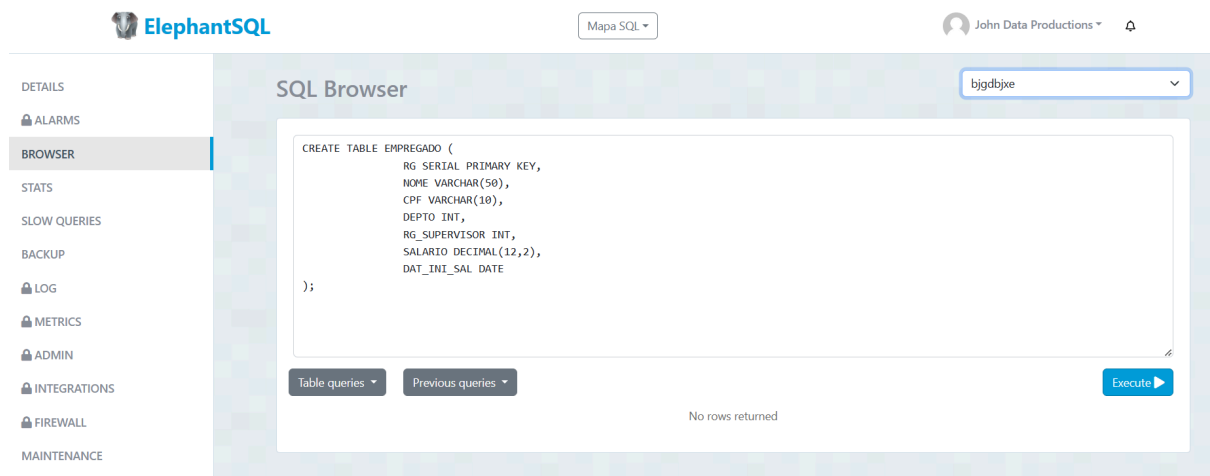
## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

### CÓDIGO:

```
CREATE TABLE EMPREGADO (  
    RG SERIAL PRIMARY KEY,  
    NOME VARCHAR(50),  
    CPF VARCHAR(10),  
    DEPTO INT,  
    RG_SUPERVISOR INT,  
    SALARIO DECIMAL(12,2),  
    DAT_INI_SAL DATE  
);
```

### IMAGEM:



## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

### CÓDIGO:

```
CREATE TABLE DEPARTAMENTO (  
    NUMERO SERIAL PRIMARY KEY,  
    NOME VARCHAR(50),  
    RG_GERENTE INT,  
    FOREIGN KEY (RG_GERENTE) REFERENCES EMPREGADO(RG)  
);
```

### IMAGEM:



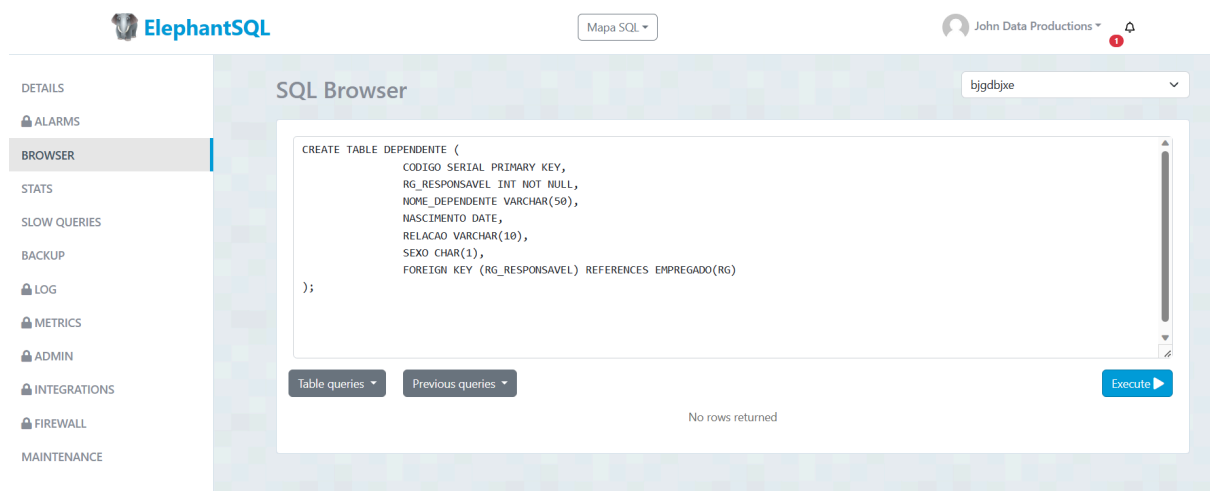
## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

## CÓDIGO:

```
CREATE TABLE DEPENDENTE (  
    CODIGO SERIAL PRIMARY KEY,  
    RG_RESPONSAVEL INT NOT NULL,  
    NOME_DEPENDENTE VARCHAR(50),  
    NASCIMENTO DATE,  
    RELACAO VARCHAR(10),  
    SEXO CHAR(1),  
    FOREIGN KEY (RG_RESPONSAVEL) REFERENCES EMPREGADO(RG)  
);
```

## IMAGEM:



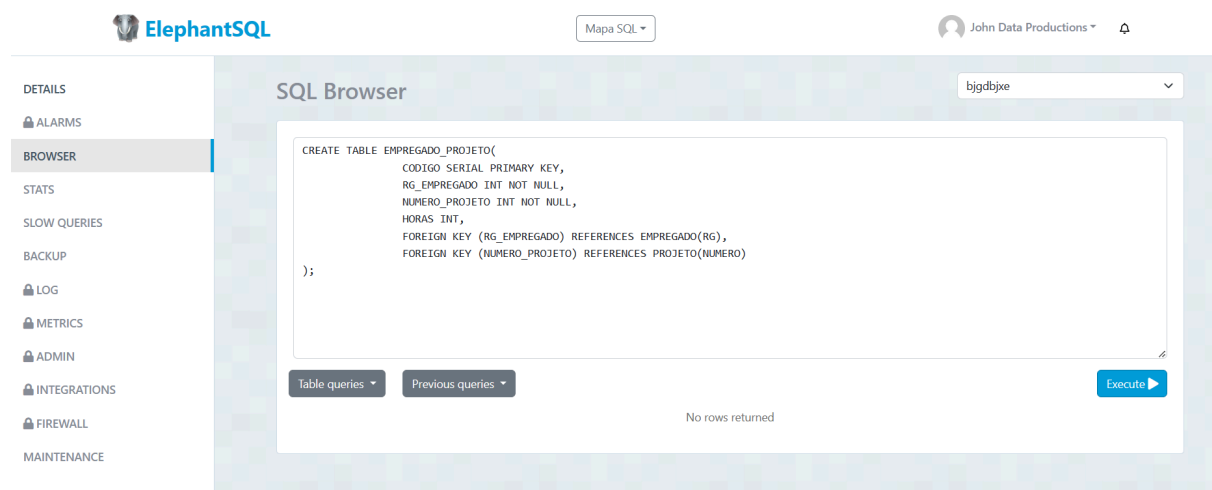
## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

## CÓDIGO:

```
CREATE TABLE EMPREGADO_PROJETO(  
    CODIGO SERIAL PRIMARY KEY,  
    RG_EMPREGADO INT NOT NULL,  
    NUMERO_PROJETO INT NOT NULL,  
    HORAS INT,  
    FOREIGN KEY (RG_EMPREGADO) REFERENCES EMPREGADO(RG),  
    FOREIGN KEY (NUMERO_PROJETO) REFERENCES PROJETO(NUMERO)  
);
```

