

INSTITUTO POLITÉCNICO DE VIANA DO CASTELO

ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

ENGENHARIA DE REDES E SISTEMAS DE COMPUTADORES

BASES DE DADOS

2020/2021

TAREFA 4

Aluno:

Luís Oliveira - 24833

Docente:

Marco Lima



**Instituto Politécnico
de Viana do Castelo**

21 de janeiro de 2021

Conteúdo

Lista de Figuras	v
1 Views,Stored Procedures, triggers e functions)	1
1.1 Continuando a tarefa anterior, os Recursos Humanos após a implementação da nova base dados necessitam algumas funcionalidades para facilitar a gestão:	1
1.2 É necessário um SP(stored procedures) com os parâmetros de entrada (mês, ano, número de dias, utilizador), que processa o vencimento de um utilizador, isto é:	
• Pelo utilizador retornar salário bruto	
• Os descontos são 11% de segurança social sobre o salário	
• Subsídio de refeição = número de dias * 4€	
• Salário liquido = Salario bruto + subsidio de refeição - Descontos(salario bruto*0.11)	1
1.3 Criar uma view que retorna o somatório, número de processamentos e média de processamento agrupados por mês e ano	2
1.4 Uma função que calcula e retorna o subsídio de refeição segundo o número de dias	2
1.5 Introdução	5
1.6 Conclusão	5

Lista de Figuras

1.1	Stored Procedures	1
1.2	Criação da View	2
1.3	Inserção na tabela execução	2
1.4	Criação da tabela histórico	3
1.5	Criação do Trigger	3
1.6	Insert	3
1.7	Update	3
1.8	Delete	4
1.9	Função	4

1 Views,Stored Procedures, triggers e functions)

- 1.1 Continuando a tarefa anterior, os Recursos Humanos após a implementação da nova base dados necessitam algumas funcionalidades para facilitar a gestão:
- 1.2 É necessário um SP(stored procedures) com os parâmetros de entrada (mês, ano, número de dias, utilizador), que processa o vencimento de um utilizador, isto é:
 - Pelo utilizador retornar salário bruto
 - Os descontos são 11% de segurança social sobre o salário
 - Subsídio de refeição = número de dias * 4€
 - Salario liquido = Salario bruto + subsidio de refeição - Descontos(salario bruto*0.11)

```
SQL> CREATE OR REPLACE PROCEDURE RH (Mes IN NUMBER, Ano IN NUMBER, Numero_Dias IN NUMBER, Utiliz
2  IS
3  Salario NUMBER;
4  Desconto NUMBER;
5  Subsidio NUMBER;
6  BEGIN
7  SELECT Salb INTO Salario FROM PROCESSAMENTO WHERE Id_func = Utilizador;
8  Desconto := 0.11 * Salario;
9  Subsidio := 4 * Numero_Dias;
10 INSERT INTO PROCESSAMENTO (Id_processamento,Id_mes , Id_func ,Descontos, Sall,Salb)
11 VALUES (23, Utilizador, Mes, Salario, Desconto, Salario + Subsidio - Desconto);
12 END;
13 /
Procedure created.
```

FIGURA 1.1: Stored Procedures

1.3 Criar uma view que retorna o somatório, número de processamentos e média de processamento agrupados por mês e ano

```
SQL> CREATE VIEW SOMATORIO_PROCESSAMENTO AS (SELECT COUNT(Id_processamento) AS Num_Total_Processamentos, AVG(Sall) AS Media_Sall FROM PROCESSAMENTO GROUP BY Id_mes);
View created.

SQL> insert into PROCESSAMENTO (Id_processamento, Id_mes, Id_func, Descontos, Sall, Salb) values (1, 2, 1, 18, 30, 312);
1 row created.

SQL> insert into PROCESSAMENTO (Id_processamento, Id_mes, Id_func, Descontos, Sall, Salb) values (2, 8, 2, 22, 306, 632);
1 row created.

SQL> insert into PROCESSAMENTO (Id_processamento, Id_mes, Id_func, Descontos, Sall, Salb) values (3, 4, 3, 65, 355, 541);
1 row created.

SQL> insert into PROCESSAMENTO (Id_processamento, Id_mes, Id_func, Descontos, Sall, Salb) values (4, 11, 4, 58, 217, 567);
1 row created.

SQL> insert into PROCESSAMENTO (Id_processamento, Id_mes, Id_func, Descontos, Sall, Salb) values (5, 1, 5, 84, 448, 498);
1 row created.

SQL> SELECT * FROM SOMATORIO_PROCESSAMENTO
2 SELECT * FROM SOMATORIO_PROCESSAMENTO;
SELECT * FROM SOMATORIO_PROCESSAMENTO
*
ERROR at line 2:
ORA-00933: SQL command not properly ended

SQL> SELECT * FROM SOMATORIO_PROCESSAMENTO;

NUM_TOTAL_PROCESSAMENTOS  MEDIA_SALL
-----
1                          30
1                          217
1                          355
1                          306
1                          448
```

