

Design e Desenvolvimento de Bancos de Dados

— PSet 3 —

Prof. Abrantes Araújo Silva Filho

Data de Entrega: 17/06/2022

LEIA COM ATENÇÃO!

Um *Problem Set* (PSet) é um conjunto de problemas e tarefas difíceis (alguns extremamente difíceis) que o forçarão a estudar e realmente compreender a matéria¹. É **essencial** que você inicie o PSet o mais rápido possível... deixar para começar um PSet faltando um ou dois dias da data de entrega significa que você não conseguirá terminar. Também é **fundamental** que você, em caso de dúvidas ou dificuldades, participe das monitorias e discuta suas dúvidas com o monitor e/ou professor.

Sumário

1	Instruções	2
1.1	Dependência do PSet-1	3
1.2	Como entregar este PSet?	3
2	Queries SQL hierárquicas (recursivas)	4
2.1	Como realizar este PSET?	7

¹Para maiores informações sobre os PSets, leia a Seção 5.3 do *Syllabus* da disciplina.

1 Instruções

Este PSet é uma das atividades **pontuadas** que, conforme detalhado no *Syllabus* (Seções 5 e 6), terão peso de 45% na nota final da disciplina. Por favor siga todas as instruções abaixo para a realização e entrega deste PSet.

Este PSet tem um grande objetivo:

- Fazer com que você aprenda a utilizar *queries* SQL hierárquicas (recursivas) para obter informações de tabelas com auto-relacionamento.

O PSet pode ser feito em grupos de até três alunos desde que as seguintes regras sejam absolutamente seguidas:

1. **Respostas individuais:** o grupo pode discutir o assunto e tirar dúvidas entre si mas, ao final, cada aluno deve escrever sua resposta de forma **separada, individual e distinta** das respostas de seus colegas;
2. **Trabalho integral:** todos os participantes de grupo devem trabalhar de forma integral em **todas as questões**. É uma infração às normas de Integridade Acadêmica desta disciplina dividir questões, por exemplo: um aluno responde um grupo de questões, outro aluno responde outro grupo de questões, e um outro aluno responde ao resto das questões e, depois, todos copiam as respostas entre si. Isso não é permitido, cada aluno deve resolver todas as questões;
3. **Código individual:** cada aluno deve escrever e preparar seu próprio código SQL (incluindo scripts ou o que mais for necessário). Não é permitido compartilhar código entre os alunos; e
4. **Identifique os colegas:** se você trabalhar em grupo você deve identificar, em todos os documentos, códigos, scripts, etc., o nome dos colegas com os quais você trabalhou.

Atenção para as regras de trabalho colaborativo!

Por favor, **LEIA ATENTAMENTE** as regras sobre Integridade Acadêmica do *Syllabus* (Seção 7), em especial a “**Política sobre trabalho colaborativo**” (Seção 7.1), para você saber **como trabalhar em grupo** nesta disciplina. Atividades que quebrarem essas regras terão a nota zerada e os alunos encaminhados à coordenação da UVV para aplicação das penalidades previstas (exceto quando o aluno utilizar a “**Cláusula de Arrependimento**”, Seção 7.2 do *Syllabus*).

As atividades deste PSet devem ser realizadas, preferencialmente, na **Máquina Virtual** da disciplina, que já tem o ambiente todo configurado, os softwares instalados e está pronta para uso. A máquina virtual pode ser baixada no site do Computação Raiz (<https://www.computacaoraiz.com.br>). Caso você opte por não utilizar

a máquina virtual, terá de instalar todos os SGBD em seu próprio computador e configurar o ambiente por conta própria (é difícil, mas não impossível).

Antes de começar, leia integralmente o PSet para ter uma noção geral do que será exigido, para estimar o grau de dificuldade que você terá, e para estimar quantas horas por dia você precisará se dedicar ao PSet. Em caso de dúvidas, entre em contato com o professor ou com os monitores o mais rápido possível.