## IMAGEM:



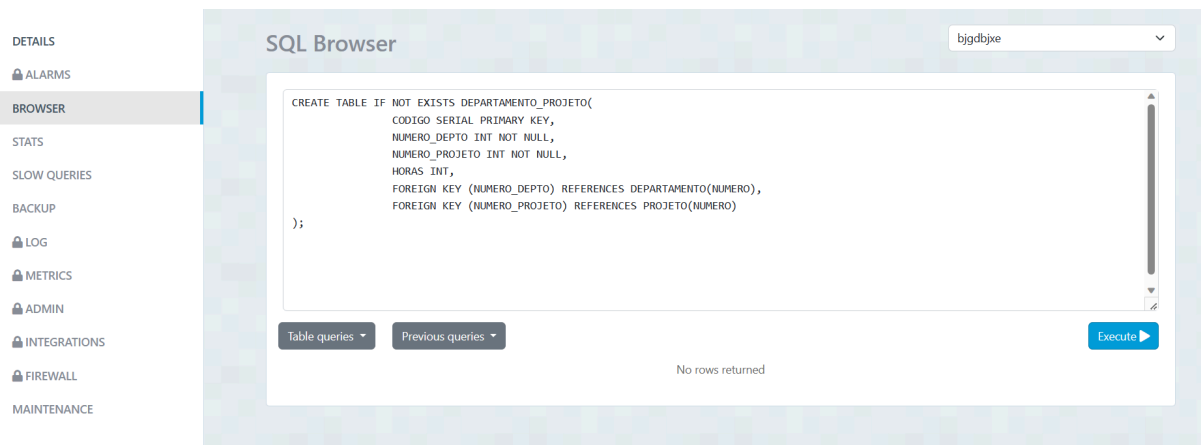
## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

### CÓDIGO:

```
CREATE TABLE IF NOT EXISTS DEPARTAMENTO_PROJETO(  
    CODIGO SERIAL PRIMARY KEY,  
    NUMERO_DEPTO INT NOT NULL,  
    NUMERO_PROJETO INT NOT NULL,  
    HORAS INT,  
    FOREIGN KEY (NUMERO_DEPTO)  
    REFERENCES DEPARTAMENTO(NUMERO),  
    FOREIGN KEY (NUMERO_PROJETO) REFERENCES PROJETO(NUMERO)  
);
```

### IMAGEM:



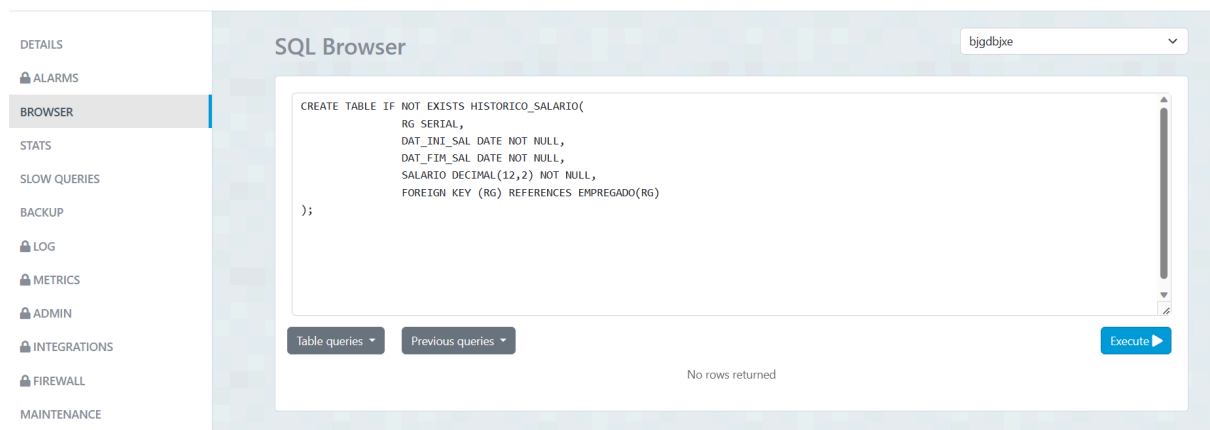
## RAMO: DDL

## INSTRUÇÃO: CREATE TABLE

### CÓDIGO:

```
CREATE TABLE IF NOT EXISTS HISTORICO_SALARIO(  
    RG SERIAL,  
    DAT_INI_SAL DATE NOT NULL,  
    DAT_FIM_SAL DATE NOT NULL,  
    SALARIO DECIMAL(12,2) NOT NULL,  
    FOREIGN KEY (RG) REFERENCES EMPREGADO(RG)  
);
```

### IMAGEM:



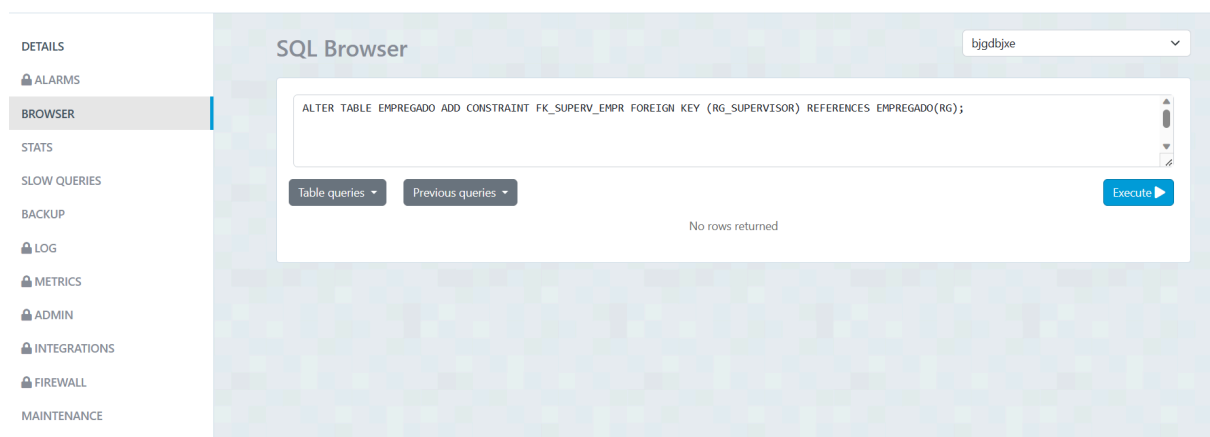
## RAMO: DDL

## INSTRUÇÃO: ALTER TABLE

### CÓDIGO:

```
ALTER TABLE EMPREGADO  
ADD CONSTRAINT FK_SUPERV_EMPR FOREIGN KEY (RG_SUPERVISOR)  
REFERENCES EMPREGADO(RG);
```

### IMAGEM:





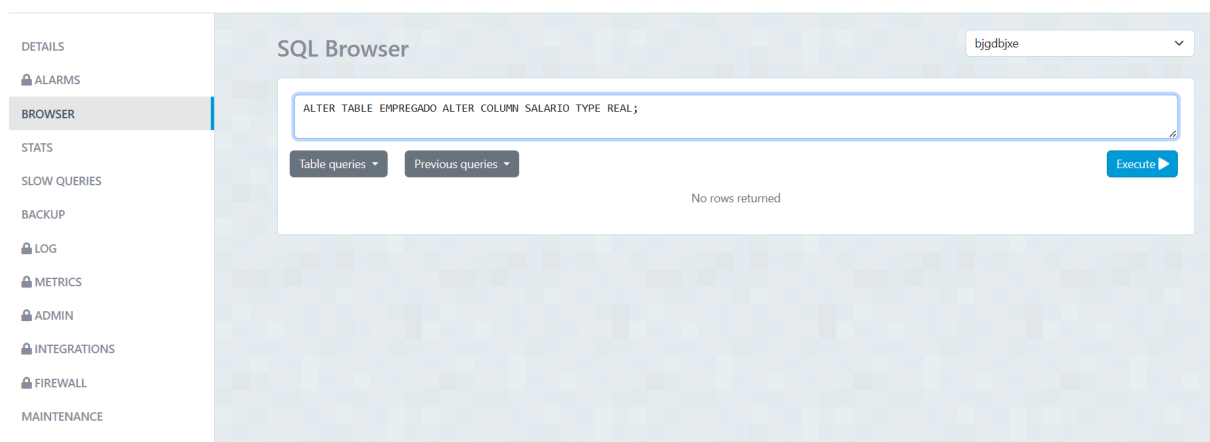
RAMO: DDL

INSTRUÇÃO: ALTER TABLE

CÓDIGO:

```
ALTER TABLE EMPREGADO ALTER COLUMN SALARIO TYPE REAL;
```

IMAGEM:



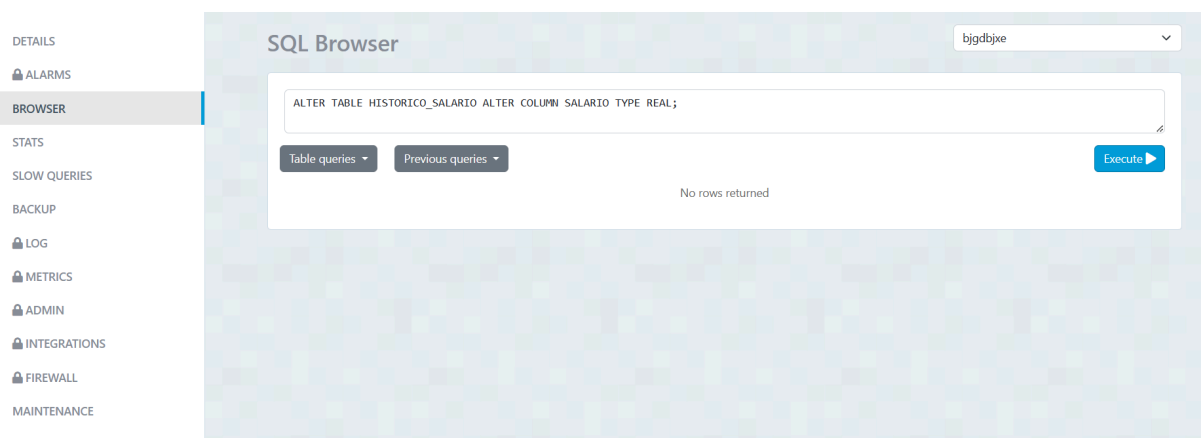
RAMO: DDL

INSTRUÇÃO: ALTER TABLE

CÓDIGO:

```
ALTER TABLE HISTORICO_SALARIO ALTER COLUMN SALARIO TYPE REAL;
```

IMAGEM:



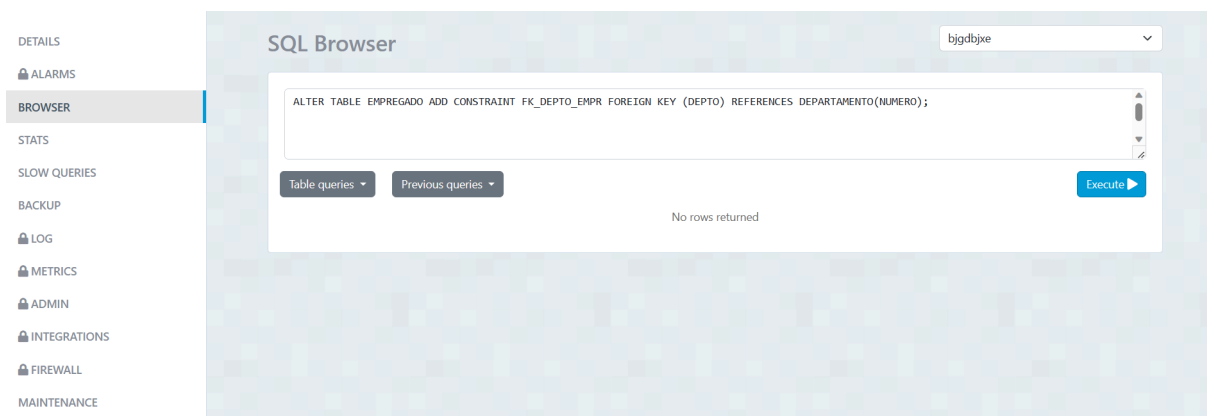
## RAMO: DDL

## INSTRUÇÃO: ALTER TABLE

### CÓDIGO:

```
ALTER TABLE EMPREGADO ADD CONSTRAINT FK_DEPTO_EMPR FOREIGN KEY (DEPTO) REFERENCES DEPARTAMENTO(NUMERO);
```

### IMAGEM:



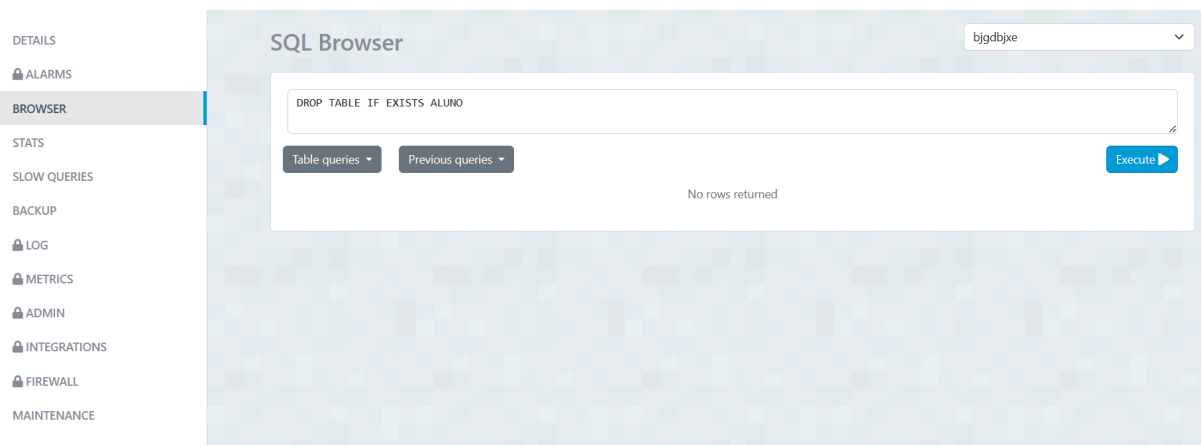
RAMO: DDL

INSTRUÇÃO: DROP TABLE

CÓDIGO:

```
DROP TABLE IF EXISTS ALUNO;
```

IMAGEM:



## RAMO: DML

## INSTRUÇÃO: INSERT

## CÓDIGO:

```
INSERT INTO PROJETO (NOME, NUMERO, LOCALIZACAO)
VALUES ('Financeiro', 5, 'São Paulo'),
       ('Motor', 10, 'Rio Claro'),
       ('Prédio Central', 20, 'Campinas'),
       ('Águas Limpas', 25, 'Vitória');
```

## IMAGEM:



## RAMO: DML

---

## INSTRUÇÃO: INSERT

---

### CÓDIGO:

```
INSERT INTO DEPARTAMENTO (NOME, NUMERO, RG_GERENTE)
VALUES ('Contabilidade', 1, NULL),
       ('Engenharia Civil', 2, NULL),
       ('Engenharia Mecânica', 3, NULL),
       ('Industrial', 4, NULL);
```

### IMAGEM:



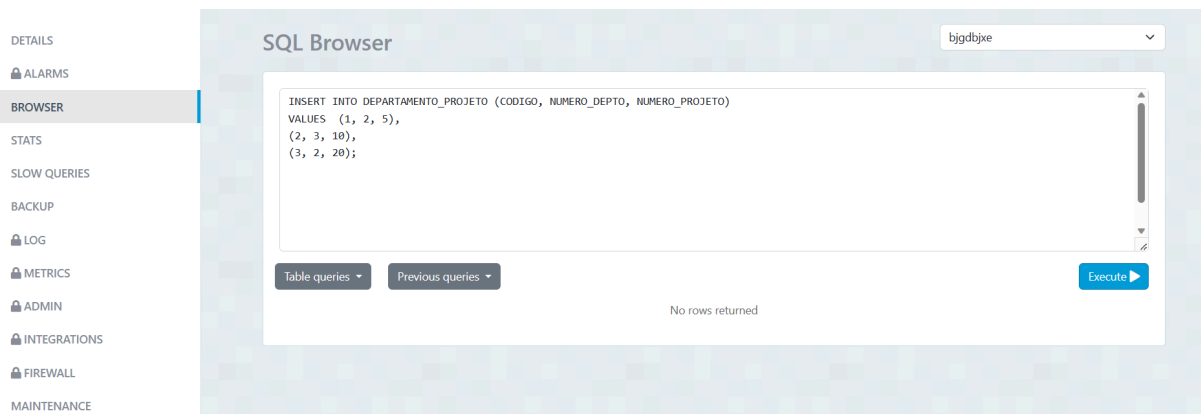
## RAMO: DML

## INSTRUÇÃO: INSERT

### CÓDIGO:

```
INSERT INTO DEPARTAMENTO_PROJETO (CODIGO, NUMERO_DEPTO,  
NUMERO_PROJETO)  
VALUES      (1, 2, 5),  
             (2, 3, 10),  
             (3, 2, 20);
```

