

Bases de Dados II

Exame

Engenharia Informática

29 de Junho de 2015

Duração: 2 horas

Grupo 1

(2 valores)

- Indique, justificando, quais as situações em que acha mais vantajosa a aplicação de cada um dos diferentes tipos de isolamento existentes em MySQL.
- Estando a decorrer uma transacção acha possível uma situação em que o utilizador consiga modificar um conjunto de dados, após os ter eliminado no decorrer da mesma transacção? Justifique.
- Num cenário de concorrência elevada, um utilizador pretende modificar um conjunto de registos da tabela Q com base em cálculos efectuados sobre um conjunto de registos da tabela R. Descreva a melhor forma de o conseguir, garantindo coerência para todas as alterações que se pretendam produzir.

Grupo 2

(7,5 valores)

Nota prévia: para responder a esta questão, assuma-se como utilizador 'root'.

Num servidor MySQL, foram executados os seguintes comandos:

```
CREATE TABLE mysql.perfil(perfil_id int primary key,
nome char(16), S_PRIV enum('Y', 'N'), I_PRIV enum('Y', 'N'),
U_PRIV enum('Y', 'N'), D_PRIV enum('Y', 'N'));
```

```
CREATE TABLE mysql.utilizador(id_user int primary key,
nome char(16), host char(60),
palavra_passe char(41), perfil int, foreign key(perfil) references
mysql.perfil(perfil_id));
```

A tabela *mysql.perfil* contém os seguintes dados:

perfi_id	nome	S_PRIV	I_PRIV	U_PRIV	D_PRIV
1	ligação	N	N	N	N
2	leitor	Y	N	N	N
3	operador	Y	Y	N	N
4	revisor	Y	Y	Y	N
5	gestor	Y	Y	Y	Y
6	teste	Y	N	Y	Y

Esta tabela define os privilégios globais a definir para os utilizadores constantes na tabela **mysql.utilizador**. Por exemplo, se ao utilizador “joao” lhe é atribuído o perfil 5 quando efectua uma ligação de localhost, então deve ser executado o seguinte comando:

```
GRANT SELECT,INSERT,UPDATE,DELETE ON *.* TO joao@localhost
IDENTIFIED BY PASSWORD(palavra_passe);
```

a. Pretende-se um conjunto de triggers que automatize as operações de concessão de privilégios sempre que houver operações que modifiquem os dados da tabela **mysql.utilizador**.

b. Foi executado o seguinte comando:

```
ALTER TABLE mysql.perfil add column G_PRIV enum('Y','N');
```

em que G_PRIV define se o perfil permite que o utilizador possa passar ou não os privilégios recebidos.

Pretende-se que os níveis actuais tenham o valor ‘Y’ para este novo atributo. Crie um procedimento que actualize a tabela **mysql.perfil** e que actualize os privilégios dos utilizadores existentes em **mysql.utilizador**. Proponha ainda as alterações necessárias aos triggers que apresentou na alínea anterior.

c. Os utilizadores existentes no perfil 4 (4, ‘revisor’, ‘Y’, ‘Y’, ‘Y’, ‘N’, ‘Y’) devem passar para um novo perfil 7 (7, ‘editor’, ‘Y’, ‘Y’, ‘N’, ‘Y’, ‘N’). Efectue essa operação.

d. Pretende-se que após um período experimental, as alterações provocadas pelos triggers e procedimentos criados sejam revertidos. Efectue essa operação.

Nota: caso lhe surjam dúvidas de interpretação, indique como resolveu interpretar.

Grupo 3

(3 valores)

- De que forma a utilização de índices influi na performance de um SGBD? Explique, para diferentes situações, quais os melhores métodos para maximizar as vantagens da utilização desse recurso e minimizar eventuais impactos negativos.
- O que entende por índice composto? Em que situações deve ser utilizado? Exemplifique para uma situação concreta a utilização de um índice composto.
- Numa recuperação de dados, que papéis desempenham o redo log (binlog) e os backups existentes?
- Quando um utilizador inicia uma transacção é criado um segmento de rollback para essa transacção. Esse segmento de rollback pode ser utilizado por transacções de outros utilizadores? De que forma?
- O que entende por balanceamento de carga no âmbito dos servidores de bases de dados? Quais são as vantagens e as desvantagens desta abordagem.

Grupo 4

(7,5 valores)

Esta grupo deve ser respondida em folha separada

Considere uma base de dados, denominada **ServicosAcademicos**, destinada a armazenar os diversos dados relativos aos alunos da nossa instituição. Considere, igualmente, a tabela **Alunos** que é definida pelo comando abaixo apresentado.

```
CREATE TABLE Alunos (
    NumAluno INT NOT NULL,
    Nome VARCHAR(30) NOT NULL,
    Email VARCHAR(30) UNIQUE NOT NULL,
    NIF CHAR(9) NOT NULL UNIQUE,
    DataNascimento DATE NOT NULL,
    PRIMARY KEY (NumAluno)
) ENGINE=InnoDB;
```

- Construa um procedimento destinado a efetuar a operação de introdução de dados de novos alunos. Só deve aceitar, como dados de entrada, o nome do aluno, o seu NIF e a sua data de nascimento. Deverá devolver, como parâmetro de saída, o nº atribuído ao aluno. Este procedimento, além de determinar o número do aluno, deverá, também, determinar o seu email.
O email corresponderá ao padrão **primeiroNome.ultimoNome@ipt.pt**. Quando não

for possível criar o email (por exemplo, igual email já foi definido para outro aluno), o procedimento deverá criar o email que será definido pela concatenação da palavra *aluno* juntamente com o *número do aluno*: **alunoNumeroDoAluno@ipt.pt**

NOTA: se criar uma função para definir o email, que receba, como parâmetro, o nome do utilizador e devolva o email, e se criar outra função para determinar o n° do aluno, ficará mais fácil fazer o procedimento.

Funções de manipulação de strings, que eventualmente poderá utilizar:

CONCAT(str1, str2, ...); LEFT(str,len); LENGTH(str); LOCATE(substr,str);
LOWER(str); REVERSE(str); RIGHT(str,len); SUBSTRING(str,pos); SUBSTRIN-
G_INDEX(str,delimiter,count); TRIM(str);

- b. Para a mesma base de dados, considere a tabela denominada *Avaliacoes*. Entre outros atributos, existe o atributo *Nota* onde é registada a avaliação atribuída a um aluno, em função da sua classificação numa determinada unidade curricular. Neste atributo apenas podem ser introduzidos valores compreendidos entre 0 e 20. Crie um *trigger* que, durante a introdução de um registo nesta tabela, valide o valor associado ao campo *Nota*, e só permita a sua introdução, se estiver compreendido entre os valores permitidos.