

Bases de Dados 2019/2020

Enunciado do projecto

Parte 3

O terceiro projeto da disciplina de Bases de Dados consiste na realização da base de dados do *Sistema TRANSLATE RIGHT* no SGBD POSTGRES passando pelo desenvolvimento do script de criação de tabelas, consultas SQL, restrições de integridade e criação de um protótipo de aplicação web e consultas OLAP.

1. Criação e Preenchimento da Base de Dados

1.1 Esquema de Tabelas

Usando a linguagem SQL, apresente um ficheiro com as 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 para cada atributo devem ser os mais apropriados. Em particular, no que diz respeito aos tipos de dados e tamanhos dos campos. A utilização de caracteres acentuados e cedilhas deve ser evitada.

1.2 Restrições de Integridade

Devem **ser também especificadas as restrições de integridade correspondentes às chaves primárias e estrangeiras presentes** em cada tabela.

Escreva o código que implemente as restrições de integridade RI-1, RI-4, RI-5 e RI-6 da Parte 3 do projecto com as extensões procedimentais do SQL (*Stored Procedures* e *Triggers*) no esquema de tabelas que elaborou anteriormente.

As restrições de integridade **que possam ser definidas sem recurso a** extensões procedimentais (*Stored Procedures* ou *Triggers*), caso existam, devem ser implementadas com mecanismos mais apropriados.

1.3 Preenchimento da Base de Dados

Uma vez criada, a base de dados deve ser preenchida, de forma consistente, com os registos necessários em cada tabela de forma a assegurar que todas as interrogações SQL, solicitadas adiante, tem **resultado não vazio**. A criação de registos e o carregamento da base de dados podem ser realizados através do método que entenda ser mais adequado (manualmente, folha Excel, script SQL, Python, PHP, ou outro).

2. SQL

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

1. Qual o local público onde estão registadas menos anomalias?
2. Qual o local public com mais anomalias de tradução no 1º semestre de 2020?
3. Quais são os utilizadores que registaram em 2020 propostas de correção em todos os locais públicos situados a sul de de Rio Maior (Portugal) (coordenadas de Rio Maior: 39.336775, -8.936379 (cf. Google Maps)
4. Quais são os utilizadores apresentaram propostas de correcção para todas anomalias por eles registadas no ano de 2020.

3. Desenvolvimento da Aplicação

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

- a) Inserir e remover Locais, Itens, anomalias
- b) Inserir, editar e remover correcções e propostas de correcção.
- c) Listar utilizadores
- d) Registar incidências e duplicados
- e) Listar todas as anomalias de incidências registadas na área compreendida entre dois locais públicos
- f) Dados (X,Y) com (latitude, longitude) em graus expressos em notação decimal, listar todas as anomalias registadas nos últimos três meses a mais ou menos (dX, dY) graus de (latitude, longitude).

A solução deve primar pela segurança, prevenindo ataques via SQL INJECTION. Adicionalmente, deve garantir-se a **atomicidade das operações actualização** da base de dados.

4. Índices

Apresente as instruções de criação de índice(s) em SQL para melhorar os tempos de resposta de cada um dos casos indicados abaixo.

Indique, justificando, que tipo de índice(s), sobre que atributo(s) e sobre que tabela(s) faria sentido criar de modo a acelerar a execução de cada uma das interrogações. Suponha que a dimensão das tabelas ultrapassa em várias ordens de grandeza a memória disponível.

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

Suponha que não existe qualquer índice sobre as tabelas para além dos implícitos na declaração das chaves primárias e estrangeiras.

4.1 Devolver o número de items de uma determinada anomalia:

```
select anomalia_id, count(*)
from incidencia
where anomalia_id = some_value
```

4.2 Listar todas as línguas e descrições entre dois momentos no tempo²:

```
select id, lingua, descricao
from anomalia
where ts between <ts1> and <ts2>
and tem_anomalia_redacao = True
and language like '<SOME_PATTERN>%'
```