### IMAGEM:



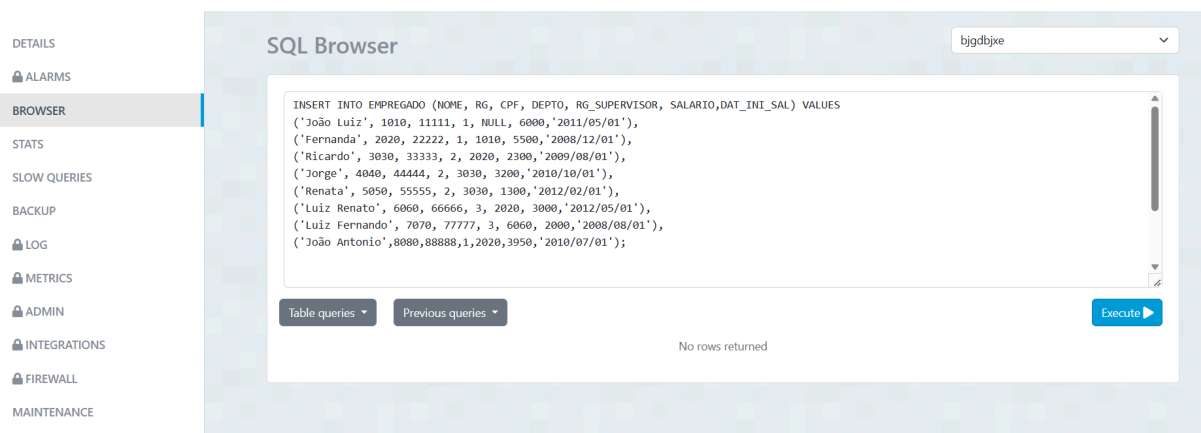
## RAMO: DML

## INSTRUÇÃO: INSERT

## CÓDIGO:

```
INSERT INTO EMPREGADO (NOME, RG, CPF, DEPTO, RG_SUPERVISOR,
SALARIO,DAT_INI_SAL) VALUES
('João Luiz', 1010, 11111, 1, NULL, 6000,'2011/05/01'),
('Fernanda', 2020, 22222, 1, 1010, 5500,'2008/12/01'),
('Ricardo', 3030, 33333, 2, 2020, 2300,'2009/08/01'),
('Jorge', 4040, 44444, 2, 3030, 3200,'2010/10/01'),
('Renata', 5050, 55555, 2, 3030, 1300,'2012/02/01'),
('Luiz Renato', 6060, 66666, 3, 2020, 3000,'2012/05/01'),
('Luiz Fernando', 7070, 77777, 3, 6060, 2000,'2008/08/01'),
('João Antonio',8080,88888,1,2020,3950,'2010/07/01');
```

## IMAGEM:





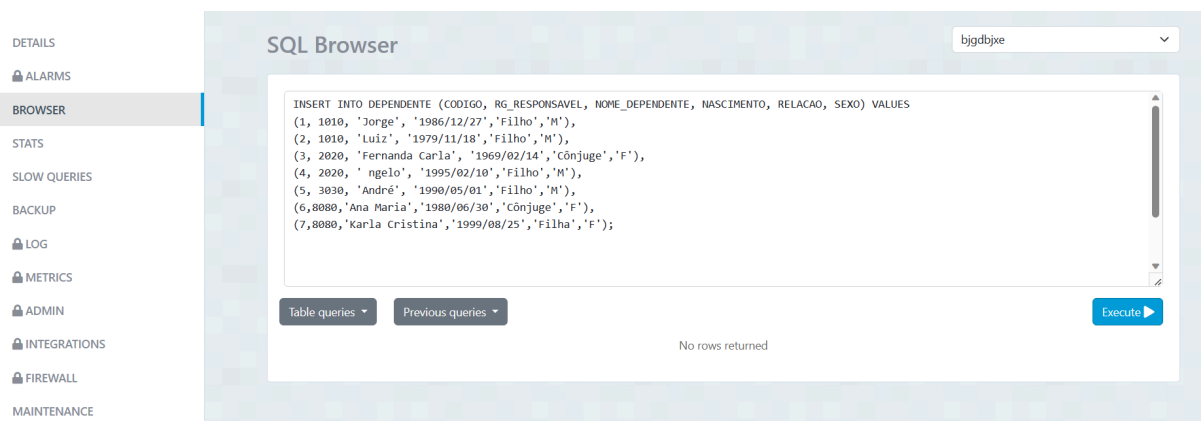
## RAMO: DML

## INSTRUÇÃO: INSERT

## CÓDIGO:

```
INSERT INTO DEPENDENTE (CODIGO, RG_RESPONSAVEL,
NOME_DEPENDENTE, NASCIMENTO, RELACAO, SEXO) VALUES
(1, 1010, 'Jorge', '1986/12/27', 'Filho', 'M'),
(2, 1010, 'Luiz', '1979/11/18', 'Filho', 'M'),
(3, 2020, 'Fernanda Carla', '1969/02/14', 'Cônjuge', 'F'),
(4, 2020, 'Ângelo', '1995/02/10', 'Filho', 'M'),
(5, 3030, 'André', '1990/05/01', 'Filho', 'M'),
(6, 8080, 'Ana Maria', '1980/06/30', 'Cônjuge', 'F'),
(7, 8080, 'Karla Cristina', '1999/08/25', 'Filha', 'F');
```

## IMAGEM:



## RAMO: DML

---

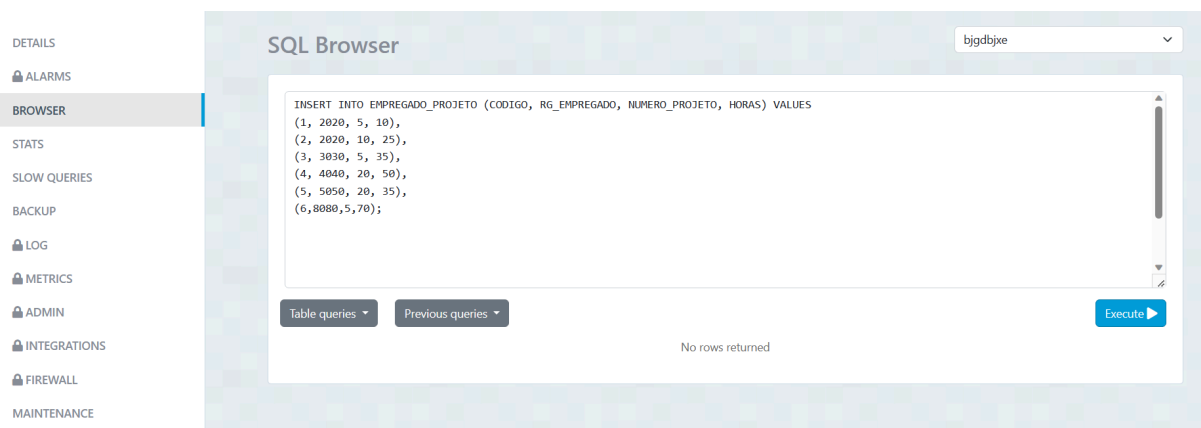
## INSTRUÇÃO: INSERT

---

### CÓDIGO:

```
INSERT INTO EMPREGADO_PROJETO (CODIGO, RG_EMPREGADO,  
NUMERO_PROJETO, HORAS) VALUES  
  (1, 2020, 5, 10),  
  (2, 2020, 10, 25),  
  (3, 3030, 5, 35),  
  (4, 4040, 20, 50),  
  (5, 5050, 20, 35),  
  (6,8080,5,70);
```

