

Bases de Dados II

Ficha de Exercícios nº6

Engenharia Informática

Docentes

Casimiro Batista

José Casimiro Pereira

Objectivos

Familiarização com Procedimentos

Procedimentos I

Declarações, variáveis de sessão e parâmetros

Crie as seguintes tabelas na base de dados **ficha6** do seu servidor:
(Nota: se a base de dados não existe, crie-a em primeiro lugar.)

```
Create Table alunos(
  Id int,nome Varchar(50),
  Primary Key(Id));
```

```
Create Table disciplina(
  Id int,nome Varchar(30),
  Primary Key(Id));
```

```
Create Table notas(
  aluno int,disc int,nota int,
  Primary Key(aluno,disc),
  Foreign Key(aluno) References alunos(Id),
  Foreign Key(disc) References disciplina(ID));
```

Em seguida introduza os seguintes valores nas tabelas criadas

```
Insert Into alunos values(1,'João Silva');
Insert Into alunos values(2,'Maria Gomes');
Insert Into alunos values(4,'André Figueira');
Insert Into alunos values(5,'Carla Marçal');
Insert Into alunos values(3,'Ana Coelho');
```

```
Insert Into disciplina values(1,'Bases de Dados');
Insert Into disciplina values(4,'Sistemas Operativos');
Insert Into disciplina values(3,'Análise Matemática');
Insert Into disciplina values(2,'Tecnologias da Internet');
Insert Into disciplina values(5,'Redes de Dados');
```

```
Insert Into notas values(1,1,12);
Insert Into notas values(1,3,15);
Insert Into notas values(2,1,15);
Insert Into notas values(3,3,10);
Insert Into notas values(1,4,17);
Insert Into notas values(3,5,12);
Insert Into notas values(4,1,16);
Insert Into notas values(5,1,12);
Insert Into notas values(2,4,18);
Insert Into notas values(3,2,11);
Insert Into notas values(5,4,14);
Insert Into notas values(2,5,18);
Insert Into notas values(4,4,10);
```

Crie o seguinte procedimento

```
DELIMITER $$

DROP PROCEDURE IF EXISTS ficha6.1_proc $$
CREATE PROCEDURE ficha6.1_proc()
BEGIN
    DECLARE tablename varchar(10) default 'alunos';
    CASE tablename
        WHEN 'alunos' THEN SELECT * FROM alunos;
        WHEN 'disciplinas' THEN SELECT * FROM biblioteca;
        WHEN 'notas' THEN SELECT * FROM notas;
    END CASE;
END $$

DELIMITER ;
```

Como verifica é efectuada uma declaração antes do corpo do procedimento. O valor dessa variável permite a escolha de qual dos SELECT vai ser utilizado.

1. Para executar o procedimento deve efectuar que comando?

Como pode verificar, o procedimento devolve sempre o mesmo resultado. Pretende-se que seja feita uma verdadeira escolha de qual das tabelas mostrar. Efectue a seguinte alteração (a sublinhado) ao procedimento .

```
DELIMITER $$

DROP PROCEDURE IF EXISTS ficha6.1_proc $$
CREATE PROCEDURE ficha6.1_proc()
BEGIN
    DECLARE tablename varchar(10) default 'alunos';
    SET tablename=@nomedatabela;
    CASE tablename
        WHEN 'alunos' THEN SELECT * FROM alunos;
        WHEN 'disciplinas' THEN SELECT * FROM diciplinas;
        WHEN 'notas' THEN SELECT * FROM notas;
    END CASE;
END $$

DELIMITER ;
```

2. Como deve proceder de forma a que, quando chama o procedimento, este não resulte em erro?
3. Reescreva o procedimento de forma a que seja dispensável a utilização de uma variável de sessão @nomedatabela.
4. Qual deve ser agora o comando de chamada?