5. Modelo Multidimensional

Crie na base de dados um esquema em estrela com informação relativa a itens e anomalias tendo como dimensões:

```
d_utilizador(id_utilizador, email, tipo)
d_tempo(id_tempo, dia, dia_da_semana, semana, mes, trimestre, ano)
d_local(id_local, latitude, longitude, nome)
d_lingua(id_lingua, lingua)

f_anomalia(id_utilizador, id_tempo, id_local, id_lingua, tipo_nomalia, com_proposta)
    • id_utilizador: FK(d_utilizador)
    • id_tempo: FK(d_tempo)
    • id_local: FK (d_local)
    • id_lingua: FK(d_lingua)
```

Escreva as instruções SQL necessárias para definir e carregar o esquema em estrela a partir das tabelas existentes, carregando primeiro as tabelas de dimensões e depois a tabela de factos. Os atributos **id_utilizador**, **id_tempo**, **id_local** e **id_lingua** são chaves substitutas.

² E que têm a flag `tem_anomalia_redacao` a `True` e que estão descritas num conjunto de línguas determinadas através do operador `LIKE` contra um padrão prefixo na forma `<SOME_PATTERN>%`.

6. Consultas de Data Analytics

Considerando o esquema em estrela criado na questão anterior, escreva uma interrogação SQL que permita analisar o total de anomalias reportadas por **tipo**, **língua** e **dia_da_semana**. A solução apresentada deverá recorrer às instruções ROLLUP, CUBE, GROUPING SETS, ou à união ("UNION") de cláusulas GROUP BY.

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	2
SQL	5
Aplicação	5
Restrições de integridade	2
Índices	2
Modelo Multidimensional	2
ETL + Data analytics	2

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 **8 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 , as RIs 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. O relatório não necessita incluir a componente OLAP.
schema.sql	Ficheiro de criação do esquema da base de dados.
queries.sql	Ficheiro com as consultas SQL.
populate.sql	Ficheiro com instruções para preencher as tabelas com dados de teste.
RI.sql	Ficheiro de criação das restrições de integridade (triggers & stored procedures)
star_schema.sql	Ficheiro de criação do esquema multidimensional
etl.sql	Ficheiro com o script de carregamento do modelo em estrela
olap.sql	Ficheiro com as consultas OLAP
web/	Pasta com os ficheiros HTML e PHP.

O trabalho terá que ser entregue em formato ZIP com nome entrega-03-GG.zip³ (onde **GG** é o número do grupo), a entregar via Fénix até às 23h59 da data de entrega.

N.B: Os grupos que não seguirem as instruções de entrega **serão descontados**. Elementos que não se encontrem nos devidos ficheiros **não serão tidos em conta para avaliação**.

³ △ O formato do ficheiro deve ser exclusivamente ZIP ou GZ. Outros formatos de arquivo (tais como RAR) não serão aceites.

Anexo A

Modelo Relacional

local_público(latitude, longitude, nome)

item(id, descrição, localização, latitude, longitude)

- latitude, longitude: FK(local_público)

anomalia(id, zona, imagem, lingua, ts, descrição, tem_anomalia_redação)

- tem_anomalia_redação é um atributo booleano

anomalia_tradução(id, zona2, língua2)

- id: FK(anomalia)

(RI-1) A zona da **anomalia_tradução** não se pode sobrepor à zona da **anomalia** correspondente

(RI-2) A língua da **anomalia_tradução** não pode ser igual à língua da **anomalia** correspondente

duplicado(item1, item2)

- item1: FK(item.id)
- item2: FK(item.id)
- **(RI-3)** item1 tem de ser sempre menor que item2. Isto é, um item não pode ser **duplicado** de si próprio e o item com menor id é sempre o item1.

utilizador(email, password)

- **(RI-4)** email de utilizador tem de figurar em **utilizador_qualificado** ou **utilizador_regular**

utilizador_qualificado(email)

- email: FK(utilizador)
- **(RI-5)** email não pode figurar em **utilizador_regular**

utilizador_regular(email)

- email: FK(utilizador)
- **(RI-6)** email não pode figurar em **utilizador_qualificado**

incidência(anomalia_id, item_id, email)

- anomalia_id: FK(anomalia.id)
- item_id: FK(item.id)
- email: FK(utilizador.email)

proposta_de_correção(email, nro, data_hora, texto)

- email: FK(utilizador_qualificado)
- **(RI-7)** email e nro têm de figurar em **correção**

correção(email, nro, anomalia_id)

- email, nro: FK(proposta_de_correção)
- anomalia_id: FK(incidência)