FIGURA 1.2: Criação da View

1.4 Uma função que calcula e retorna o subsídio de refeição segundo o número de dias

```
SQL>
SQL> CREATE TABLE TIPO_DE_EXE (
2 Id_Tipo_De_Exe INT NOT NULL,
3 Tipo_De_Exe VARCHAR(50),
4 PRIMARY KEY (Id_Tipo_De_Exe)
5 );

Table created.

SQL> insert into TIPO_DE_EXE values (1,'insert');
1 row created.

SQL> insert into TIPO_DE_EXE values (2,'delete');
1 row created.

SQL> insert into TIPO_DE_EXE values (3,'update');
1 row created.
```

FIGURA 1.3: Inserção na tabela execução

- Pelo utilizador retornar salário bruto
 - Os descontos são 11% de segurança social sobre o salário
 - Subsídio de refeição = número de dias * 4€
 - Salario liquido = Salario bruto + subsidio de refeição - Descontos(salario bruto*0.11)
-

```
SQL> CREATE TABLE HISTORICO (  
  2 Id_Historico INT NOT NULL,  
  3 Id_Tipo_De_Exe INT NOT NULL,  
  4 Data VARCHAR(50),  
  5 PRIMARY KEY (Id_Historico),  
  6 FOREIGN KEY (Id_Tipo_De_Exe) REFERENCES TIPO_DE_EXE (Id_Tipo_De_Exe )  
  7 );  
  
Table created.
```

FIGURA 1.4: Criação da tabela histórico

```
SQL> CREATE OR REPLACE TRIGGER INSERIR AFTER INSERT ON FUNCIONARIO  
  2 BEGIN  
  3 insert into HISTORICO values (1,1,SYSDATE);  
  4 END;  
  5 /  
  
Trigger created.  
  
SQL>  
SQL>  
SQL>  
SQL> CREATE OR REPLACE TRIGGER APAGAR AFTER DELETE ON FUNCIONARIO  
  2 BEGIN  
  3 insert into HISTORICO values (2,2,SYSDATE);  
  4 END;  
  5 /  
  
Trigger created.  
  
SQL>  
SQL>  
SQL> CREATE OR REPLACE TRIGGER ATUALIZAR AFTER UPDATE ON FUNCIONARIO  
  2 BEGIN  
  3 insert into HISTORICO values (3,3,SYSDATE);  
  4 END;  
  5 /  
  
Trigger created.
```

FIGURA 1.5: Criação do Trigger

```
SQL> insert into FUNCIONARIO (Id_func, Nome, Dat_nasc, Tipo_func , Salb ) values (23, 'Luís',SYSDATE,  
1 row created.
```

FIGURA 1.6: Insert

```
SQL> UPDATE FUNCIONARIO SET Salb = 1000 WHERE Id_func = 23;  
1 row updated.
```

FIGURA 1.7: Update

```
SQL> DELETE FROM FUNCIONARIO WHERE Id_func = 23;  
1 row deleted.
```

FIGURA 1.8: Delete

```
SQL> CREATE OR REPLACE FUNCTION CALC (Numero_Dias IN NUMBER) RETURN NUMBER  
2 IS  
3 Subs_Refeicao NUMBER;  
4 BEGIN  
5 Subs_Refeicao := Numero_Dias * 7;  
6 RETURN Subs_Refeicao;  
7 END;  
8 /  
  
Function created.
```

FIGURA 1.9: Função

1.5 Introdução

O objectivo deste trabalho era realizar scripts sql (Funcções, Views, SP, Triggers). Para a realização, utilizamos o sql plus.

1.6 Conclusão

Com a realização deste trabalho, consegui melhorar na minha aprendizagem sobre SP, Triggers e Funções. Tive alguma dificuldade na criação dos SP, mas com a ajuda dos slides consegui interpretar e realizar o código.