Considere o seguinte procedimento:

```
DELIMITER $$

DROP PROCEDURE IF EXISTS ficha6.2_proc $$
CREATE PROCEDURE ficha6.2_proc(in ID_ALUNO int)
BEGIN
    SELECT disciplina.nome, notas.nota
    FROM disciplina, notas
    WHERE disciplina.Id=notas.disc AND notas.aluno=ID_ALUNO;
END $$

DELIMITER ;
```

5. Após observar o seu funcionamento, reescreva o procedimento anterior de forma a que este devolva em duas variáveis de sessão (@A e @B, por exemplo) a nota mais alta e a média obtida pelo aluno escolhido no parâmetro de entrada.
6. Crie um procedimento que permita ao utilizador escolher uma disciplina e devolva, para o melhor aluno da disciplina, em duas variáveis de sessão, (@aluno e @nota, por exemplo), os valores de nome de aluno e da nota obtida. Considere que a disciplina é introduzida utilizando o seu o nome e em caso de empate só é devolvido o nome de um dos alunos.

Procedimentos II

Condições e “Handlers”

Crie o seguinte procedimento:

```
DELIMITER $$

DROP PROCEDURE IF EXISTS ficha6.3_proc $$
CREATE PROCEDURE ficha6.3_proc(OUT p VARCHAR(50))
BEGIN
    INSERT INTO alunos VALUES(1, 'Sérgio Antunes');
    SET p='Não posso inserir';
end$$

DELIMITER ;
```

7. Quando tenta executar o procedimento, qual o valor da variável de retorno (use @C por exemplo). Porquê?

Altere o seu procedimento da forma seguinte:

```
DELIMITER $$

DROP PROCEDURE IF EXISTS ficha6.3_proc$$
CREATE PROCEDURE ficha6.3_proc(OUT p VARCHAR(50))
BEGIN
    DECLARE q varchar(50);
    DECLARE CONTINUE HANDLER FOR SQLSTATE '23000' SET q='se
persistir em fazer asneira';
    INSERT INTO alunos VALUES(1, 'Sérgio Antunes');
    SET p=Concat('Não consigo inserir ', q);
end$$

DELIMITER ;
```

8. Execute o procedimento e verifique a variável de retorno (use @D, por exemplo). **Nota: não necessita de responder a esta alínea.**
9. Crie dois procedimentos que permitam identificar o erro 1062 e o erro 1048 (ambos têm SQLSTATE=23000). A mensagem deverá ser do tipo: ‘Não consigo inserir se persistir em fazer a asneira nº xxxx’.

Procedimentos III

Controlo de Fluxo

Crie a seguinte tabela:

```
CREATE TABLE ficha6.teste(
id int primary key auto_increment,
valor int);
```

Execute os seguintes procedimentos.

```
DELIMITER $$
DROP PROCEDURE IF EXISTS ficha6.4_proc$$
CREATE PROCEDURE ficha6.4_proc()
BEGIN
    DECLARE a INT Default 0 ;
    ciclo_simples: LOOP
        SET a=a+1;
        select a;
        IF a=50 THEN
            LEAVE ciclo_simples;
        END IF;
    END LOOP ciclo_simples;
END $$
```

```
DELIMITER $$
DROP PROCEDURE IF EXISTS ficha6.5_proc$$
CREATE PROCEDURE ficha6.5_proc()
BEGIN
    DECLARE x INT;
    DECLARE str VARCHAR(255);
    SET x = 1;
    SET str = '';
    WHILE x <= 5 DO
        SET str = CONCAT(str,x,',');
        SET x = x + 1;
    END WHILE;
    SELECT str;
END$$
DELIMITER ;
```

10. Explique o funcionamento dos dois procedimentos.

11. Altere o procedimento **ficha6.4_proc**, de forma a que sejam inseridos no atributo “valor” da tabela **ficha6.teste** apenas valores pares compreendidos entre 0 e 50.

12. Altere o procedimento **ficha6.4_proc**, de forma a que sejam inseridos na tabela **ficha6.teste** valores ímpares compreendidos entre dois valores especificados pelo utilizador. Considere ainda a possibilidade do utilizador ordenar a eliminação (S/N) dos registos previamente existentes na tabela.