### IMAGEM:



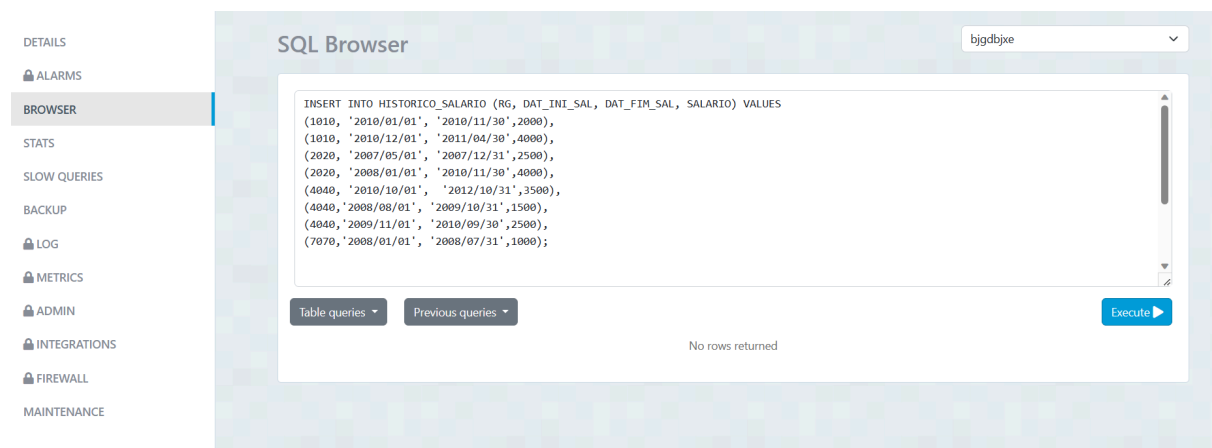
## RAMO: DML

## INSTRUÇÃO: INSERT

## CÓDIGO:

```
INSERT INTO HISTORICO_SALARIO (RG, DAT_INI_SAL, DAT_FIM_SAL,
SALARIO) VALUES
(1010, '2010/01/01', '2010/11/30',2000),
(1010, '2010/12/01', '2011/04/30',4000),
(2020, '2007/05/01', '2007/12/31',2500),
(2020, '2008/01/01', '2010/11/30',4000),
(4040, '2010/10/01', '2012/10/31',3500),
(4040,'2008/08/01', '2009/10/31',1500),
(4040,'2009/11/01', '2010/09/30',2500),
(7070,'2008/01/01', '2008/07/31',1000);
```

## IMAGEM:



## RAMO: DML

---

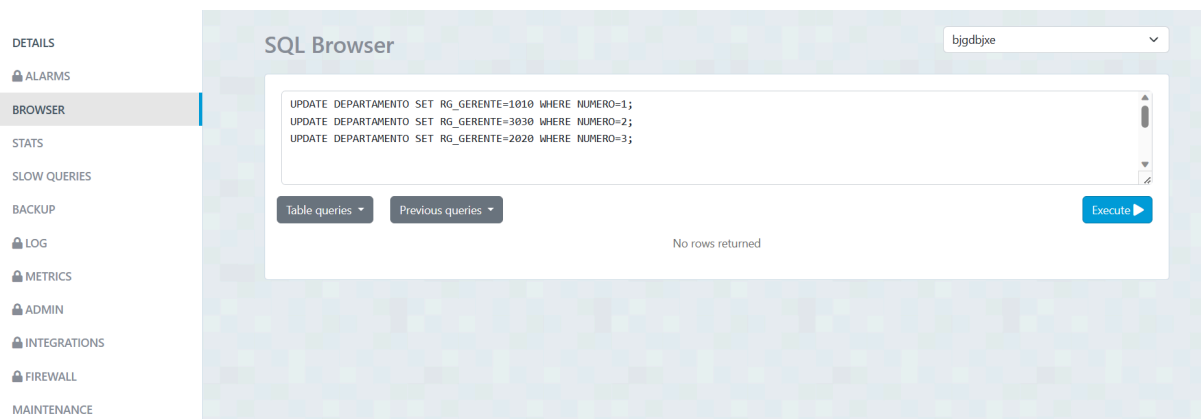
## INSTRUÇÃO: UPDATE

---

### CÓDIGO:

```
UPDATE DEPARTAMENTO SET RG_GERENTE=1010 WHERE NUMERO=1;  
UPDATE DEPARTAMENTO SET RG_GERENTE=3030 WHERE NUMERO=2;  
UPDATE DEPARTAMENTO SET RG_GERENTE=2020 WHERE NUMERO=3;
```

### IMAGEM:



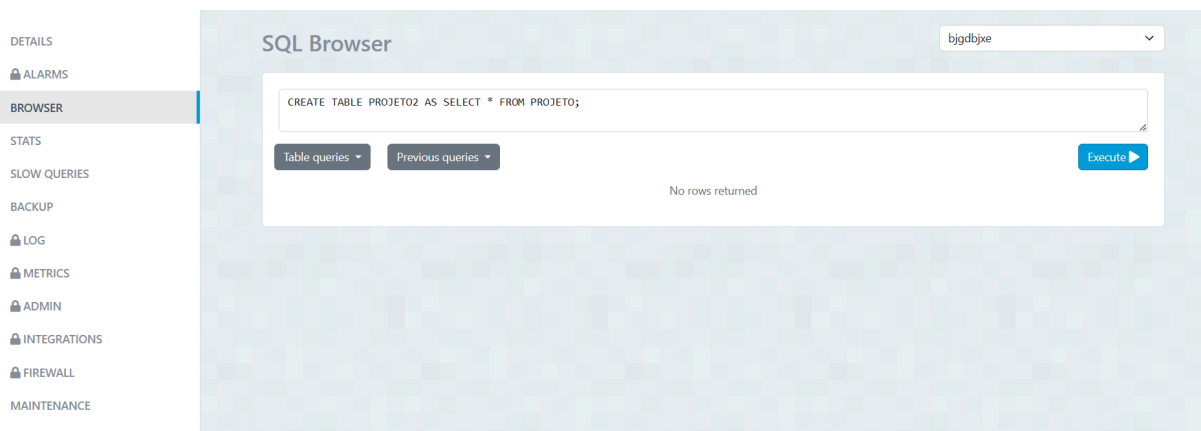
RAMO: DML

INSTRUÇÃO: CREATE .. AS ..

CÓDIGO:

```
CREATE TABLE PROJETO2 AS SELECT * FROM PROJETO;
```

IMAGEM:



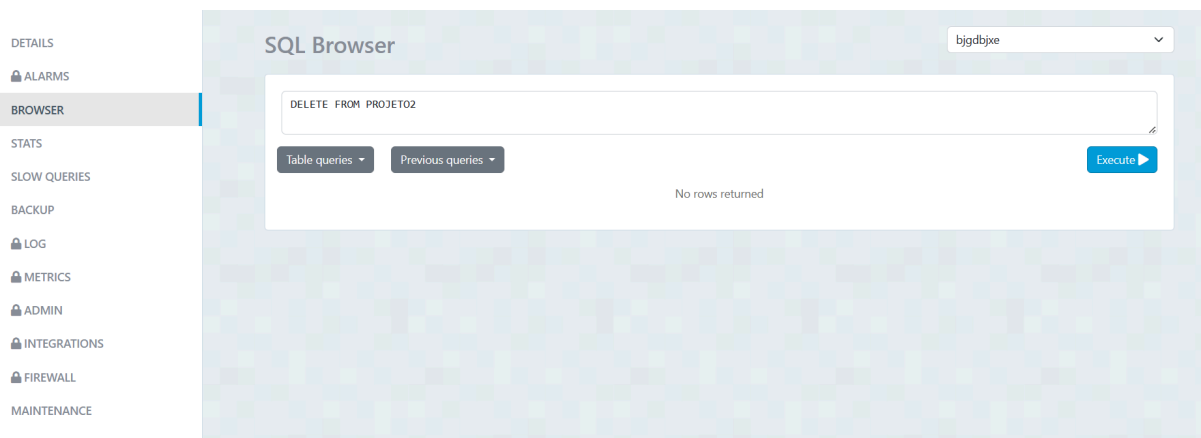
**RAMO: DML**

**INSTRUÇÃO: DELETE**

**CÓDIGO:**

```
DELETE FROM PROJETO2;
```

**IMAGEM:**



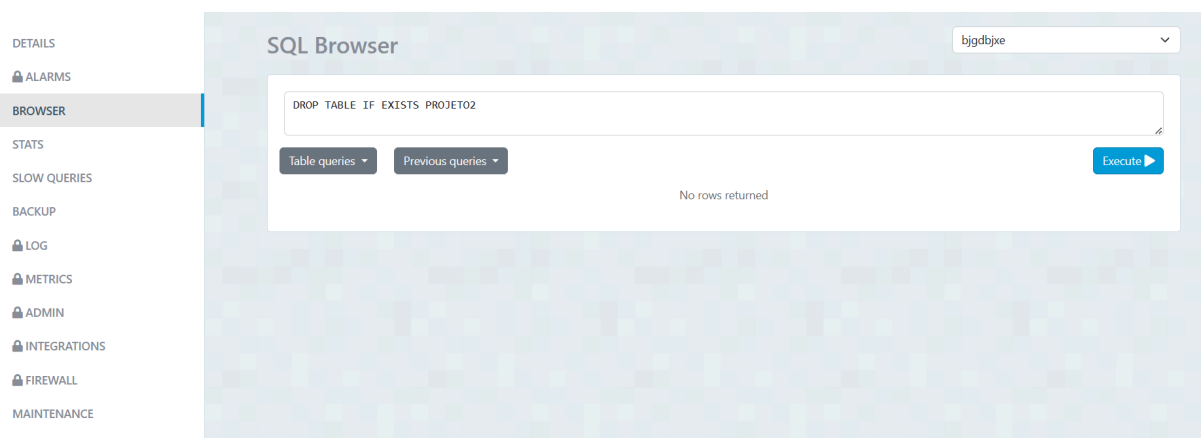
**RAMO: DDL**

**INSTRUÇÃO: DROP TABLE**

**CÓDIGO:**

```
DROP TABLE IF EXISTS PROJETO2;
```

**IMAGEM:**



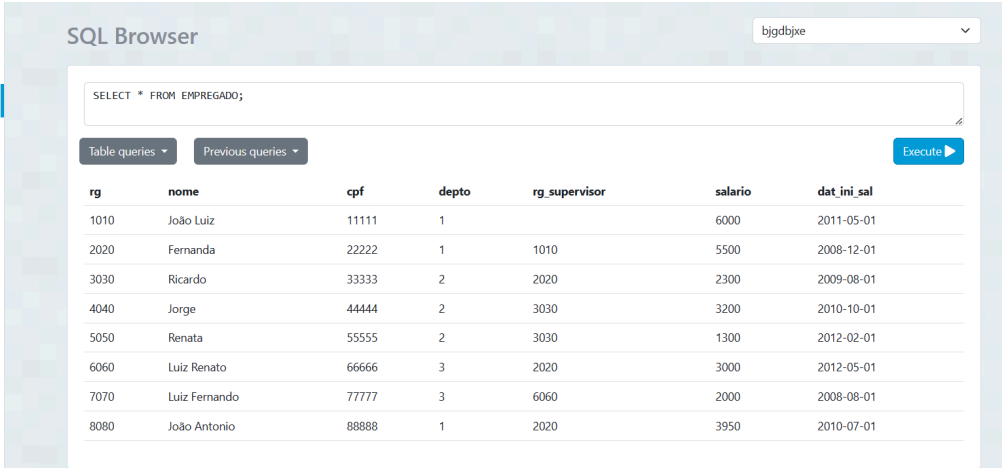
## RAMO: DQL

## INSTRUÇÃO: SELECT SIMPLES

### CÓDIGO:

```
SELECT *  
FROM EMPREGADO;
```

### IMAGEM:



The screenshot shows the SQL Browser interface. On the left is a sidebar with navigation links: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main area is titled 'SQL Browser' and shows a query input field with the text 'SELECT \* FROM EMPREGADO;'. Below the input field are buttons for 'Table queries', 'Previous queries', and an 'Execute' button. The query result is displayed as a table with 7 columns: rg, nome, cpf, depto, rg\_supervisor, salario, and dat\_ini\_sal. The table contains 8 rows of data.

rg	nome	cpf	depto	rg_supervisor	salario	dat_ini_sal
1010	João Luiz	11111	1		6000	2011-05-01
2020	Fernanda	22222	1	1010	5500	2008-12-01
3030	Ricardo	33333	2	2020	2300	2009-08-01
4040	Jorge	44444	2	3030	3200	2010-10-01
5050	Renata	55555	2	3030	1300	2012-02-01
6060	Luiz Renato	66666	3	2020	3000	2012-05-01
7070	Luiz Fernando	77777	3	6060	2000	2008-08-01
8080	João Antonio	88888	1	2020	3950	2010-07-01



## RAMO: DQL

### INSTRUÇÃO: SELECT COM WHERE

#### CÓDIGO:

```
SELECT *  
FROM EMPREGADO WHERE  
SALARIO > 3000;
```

#### IMAGEM:

The screenshot displays the SQL Browser application interface. On the left is a sidebar menu with options: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a database connection 'bjgdbjxe'. A text area contains the SQL query: 'SELECT \* FROM EMPREGADO WHERE SALARIO > 3000;'. Below the query are buttons for 'Table queries', 'Previous queries', and 'Execute'. The 'Execute' button has been clicked, resulting in a table of data. The table has columns: rg, nome, cpf, depto, rg\_supervisor, salario, and dat\_ini\_sal. The data rows are: (1010, João Luiz, 11111, 1, 6000, 2011-05-01), (2020, Fernanda, 22222, 1, 5500, 2008-12-01), (4040, Jorge, 44444, 2, 3200, 2010-10-01), and (8080, João Antonio, 88888, 1, 3950, 2010-07-01).

rg	nome	cpf	depto	rg_supervisor	salario	dat_ini_sal
1010	João Luiz	11111	1		6000	2011-05-01
2020	Fernanda	22222	1	1010	5500	2008-12-01
4040	Jorge	44444	2	3030	3200	2010-10-01
8080	João Antonio	88888	1	2020	3950	2010-07-01

## RAMO: DQL

## INSTRUÇÃO: SELECT COM ORDER BY

## CÓDIGO:

```
SELECT *  
FROM EMPREGADO  
WHERE SALARIO > 3000  
ORDER BY SALARIO;
```

## IMAGEM:

The screenshot displays the SQL Browser application interface. On the left, a sidebar contains navigation links: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a query editor with the following SQL code:

```
SELECT *  
FROM EMPREGADO  
WHERE SALARIO > 3000  
ORDER BY SALARIO;
```

Below the query editor, there are two buttons: 'Table queries' and 'Previous queries'. To the right of the query editor is an 'Execute' button. The results of the query are displayed in a table with the following columns: rg, nome, cpf, depto, rg\_supervisor, salario, and dat\_ini\_sal.

rg	nome	cpf	depto	rg_supervisor	salario	dat_ini_sal
4040	Jorge	44444	2	3030	3200	2010-10-01
8080	João Antonio	88888	1	2020	3950	2010-07-01
2020	Fernanda	22222	1	1010	5500	2008-12-01
1010	João Luiz	11111	1		6000	2011-05-01

## RAMO: DQL

## INSTRUÇÃO: SELECT COM LIMIT

### CÓDIGO:

```
SELECT *  
FROM EMPREGADO  
WHERE SALARIO > 3000  
ORDER BY SALARIO  
LIMIT 3;
```

### IMAGEM:

The screenshot shows the SQL Browser interface. On the left is a sidebar with navigation options: DETAILS, ALARMS, BROWSER (selected), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main area is titled 'SQL Browser' and shows a query editor with the following SQL code:

```
SELECT *  
FROM EMPREGADO  
WHERE SALARIO > 3000  
ORDER BY SALARIO  
LIMIT 3;
```