Lembre-se: deixar para começar um PSet faltando apenas um ou dois dias da data de entrega significa que você não conseguirá terminar! Comece cedo, trabalhe um pouco todo dia, mantenha-se atento à data de entrega e tire suas dúvidas com os monitores e/ou professor. Não ultrapasse a data de entrega!

1.1 Dependência do PSet-1

Este PSet-3 depende do PSet-1² pois, agora, você deverá utilizar o banco de dados “uvv” e o esquema “elmasri” para criar uma nova tabela e inserir os dados com os quais você trabalhará.

ATENÇÃO: Se você não fez o PSet-1 poderá utilizar o script padrão de resposta publicado no Portal do Aluno para criar a estrutura necessária para o PSet 3.

1.2 Como entregar este PSet?

A entrega do resultado deste PSet será feita exclusivamente através do repositório padronizado que você criou no GitHub. Faça o seguinte:

1. Dentro do repositório da disciplina (que foi criado com o nome padronizado `uvv_bd_1_TURMA`), crie o subdiretório `pset3`;
2. Faça a documentação do diretório `pset3` utilizando Markdown;
3. Dentro do diretório deste PSet crie um script SQL com a resposta à questão deste PSet;
4. Envie um e-mail para o professor (`abrantesasf@uvv.br`), até a data limite de entrega, especificando: seu nome completo, matrícula, turma e a URL do repositório GitHub;
5. Observação: mesmo que você inclua ou modifique arquivos no repositório GitHub após a data limite, somente serão avaliadas as respostas comitadas no repositório até a data limite.

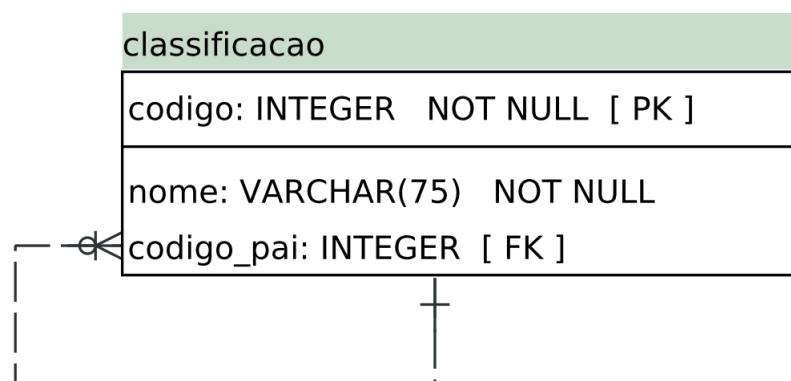
²Durante o PSet-1 você criou o seu usuário, o banco de dados, os esquemas, as tabelas, os relacionamentos e, por fim, fez a inserção dos dados em todas as tabelas.

2 Queries SQL hierárquicas (recursivas)

Nós já vimos que uma tabela pode estar relacionada com ela mesma, criando um auto-relacionamento. Esse tipo de relacionamento é extremamente útil para guardar informações hierárquicas, ou seja, para armazenar quais linhas são “filhas” de outras linhas.

Considere a tabela `classificacao` ilustrada na Figura 1, abaixo. Essa tabela armazena a hierarquia de classificação de produtos de uma empresa real, da área da saúde.

Figura 1: Tabela de Classificação de Produtos



Usando o auto-relacionamento é possível armazenar classificações “pai” e “filhas”, como por exemplo:

1. Enteral (pai)
 - (a) Módulos (filha de “Enteral”)
 - i. Módulos de Fibra (filha de “Módulos”)
 - ii. Módulos Espessantes (filha de “Módulos”)
 - iii. Módulos Protéicos (filha de “Módulos”)
 - iv. Módulos Sacarídeos (filha de “Módulos”)
 - v. Módulos Vitamínicos (filha de “Módulos”)
 - (b) Suplementos (filha de “Enteral”)
 - i. Suplementos Cremosos (filha de “Suplementos”)
 - ii. Suplementos Líquidos (filha de “Suplementos”)
 - iii. ⋮
2. Parenteral (pai)
 - (a) Insumos (filha de “Parenteral”)
 - (b) Limpeza e Higienização (filha de “Parenteral”)
 - (c) ⋮

