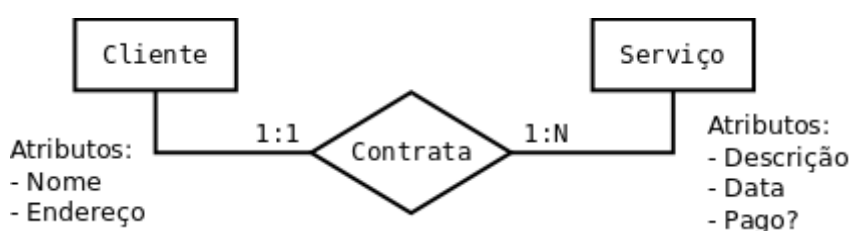


**Instituto Federal Sul-rio-grandense – Campus Camaquã**  
**Disciplina: Banco de Dados II – Turma 2024**  
**Professor: Vinícius Alves Hax**  
**Assunto: Revisão do bimestre**

1) Elabore o diagrama entidade relacionamento de um sistema de atendimento para um profissional de serviços gerais na área hidráulica. Ele deverá registrar as informações básicas de um cliente, como nome e endereço, e também sobre o serviço prestado (que nesse momento deverá ser somente um campo de texto com a descrição do mesmo). É preciso salvar também a data de prestação do serviço. O profissional trabalha sozinho (ou seja, não é preciso registrar quem fez o atendimento) e a única informação que precisa estar armazenada é se o serviço foi pago ou não.

**Resposta:**



---

**Para os exercícios de 2 até 6 utilize as tabelas e registros abaixo:**

```
CREATE TABLE teacher(
    id INT,
    name VARCHAR(200),
    side VARCHAR(200),
    level REAL
);
```

```
CREATE TABLE student(
    id SERIAL,
    name VARCHAR(200),
    teacher_id INT
);
```

```
INSERT INTO teacher (id, name, side, level) VALUES (1, 'Palpatine', 'Dark', 100.0);
INSERT INTO teacher (id, name, side, level) VALUES (2, 'Yoda', 'Light', 100.0);
INSERT INTO teacher (id, name, side, level) VALUES (3, 'Dookan', 'Light', 90.0);
INSERT INTO teacher (id, name, side, level) VALUES (4, 'Qui-gon Jin', 'Light', 90.0);
INSERT INTO teacher (id, name, side, level) VALUES (5, 'Obi Wan', 'Light', 85.0);
INSERT INTO teacher (id, name, side, level) VALUES (6, 'Anakin', 'Light', 95.0);
INSERT INTO teacher (id, name, side, level) VALUES (7, 'Luke', 'Light', 100.0);
INSERT INTO teacher (id, name, side, level) VALUES (8, 'Mace Windu', 'Light', 100.0);
```

```
INSERT INTO student (name, teacher_id) VALUES ('Luke', 2);
INSERT INTO student (name, teacher_id) VALUES ('Anakin', 5);
INSERT INTO student (name, teacher_id) VALUES ('Anakin', 1);
```

```
INSERT INTO student (name, teacher_id) VALUES ('Obi Wan', 4);
INSERT INTO student (name, teacher_id) VALUES ('Qui-gon Jin', 3);
INSERT INTO student (name, teacher_id) VALUES ('Ahsoka', 6);
INSERT INTO student (name, teacher_id) VALUES ('Rei', 7);
INSERT INTO student (name, teacher_id) VALUES ('Kylo Ren', 7);
INSERT INTO student (name) VALUES ('Palpatine');
```

2) Utilizando sub-consultas crie um SQL que mostre o nome do professor do aluno com nome “Obi Wan”.

```
select teacher.name from teacher where teacher.id =
(select student.teacher_id from student where student.name = 'Obi Wan')
```

3) Crie um SQL join que mostre o nome de todos os professores e estudantes, mesmo os professores que não tem nenhum estudante cadastrado e os estudantes que não tem nenhum professor cadastrado.

```
select * from
teacher join
student on
teacher.id = student.teacher_id;
```

Para as questões de 4 à 6 utilize PL/pgSQL

4) Mostre o nome do aluno (student) cujo ID é 5.

```
do $$
declare
nome varchar(500);
begin
-- Primeiro pegamos o id do professor
select student.name
into nome
from student where id = 5;

raise notice 'O nome de ID 5 é %', nome;

end $$
```

5) Mostre todos os professores e seu nível de poder, sendo que o nível de poder deverá ser mostrado em porcentagem em relação ao valor máximo, ou seja, primeiro descubra o maior valor de nível e depois divida o nível de cada personagem por esse máximo.

```
DO $$
DECLARE
maximo real;
nome varchar(50);
nivel real;
BEGIN

SELECT level
INTO maximo
```

```
FROM teacher ORDER BY level DESC LIMIT 1;
```

```
RAISE NOTICE 'Máximo %', maximo;
```

```
FOR nome, nivel IN
```

```
SELECT teacher.name, teacher.level FROM teacher
```

```
LOOP
```

```
-- Exibimos os valores das colunas da linha atual
```

```
RAISE NOTICE 'Professor %, nível % %%', nome, (nivel/maximo)*100;
```

```
END LOOP;
```

```
END $$;
```

6) Mostre todos os alunos, seguido do nome do seu professor formatado da maneira abaixo:

“Eu sou X, e fui treinado por Y, que tem nível de poder Z”

```
DO $$
```

```
DECLARE
```

```
linha record;
```

```
BEGIN
```

```
-- Usamos um loop FOR para iterar sobre cada linha do join
```

```
FOR linha IN
```

```
(SELECT teacher.name AS teacher_name, student.name AS student_name, teacher.level AS  
teacher_level
```

```
FROM teacher
```

```
RIGHT JOIN student ON teacher.id = student.teacher_id)
```

```
LOOP
```

```
RAISE NOTICE 'Eu sou %, e fui treinado por %, que tem nível de poder %',
```

```
linha.student_name, linha.teacher_name, linha.teacher_level;
```

```
END LOOP;
```

```
END $$;
```