Below the query editor are two buttons: 'Table queries' and 'Previous queries'. To the right of the query editor is an 'Execute' button. Below the query editor is a table with the following data:

rg	nome	cpf	depto	rg_supervisor	salario	dat_ini_sal
4040	Jorge	44444	2	3030	3200	2010-10-01
8080	João Antonio	88888	1	2020	3950	2010-07-01
2020	Fernanda	22222	1	1010	5500	2008-12-01

## RAMO: DQL

### INSTRUÇÃO: SELECT COM AGRUPAMENTO

#### CÓDIGO:

```
SELECT RELACAO , COUNT(*)  
FROM DEPENDENTE  
GROUP BY RELACAO;
```

#### IMAGEM:

The screenshot displays the SQL Browser application interface. On the left is a sidebar menu with options: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a database connection 'bjgdbjxe'. A text area contains the SQL query: `SELECT RELACAO , COUNT(*)  
FROM DEPENDENTE  
GROUP BY RELACAO;`. Below the query area are buttons for 'Table queries', 'Previous queries', and 'Execute'. The 'Execute' button has been clicked, resulting in a table of results with two columns: 'relacao' and 'count'. The results are as follows:

relacao	count
Filho	4
Cônjuge	2
Filha	1

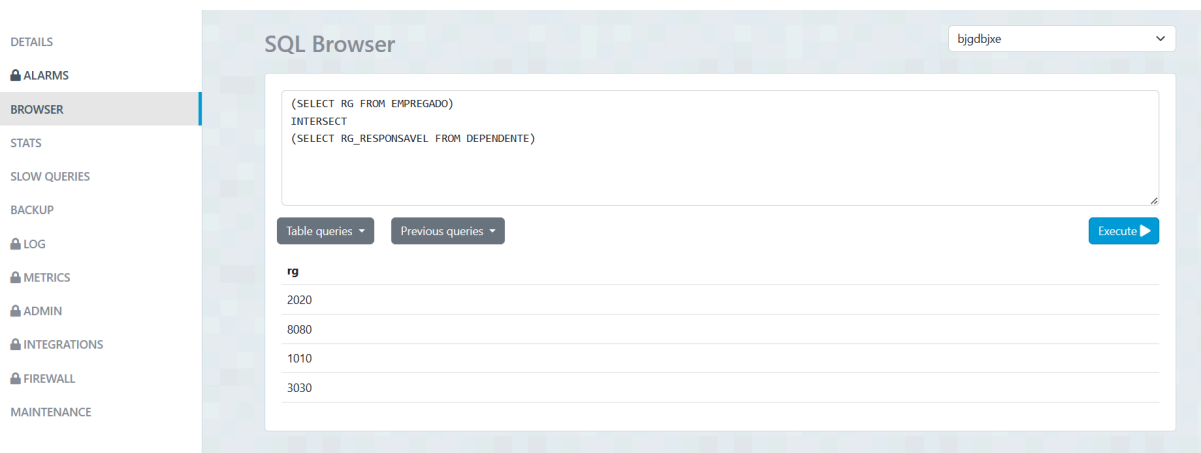
## RAMO: DQL

### INSTRUÇÃO: SELECT COM INTERSECT

#### CÓDIGO:

```
(SELECT RG FROM EMPREGADO)  
INTERSECT  
(SELECT RG_RESPONSAVEL FROM DEPENDENTE);
```

#### IMAGEM:



## RAMO: DQL

## INSTRUÇÃO: SELECT COM UNION

### CÓDIGO:

```
(SELECT NOME FROM EMPREGADO)
UNION
(SELECT NOME_DEPENDENTE AS NOME FROM DEPENDENTE)
ORDER BY NOME;
```

### IMAGEM:

The screenshot shows the SQL Browser interface. On the left is a sidebar with navigation links: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and has a dropdown menu set to 'bjgdbjxe'. The query editor contains the following SQL code:

```
(SELECT NOME FROM EMPREGADO)
UNION
(SELECT NOME_DEPENDENTE AS NOME FROM DEPENDENTE)
ORDER BY NOME;
```

Below the query editor are buttons for 'Table queries', 'Previous queries', and an 'Execute' button. The results are displayed in a table with the following data:

nome
Ana Maria
André
Fernanda
Fernanda Carla
João Antonio
João Luiz

## RAMO: DQL

### INSTRUÇÃO: SELECT COM EXCEPT

#### CÓDIGO:

```
(SELECT NUMERO FROM DEPARTAMENTO)
EXCEPT
(SELECT NUMERO_DEPTO FROM DEPARTAMENTO_PROJETO);
```

#### IMAGEM:

The screenshot displays the SQL Browser application interface. On the left is a sidebar menu with options: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a database connection 'bjgdbxe'. A text area contains the SQL query: 

```
(SELECT NUMERO FROM DEPARTAMENTO)
EXCEPT
(SELECT NUMERO_DEPTO FROM DEPARTAMENTO_PROJETO)
```

 Below the query area are buttons for 'Table queries', 'Previous queries', and an 'Execute' button. The result is shown in a table with one column 'numero' and two rows: 4 and 1.

numero
4
1

## RAMO: DQL

## INSTRUÇÃO: SELECT COM JUNÇÃO INNER

## CÓDIGO:

```
SELECT NOME,NUMERO,NUMERO_PROJETO
FROM DEPARTAMENTO_PROJETO
INNER JOIN PROJETO
ON (DEPARTAMENTO_PROJETO.NUMERO_PROJETO = PROJETO.NUMERO);
```

## IMAGEM:

The screenshot displays the SQL Browser interface. On the left is a sidebar with navigation options: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main area is titled 'SQL Browser' and shows a query editor with the following SQL code:

```
SELECT NOME,NUMERO,NUMERO_PROJETO
FROM DEPARTAMENTO_PROJETO
INNER JOIN PROJETO
ON (DEPARTAMENTO_PROJETO.NUMERO_PROJETO = PROJETO.NUMERO)
```

Below the editor are buttons for 'Table queries', 'Previous queries', and an 'Execute' button. The result of the query is displayed in a table with three columns: nome, numero, and numero\_projeto.

nome	numero	numero_projeto
Financeiro	5	5
Motor	10	10
Prédio Central	20	20



## RAMO: DQL

## INSTRUÇÃO: SELECT COM JUNÇÃO LEFT

## CÓDIGO:

```
SELECT NUMERO_DEPTO,NUMERO_PROJETO,D.NOME,D.NUMERO  
FROM DEPARTAMENTO_PROJETO DP  
LEFT OUTER JOIN DEPARTAMENTO D  
ON (DP.NUMERO_DEPTO = D.NUMERO);
```

## IMAGEM:

The screenshot displays the SQL Browser application interface. On the left is a sidebar with navigation links: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a dropdown menu set to 'bjgdbjxe'. Below the title bar is a text area containing the SQL query: `SELECT NUMERO_DEPTO,NUMERO_PROJETO,D.NOME,D.NUMERO FROM DEPARTAMENTO_PROJETO DP LEFT OUTER JOIN DEPARTAMENTO D ON (DP.NUMERO_DEPTO = D.NUMERO);`. Below the query area are two buttons: 'Table queries' and 'Previous queries'. To the right of these is an 'Execute' button with a play icon. Below the buttons is a table with the following data:

numero_depto	numero_projeto	nome	numero
2	5	Engenharia Civil	2
3	10	Engenharia Mecânica	3
2	20	Engenharia Civil	2