Sua tarefa nest PSET-3 é descobrir **como criar uma *query* hierárquica (recursiva)** para obter todas as classificações de produtos, indicando claramente que classificação é filha e que classificação é pai. O resultado final deve ser algo semelhante ao exibido na Figura 2:

Figura 2: Resultado de uma *query* hierárquica

	A
1	Enteral
2	Enteral --> A Classificar Enteral
3	Enteral --> A Classificar Enteral --> A Classificar Enteral
4	Enteral --> Complementos em Pó
5	Enteral --> Complementos em Pó --> Complemento em Pó Padrão
6	Enteral --> Complementos Líquidos
7	Enteral --> Complementos Líquidos --> Complemento Líquido Padrão
8	Enteral --> Descartáveis/Material Auxiliar
9	Enteral --> Descartáveis/Material Auxiliar --> Equipo
10	Enteral --> Descartáveis/Material Auxiliar --> Material Auxiliar
11	Enteral --> Descartáveis/Material Auxiliar --> Sonda
12	Enteral --> Dietas em Pó
13	Enteral --> Dietas em Pó --> Dieta Enteral/Oral Especializa
14	Enteral --> Dietas em Pó --> Dieta Enteral/Oral Padrão
15	Enteral --> Dietas em Pó --> Dieta Enteral/Oral Pediátrica
16	Enteral --> Dietas Líquidas
17	Enteral --> Dietas Líquidas --> Especializada
18	Enteral --> Dietas Líquidas --> Padrão
19	Enteral --> Dietas Líquidas --> Pediátrica
20	Enteral --> Módulos
21	Enteral --> Módulos --> Módulos De Fibra
22	Enteral --> Módulos --> Módulos Espessantes
23	Enteral --> Módulos --> Módulos Protéicos
24	Enteral --> Módulos --> Módulos Sacarídeos
25	Enteral --> Módulos --> Módulos Vitamínicos
26	Enteral --> Suplementos
27	Enteral --> Suplementos --> Suplementos Cremosos
28	Enteral --> Suplementos --> Suplementos Cremosos --> Suplemento Cremoso Padrão
29	Enteral --> Suplementos --> Suplementos Líquidos
30	Enteral --> Suplementos --> Suplementos Líquidos --> Suplemento Líquido Padrão
31	Enteral --> Suplementos --> Suplementos Líquidos --> Suplemento Líquido Pediátrico
32	Enteral --> Suplementos --> Suplementos Líquidos --> Suplemento Líquido Especializado
33	Enteral --> Suplementos --> Suplementos Pó
34	Enteral --> Suplementos --> Suplementos Pó --> Suplemento Pó Padrão
35	Frete
36	Frete --> Servicos de Frete
37	Frete --> Servicos de Frete --> Frete em Geral
38	Parenteral
39	Parenteral --> A Classificar Parenteral
40	Parenteral --> A Classificar Parenteral --> A Classificar Parenteral
41	Parenteral --> Descartáveis/Material da Produção
42	Parenteral --> Descartáveis/Material da Produção --> Bolsas
43	Parenteral --> Descartáveis/Material da Produção --> Equipos da Produção
44	Parenteral --> Descartáveis/Material da Produção --> Material Produção

Na Figura 2 eu escolhi os caracteres “-->” para indicar quem é filho de quem. Você pode utilizar o que for mais conveniente pois o que importa é que as relações entre as linhas estejam corretas.