Considere o seguinte procedimento:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS ficha6.6_proc $$
CREATE PROCEDURE ficha6.6_proc (IN p varchar(15))
CASE p
    WHEN 'aumentar' THEN UPDATE notas SET
nota=(IF(nota*1.2<20,nota*1.2,20));
    WHEN 'diminuir' THEN UPDATE notas SET
nota=(IF(nota*1.05<20,nota*1.05,20));
ELSE
    UPDATE notas SET nota=(IF(nota*1.1<20,nota*1.1,20));
END CASE $$
DELIMITER ;
```

13. Produza o mesmo resultado, mas utilizando IF...THEN...ELSE

Procedimentos IV

Cursores

Crie a seguinte tabela

```
CREATE TABLE ficha6.sumario_notas (
  id_aluno int,
  nome_aluno varchar(50),
  nota_mais_alta int,
  media decimal(5,2),
  disciplinas_feitas int,
  PRIMARY KEY (id_aluno));
```

Execute o seguinte procedimento:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS ficha6.7_proc $$
CREATE PROCEDURE ficha6.7_proc ()
BEGIN
  DECLARE d INT DEFAULT 0;
  DECLARE nome VARCHAR(50);
  DECLARE num_aluno,nota_mais_alta,num_disciplinas int;
  DECLARE media decimal(5,2);
  DECLARE cur CURSOR FOR SELECT a.Id,a.nome,count(b.disc) as
                                num_disc,avg(b.nota) as
                                media_notas,max(b.nota) as nota_alta
                                FROM alunos a,notas b
                                WHERE a.Id=b.aluno
                                GROUP BY a.Id;
  DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET d=1;
  DECLARE CONTINUE HANDLER FOR SQLSTATE '23000' SET d=1;
  DELETE FROM ficha6.sumario_notas;
  OPEN cur;
  lbl: LOOP
    IF d=1 THEN
      LEAVE lbl;
    END IF;
    IF NOT d=1 THEN
      FETCH cur INTO num_aluno,nome,num_disciplinas,
                    media,nota_mais_alta;
      INSERT INTO ficha6.sumario_notas
        VALUES(num_aluno,nome,num_disciplinas,media,
              nota_mais_alta);
    END IF;
  END LOOP;
  CLOSE cur;
END $$
DELIMITER ;
```

14. Explique detalhadamente o seu funcionamento (sugestão: comente no código as linhas relevantes).
15. Efectue as alterações necessárias na definição da tabela e no procedimento, de forma a que se inclua o campo “nota_mais_baixa”, correspondente à nota mais baixa obtida por cada aluno. Se a nota mais baixa for igual à nota mais elevada, não deve ser indicada na tabela.
16. Com base no exemplo anterior crie um procedimento que crie a seguinte tabela:

Disciplina	Nota mais elevada	Nº de alunos aprovados
Base de Dados	16	4
Sistemas Operativos	18	4
...		
...		

Procedimentos V

Diversos

Crie a seguinte tabela:

```
CREATE TABLE ficha6.novos_alunos (  
    naluno int,  
    nome varchar(25) NOT NULL,  
    apelido varchar(25) NOT NULL,  
    ano_entrada int NOT NULL,  
    email varchar(5) NOT NULL,  
    PRIMARY KEY (naluno)) ENGINE=InnoDB ;
```

17. Crie um procedimento que permita inserir dados na tabela, verificando se o valor de *ano_entrada* é 2019. Caso não seja, deve substituir o valor indicado, na chamada do procedimento por 2019. Pretende-se ainda que o número de aluno (*naluno*) seja verificado e se for inferior a 20000, o registo deve ser rejeitado, não se permitindo a inserção.
18. Adicione ao procedimento anterior a possibilidade de capitalizar as primeiras letras do nome (*nome*) e do apelido(*apelido*), caso o utilizador não o tenha efetuado. Exemplo: antónio Casimiro baptista -> António Casimiro Baptista. **Nota:sugere-se a criação de uma função e a sua utilização no procedimento.**
19. Pretende-se que os alunos a serem introduzidos na tabela possuam email no domínio do IPT (*@ipt.pt). Caso o email a ser introduzido já pertença ao domínio pretendido, o procedimento fará a inserção na tabela sem alterações, sendo que no caso contrário, deverá reter o nome de utilizador e substituir o domínio. Exemplo: josesilva@gmail.com -> josesilva@ipt.pt