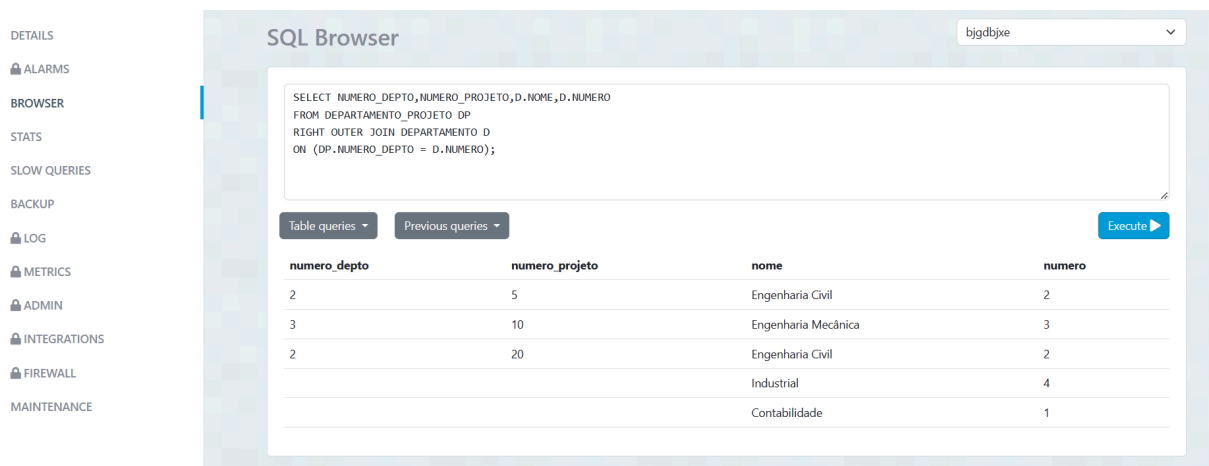
RAMO: DQL

INSTRUÇÃO: SELECT COM JUNÇÃO RIGHT

CÓDIGO:

```
SELECT NUMERO_DEPTO,NUMERO_PROJETO,D.NOME,D.NUMERO
FROM DEPARTAMENTO_PROJETO DP
RIGHT OUTER JOIN DEPARTAMENTO D
ON (DP.NUMERO_DEPTO = D.NUMERO);
```

IMAGEM:



RAMO: DQL

INSTRUÇÃO: SELECT COM JUNÇÃO FULL

CÓDIGO:

```
SELECT * FROM PROJETO P
FULL OUTER JOIN DEPARTAMENTO_PROJETO DP
ON (P.NUMERO = DP.NUMERO_PROJETO);
```

IMAGEM:

The screenshot shows the SQL Browser interface. On the left is a sidebar with navigation links: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and has a dropdown menu set to 'bjgdbjxe'. It contains a text area with the following SQL query:

```
SELECT *
FROM PROJETO P
FULL OUTER JOIN DEPARTAMENTO_PROJETO DP
ON (P.NUMERO = DP.NUMERO_PROJETO);
```

Below the query area are two buttons: 'Table queries' and 'Previous queries'. To the right is an 'Execute' button with a play icon. Below these is a table with the following data:

numero	nome	localizacao	codigo	numero_depto	numero_projeto	horas
5	Financeiro	São Paulo	1	2	5	
10	Motor	Rio Claro	2	3	10	
20	Prédio Central	Campinas	3	2	20	
25	Águas Limpas	Vitória				

## RAMO: DQL

### INSTRUÇÃO: SELECT COMBINANDO INSTRUÇÕES

#### CÓDIGO:

```
SELECT D.NOME AS "Nome do departamento",E.NOME AS "Nome do
gerente",P.NOME AS "Nome do projeto"
FROM DEPARTAMENTO_PROJETO AS DP
RIGHT JOIN DEPARTAMENTO AS D ON (D.NUMERO = DP.NUMERO_DEPTO)
INNER JOIN EMPREGADO AS E ON (D.RG_GERENTE = E.RG)
INNER JOIN PROJETO AS P ON (P.NUMERO = DP.NUMERO_PROJETO)
WHERE E.RG < 7070
GROUP BY D.NOME,E.NOME,P.NOME
HAVING COUNT(D.NOME) = 1
ORDER BY D.NOME ASC
LIMIT 1 OFFSET 1;
```

#### IMAGEM:

The screenshot displays the SQL Browser interface. On the left is a sidebar with navigation options: DETAILS, ALARMS, BROWSER (selected), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main area is titled 'SQL Browser' and contains a text editor with the following SQL query:

```
SELECT D.NOME AS "Nome do departamento",E.NOME AS "Nome do gerente",P.NOME AS "Nome do projeto"
FROM DEPARTAMENTO_PROJETO AS DP
RIGHT JOIN DEPARTAMENTO AS D ON (D.NUMERO = DP.NUMERO_DEPTO)
INNER JOIN EMPREGADO AS E ON (D.RG_GERENTE = E.RG)
INNER JOIN PROJETO AS P ON (P.NUMERO = DP.NUMERO_PROJETO)
WHERE E.RG < 7070
GROUP BY D.NOME,E.NOME,P.NOME
HAVING COUNT(D.NOME) = 1
ORDER BY D.NOME ASC
LIMIT 1 OFFSET 1
```

Below the query editor are buttons for 'Table queries', 'Previous queries', and 'Execute'. The 'Execute' button has been clicked, resulting in a table of results:

Nome do departamento	Nome do gerente	Nome do projeto
Engenharia Civil	Ricardo	Financeiro

RAMO: DQL

---

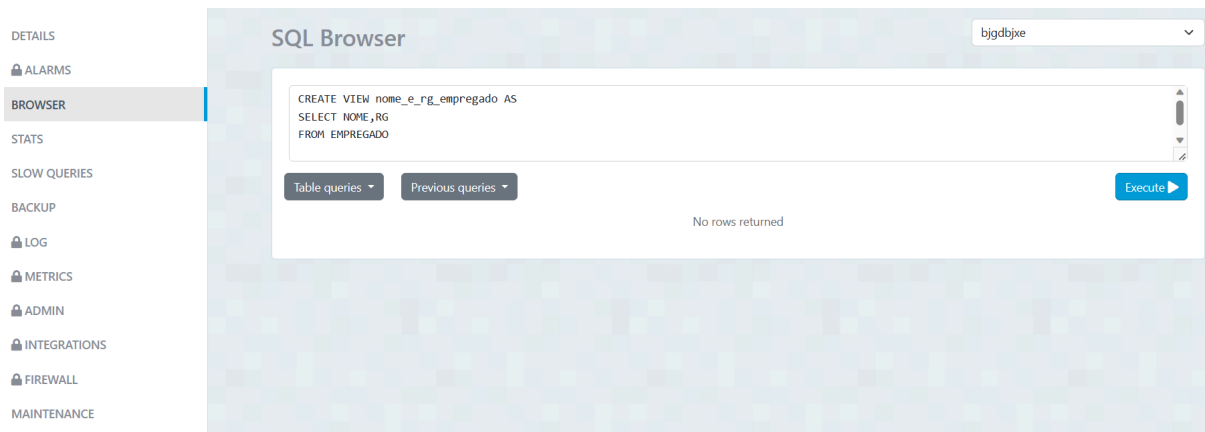
## INSTRUÇÃO: CREATE VIEW

---

CÓDIGO:

```
CREATE VIEW nome_e_rg_empregado AS  
SELECT NOME, RG  
FROM EMPREGADO;
```

IMAGEM:



RAMO: DQL

INSTRUÇÃO: SELECT VIEW

CÓDIGO:

```
SELECT *  
FROM NOME_E_RG_EMPREADO;
```

IMAGEM:

The screenshot displays the SQL Browser application interface. On the left is a sidebar menu with options: DETAILS, ALARMS, BROWSER (highlighted), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main panel is titled 'SQL Browser' and shows a database connection 'bjgdbjxe'. A text area contains the SQL query: 'SELECT \* FROM NOME\_E\_RG\_EMPREGADO'. Below the text area are buttons for 'Table queries', 'Previous queries', and an 'Execute' button. The query result is displayed as a table with two columns: 'nome' and 'rg'.

nome	rg
João Luiz	1010
Fernanda	2020
Ricardo	3030
Jorge	4040
Renata	5050
Luiz Renato	6060
Luiz Fernando	7070
João Antonio	8080

## RAMO: DQL

---

## INSTRUÇÃO: DROP VIEW

---

### CÓDIGO:

```
DROP VIEW NOME_E_RG_EMPREADO;
```

### IMAGEM:

