

Bases de Dados 2017/2018

Enunciado do projecto

Parte 3

O terceiro projeto da disciplina de Bases de Dados consiste na implementação da base de dados de uma aplicação de reservas no SGBD POSTGRES passando pelo desenvolvimento do script de criação de tabelas, consultas SQL, restrições de integridade e um protótipo de aplicação web.

Criação da Base de Dados

Usando a linguagem SQL, apresente um ficheiro instruções (*statements*) para criar o esquema de base de dados no SGBD Postgres correspondente ao esquema relacional **apresentado no Anexo A**. Os tipos de dados escolhidos de cada atributo devem ser o mais apropriados. Em particular, no que diz respeito aos tipos de dados e tamanhos dos campos. Devem **ser também especificadas as restrições de integridade correspondentes às chaves primárias e estrangeiras presentes** em cada tabela. A utilização de caracteres acentuados e cedilhas deve ser evitada.

SQL

Apresente, a consulta SQL¹ mais sucinta correspondente a cada uma das seguintes consultas:

- a) Qual o nome do fornecedor que forneceu o maior número de categorias? Note que pode ser mais do que um fornecedor.
- b) Quais os fornecedores primários (nome e nif) que forneceram produtos de todas as categorias simples?
- c) Quais os produtos (ean) que nunca foram repostos?
- d) Quais os produtos (ean) com um número de fornecedores secundários superior a 10?
- e) Quais os produtos (ean) que foram repostos sempre pelo mesmo operador?

¹ Não podem ser utilizadas instruções SQL não façam parte do standard (tais como a instrução LIMIT).

Desenvolvimento da Aplicação

Crie um conjunto de páginas em PHP e HTML simples que permita ao utilizador:

- a) Inserir e remover categorias e sub-categorias;
- b) Inserir e remover um novo produto e os seus respectivos fornecedores (primários ou secundários), garantindo que esta operação seja atómica;
- c) Listar eventos de reposição de um dado produto, incluindo o número de unidades repostas;
- d) Alterar a designação de um produto;
- e) Listar todas as sub-categorias de uma super-categoria, a todos os níveis de profundidade.

Relatório

O projeto será avaliado a partir do relatório entregue pelos alunos e pela discussão. O relatório deverá conter todas as respostas aos itens pedidos acima. Na tabela seguinte indica-se a valorização de cada parte do trabalho a desenvolver.

Item	Valores
Criação da Base de Dados	4
SQL	10
Aplicação	6

O relatório deverá começar com uma folha de rosto com a indicação **“Projeto de Bases de Dados, Parte 3”**, o **nome e número dos alunos**, **a percentagem relativa de contribuição de cada um, juntamente com o esforço (em horas)** que cada elemento do grupo dedicou ao projeto, o **número do grupo**, o **turno** a que o grupo pertence, o **nome do docente de laboratório** e, além da folha de rosto, o relatório deverá ter no máximo **6 páginas**.

Entrega

A entrega no sistema fénix deve ser um ficheiro **zip** estruturado da seguinte forma:

relatorioGG.pdf (onde GG é o número do grupo)	O relatório em pdf onde GG é o número do grupo, contendo os comandos de criação da base de dados , as consultas em SQL e uma explicação da arquitetura da aplicação PHP e das relações
---	---

	entre os diversos ficheiros. Não deve incluir as instruções de população da base de dados.
schema.sql	Ficheiro de criação do esquema da base de dados.
queries.sql	Ficheiro com as consultas SQL.
populate.sql	Ficheiro com instruções para popular as tabelas com dados de teste.
web/	Pasta com os ficheiros HTML e PHP.

O trabalho terá que ser entregue em duas versões:

1. **Versão digital**, em formato ZIP, a entregar via Fénix até às 23h59 da data de entrega.
2. **Versão em papel**, a entregar na portaria do IST TagusPark (LEIC-T) ou na secretaria do DEI no IST Alameda (LEIC-A) até às **17:00h do dia útil seguinte** à data da entrega.

Anexo A

categoria(nome)

- **RI-RE1:** **nome** tem de existir em **categoria_simples** ou em **super_categoria**
- **RI-RE2:** **nome** não pode existir simultaneamente em **categoria_simples** e em **super_categoria**

categoria_simples(nome)

- nome: FK(categoria)

super_categoria(nome)

- nome: FK(categoria)
- **RI-RE3:** **nome** tem de existir na atributo **super_categoria** da relação **constituída**

constituída(super_categoria, categoria)

- super_categoria: FK(super_categoria)
- categoria: FK(categoria)
- **RI-EA1:** não podem existir ciclos super_categoria e categoria
- **RI-RA2:** super_categoria != categoria

produto(ean, design, categoria, forn_primario, data)

- categoria: FK(categoria)
- forn_primario: FK(fornecedor)
- **RI-RE3:** Todo o **EAN** de **produto** tem de existir na relação **fornece_sec**.

fornecedor(nif, nome)

fornece_sec(nif, ean)

- nif: FK(fornecedor)
- ean: FK(produto)
- **RI-EA4:** O **fornecedor** (primário) de um **produto** não pode existir na relação **fornece_sec** para o mesmo produto

corredor(nro, largura)

prateleira(nro, lado, altura)

- nro: FK(corredor)

planograma(ean, nro, lado, altura, face, unidades, loc)

- ean: FK(produto)
- nro, lado, altura: FK(prateleira)

evento_reposicao(operador, instante)

- **RI-EA3:** O **instante** mais recente de reposição tem de ser sempre anterior ou igual à data atual

reposicao(ean, nro, lado, altura, operador, instante, unidades)

- ean, nro, lado, altura: FK(planograma)
- operador, instante: FK(evento_reposição)