E como os dados estão armazenados na tabela `classificacao`? De modo bem simples! O que faz a hierarquia é a relação entre o atributo `codigo` e o atributo `codigo_pai`. A Figura 3 mostra alguns dos dados armazenados:

Figura 3: Parte dos dados na tabela `classificacao`

codigo	nome	codigo_pai
1	Enteral	
94	A Classificar Enteral	1
95	A Classificar Enteral	94
56	Complementos Líquidos	1
58	Complemento Líquido Padrão	56
57	Complementos em Pó	1
59	Complemento em Pó Padrão	57
2	Descartáveis/Material Auxiliar	1
3	Equipo	2
4	Material Auxiliar	2
5	Sonda	2
6	Dietas em Pó	1
7	Dieta Enteral/Oral Especializa	6
8	Dieta Enteral/Oral Padrão	6
9	Dieta Enteral/Oral Pediátrica	6
10	Dietas Líquidas	1
11	Especializada	10
12	Padrão	10
13	Pediátrica	10
14	Módulos	1
15	Módulos De Fibra	14
16	Módulos Espessantes	14
17	Módulos Protéicos	14
18	Módulos Sacarídeos	14
60	Módulos Vitamínicos	14
99	Suplementos	1
19	Suplementos Cremosos	99
20	Suplemento Cremoso Padrão	19
21	Suplementos Líquidos	99
23	Suplemento Líquido Padrão	21
24	Suplemento Líquido Pediátrico	21
22	Suplementos Líquido Especializado	21
25	Suplementos Pó	99
26	Suplemento Pó Padrão	25
50	Frete	
51	Servicos de Frete	50
52	Frete em Geral	51
27	Parenteral	
96	A Classificar Parenteral	27
97	A Classificar Parenteral	96
28	Descartáveis/Material da Produção	27
29	Bolsas	28
30	Equipos da Produção	28

Note que algumas classificações ilustradas na Figura 3 (Enteral, Frete, Parenteral etc.) **não possuem nada** no atributo `codigo_pai`: isso significa que elas são as classificações de maior nível e todas as outras classificações serão filhas (ou netas ou bisnetas...) de alguma dessas classificações de maior nível.

Para ilustrar os dados armazenados e uma das possíveis repostas do PSET, segue em anexo duas planilhas: “`informacoes.ods`” e “`informacoes.xls`” (para o LibreOffice e o Excel, respectivamente). Consulte essas planilhas para conhecer todos os dados armazenados e um dos padrões de resposta que você deve obter.

2.1 Como realizar este PSET?

Você deve utilizar o arquivo “`pset3.sql`”, que está em anexo, para criar a tabela `classificacao` e inserir todos os dados no banco de dados “`uvv`” que criamos no PostgreSQL no PSet-1.

ATENÇÃO: abra o arquivo “`pset3.sql`” e LEIA AS INSTRUÇÕES de uso que estão no começo do arquivo. Você pode executar esse script SQL diretamente via terminal (e, nesse caso, você precisará alterar o USUÁRIO e a SENHA conforme as instruções no próprio arquivo). Você também pode copiar o código e usar o DBeaver para criar a tabela e inserir os dados.

ATENÇÃO 2: as *queries* recursivas são de difícil entendimento. Como explicado em sala de aula, eu não espero que vocês consigam entender exatamente o funcionamento dessa *query* agora. Durante a gradação, em algum momento, você terá um “estalo”, um “click”, e passará a ter o entendimento necessário de funções recursivas. Nesse momento você entenderá como os SQL recursivos funcionam.

ATENÇÃO 3: mesmo não entendendo completamente como um SQL recursivo funciona, é possível criar o código necessário para este PSET! Basta ler com calma (e fazer as adaptações necessárias) os dois tutoriais a seguir:

- [SQL Server: Introduction to Hierarchical Query Using a Recursive CTE³](https://blog.sqlauthority.com/2012/04/24/sql-server-introduction-to-hierarchical-query-using-a-recursive-cte-a-primer)
- [SQL Server: Recursive CTE⁴](https://www.sqlservertutorial.net/sql-server-basics/sql-server-recursive-cte)

ATENÇÃO 4: os tutoriais acima estão escritos na sintaxe do SQL Server, mas nós usaremos o PostgreSQL. Você terá que fazer as adaptações necessárias.

Bom PSET!

³<https://blog.sqlauthority.com/2012/04/24/sql-server-introduction-to-hierarchical-query-using-a-recursive-cte-a-primer>

⁴<https://www.sqlservertutorial.net/sql-server-basics/sql-server-recursive-cte>