

Tarefa 2 – Scripts SQL-DDL e DML em grande escala (Tarefa individual)

Obs.: o texto completo será disponibilizado a semana que vêm no SIGAA

Avisos: Os *scripts* desenvolvidos por esta tarefa 2 foram disponibilizados no *site* da disciplina no SIGAA no arquivo *script_tarefa2.zip*.

Submeta as respostas dos exercícios em um único arquivo (.doc, .odt. ou .pdf) com a identificação do aluno, do professor e da disciplina contendo a respostas dos exercícios enumeradas. Os script .sql deve estar dentro da questão enumerada. Logo não é necessário fazer um *script* .sql para cada resposta, basta enumerar a questão e escrever o código .sql na própria questão.

1) Crie um script em SQL para criar um usuário no SGBD PostgreSQL como o seu nome e sobrenome que seja DBA e de acordo com a regra de nome. **Como resposta deste exercício copie o código *script* SQL de criação deste usuário.**

Exemplo: Caso o nome do estudante seja **Fulano de Tal da Silva + 002** o nome do usuário criado deverá ser **fulanots002**.

2) Faça um script que crie um banco de dados no PostgreSQL Nome definido no exercício 1 usando o nome do usuário criado no exercício 1 adicionado + 2. **Como resposta deste exercício copie o código *script* SQL de criação deste usuário.**

Exemplo: Caso o nome do estudante seja **Fulano de Tal da Silva** o nome do banco de dados é **FulanoTS2**.

3) Rode o *script* Tarefa2_criacao.sql no banco de dados criado no exercício 2. **A resposta desta questão é a captura com a ferramenta Browser do pgAdmin4 com da figura com as tabelas criadas no *schema* public capturando informações (metadados) sobre a Tabela definida para você de acordo com o último dígito à direita da sua matrícula.**

Exemplo se matrícula for 201704632 o último dígito à direita é 2, logo o aluno deverá destacar a **Tabela duração** criada no banco de dados como descrito no exercício 2. O executor deste *script* que deverá ser o usuário criado no exercício 1.

Interpretando os metadados da Figura 1: Na figura 1, na parte superior à direita, temos a seguinte informação **public.duracao/FulanoTS2/postgres@localhost**, neste caso, indica que o esquema onde foi criada a tabela é **public**, o nome da tabela **duração** seguida de um ponto, depois uma **/** em seguida o nome do banco de dados criado com descrito no exercício 2 é **FulanoTS2**, depois outra **/**. Logo após o nodo do usuário que executou o *script*, neste exemplo o **postgres**, na mesma máquina que está instalado o banco indicado por **@localhost**.

Obs.: A Tabela ou relação faz_pertence não foi definida para nenhum aluno. Observe na **Figura 1** os números em vermelho definem cada tabela do aluno (vide Figura 1 na próxima página)

The screenshot shows a PostgreSQL interface. On the left, a tree view of the 'public' schema lists 11 tables. Each table is followed by a red number: curso_unid_acad (0), disciplina (1), duracao (2), estudante (3), faz_pertence (-), instrutor (4), orientador (5), prereq (6), professor (7), sala (8), and turma (9). The 'faz_pertence' table is marked with a red minus sign, indicating it is not defined. On the right, the 'Query Editor' shows a SQL query: `SELECT * FROM public.duracao`. Below the query editor, the 'Data Output' tab displays the results of the query in a table with two columns: 'duracao_id' and 'dia'.

	duracao_id [PK] character varying (4)	dia [PK] character varying (1)
1	A	M
2	A	W
3	A	F
4	B	M
5	B	W
6	B	F

Figura 1 – Resultado ao rodar o script Tarefa2_criacao.sql com a numeração da tabela que será definida de acordo com o último dígito à direita da matrícula do aluno.

4) Faça seu próprio *script* de inserção de três tuplas e alteração de mais três tuplas da tabela definida para você no exercício 3. **Como resposta deste exercício copie o código *script* SQL da inserção e alterações em sql na tabela definida para você de acordo com o último dígito à direita de sua matrícula que foi explicado na questão 3.**

5) Cada estudante fará a consulta com a tabela definida no exercício 3 e também de acordo com a cláusula definida com o último dígito da matrícula à direita. **Como resposta desta questão copie o script do enunciado da consulta SQL criada por você que deverá ser escrito como comentário em SQL juntamente como a código em SQL da consulta.**

Exemplo: Se o último dígito à direita da matrícula for 5, o nome da tabela da consulta é **orientador** (de acordo com o exercício 3) e deverá usar a cláusula do **SELECT** de número 5 do comando **Select** do **SQL** que é **order by**. As consultas que envolverem mais de uma tabela as demais tabelas serão de escolha livre do aluno, neste exemplo, a tabela obrigatório é **orientador**, as demais poderão ser qualquer uma que aparece no Figura 2, incluindo a tabela faz_pertence.

A sintaxe resumida do comando **SELECT** do **PostgreSQL** com o número da cláusula definida (**em vermelho**) para consulta **SQL** está descrita abaixo.

```
SELECT [ ALL | DISTINCT [ ON ( expressão [, ...] ) ] ] * | expressão [ AS nome_de_saída ] [, ...]  
0) [ FROM item_do_from [, ...] ]  
9) [ WHERE condição ou predicado ]  
8) [ GROUP BY expressão de atributos de agrupamento [, ...] ]  
7) [ HAVING condição para agrupamentos [, ...] ]  
6) [ { UNION | INTERSECT | EXCEPT } [ ALL ] seleção ]  
5) [ ORDER BY expressão [ ASC | DESC | USING operador ] [, ...] ]  
4) [ LIMIT { contador | ALL } ]  
3) [ FOR UPDATE [ OF nome_da_tabela [, ...] ] ]
```

Além destas 8 cláusulas principais exploraremos os comandos referentes aos **SQL Join**.

Neste exercício será explorado **LEFT JOIN (2)**, **RIGHT JOIN (1)**.

6) Faça uma consulta que capture as linhas afetadas no exercício 3. Como resposta a questão copie o código *script* **SQL** de criação deste usuário. Além disso, capture a tela com o resultado da execução da consulta que aparece na parte inferior *data output* da tela da ferramenta pgAdmin4.

7) Faça uma consulta que capture os donos dos banco de dados do seu servidor. Como resposta a questão copie o código *script* **SQL** de criação deste usuário. Além disso, capture a tela com o resultado da execução da consulta que aparece na parte inferior *data output* da tela da ferramenta pgAdmin4.

8) Sobre os tipos de dados e os metadados do postgresQL responda a seguinte questões.

a) Faça o código em SQL de uma tabela no do schema *public* com um nome definido por você. Esta tabela deve conter três atributos de tipo de dados diferentes.

Os tipos de atributos existentes em **SQL** no **PostgreSQL** (<https://www.postgresql.org/docs/9.2/datatype.html>).

Como resposta desta letra copie o código **SQL** que cria a tabela.

b) Faça uma consulta em sql das tabelas do banco de dados do seu computador. Como resposta desta letra copie o código *script* **SQL** de criação deste usuário. Além disso, capture a tela com o resultado da execução da consulta que aparece na parte inferior *data output* da tela da ferramenta pgAdmin4.