

Goiânia 08 de novembro de 2019 - Sexta Feira
 Alunos: Mylena Mariana Pires Barbosa
 Matheus Ferreira Alves Toledo
 Prof.: Edmundo Spoto
 Disciplina: Tópicos Especiais em Banco de Dados

////////////////////////////////////

EX01) Criar uma procedure que conceda 10% de aumento aos empregados.

15:49:36 Matheus > SELECT * FROM EMP;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7839	KING	PRESIDENT		17/11/81	5000	
10	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
20	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7900	JAMES	CLERK	7698	03/12/81	950	
30	7782	CLARK	MANAGER	7839	09/06/81	2450	
10	7934	MILLER	CLERK	7782	23/01/82	1300	
10							

14 linhas selecionadas.

15:49:21 Matheus > ED
 Gravou file afiedt.buf

```

1 create or replace procedure AUMENTA_SAL (PORCENTAGEM IN NUMBER) AS
2 BEGIN
3 UPDATE EMP
4 SET SAL = SAL+PORCENTAGEM*SAL;
```

```
5* END AUMENTA_SAL;
15:49:30 Matheus > /
```

Procedimento criado.

```
15:51:33 Matheus > EXECUTE AUMENTA_SAL(0.1);
```

Procedimento PL/SQL concluÝdo com sucesso.

```
15:52:02 Matheus > SELECT * FROM EMP;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7839	KING	PRESIDENT		17/11/81	5500	
10	7566	JONES	MANAGER	7839	02/04/81	3272,5	
20	7788	SCOTT	ANALYST	7566	19/04/87	3300	
20	7876	ADAMS	CLERK	7788	23/05/87	1210	
20	7902	FORD	ANALYST	7566	03/12/81	3300	
20	7369	SMITH	CLERK	7902	17/12/80	880	
20	7698	BLAKE	MANAGER	7839	01/05/81	3135	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1760	300
30	7521	WARD	SALESMAN	7698	22/02/81	1375	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1375	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1650	0
30	7900	JAMES	CLERK	7698	03/12/81	1045	
30	7782	CLARK	MANAGER	7839	09/06/81	2695	
10	7934	MILLER	CLERK	7782	23/01/82	1430	
10							

14 linhas selecionadas.

//

EX02)Criar uma procedure que imprima a seguinte mensagem na tela: CURSO DE PROCEDURAL OPTION

```
16:07:50 Matheus > ed
Gravou file afiedt.buf
```

```

1 CREATE OR REPLACE PROCEDURE PRINTA_MSG AS
2 BEGIN
3     dbms_output.put_line('CURSO DE PROCEDURAL OPTION');
4* END PRINTA_MSG;
16:07:58 Matheus > /

```

Procedimento criado.

```

16:08:00 Matheus > execute PRINTA_MSG;
CURSO DE PROCEDURAL OPTION

```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

EX03) Criar uma procedure que mostre o numero, nome, data admissão (formatar a data em dd/mm/yyyy) e o nome do departamento do empregado 7369. Caso o salário seja menor do que R\$1000,00, conceder 20% de aumento. Caso seja maior ou igual a R\$1000,00 e menor do que R\$10000,00 conceder 15% de aumento. Caso o salário seja maior ou igual a R\$10000,00, conceder 10% de aumento.

OBS.: Além de mostrar os dados acima, mostrar o salário antigo, o percentual aplicado e o salário novo.

```

CREATE OR REPLACE PROCEDURE FUNC_DADOS(MAT IN EMP.EMPNO%TYPE) AS
CURSOR C1 IS SELECT EMPNO, ENAME, SAL, to_char(HIREDATE,'dd/mm/yyyy') as
data_contratacao, dname
FROM EMP JOIN DEPT ON EMP.DEPTNO=DEPT.DEPTNO
WHERE EMP.EMPNO=MAT;
R_c1 C1%ROWTYPE;
BEGIN
open C1;
FETCH C1 INTO R_c1;
DBMS_OUTPUT.PUT_LINE('Matrícula: ' || R_c1.empno || ' Nome: ' || R_c1.ename || '
Data de contratação: ' || R_c1.data_contratacao || ' Nome do departamento: ' ||
R_c1.dname);
IF R_c1.sal < 1000
THEN
DBMS_OUTPUT.PUT_LINE('Salário antigo: ' || R_c1.sal || ' Percentual aplicado: 20%
' || 'Novo salário: ' || R_c1.sal*1.2);
UPDATE emp
SET sal = 1.2*sal
where empno=MAT;
ELSIF R_c1.sal >= 1000 AND R_c1.sal < 10000
THEN
DBMS_OUTPUT.PUT_LINE('Salário antigo: ' || R_c1.sal || ' Percentual aplicado: 15%
' || 'Novo salário: ' || R_c1.sal*1.15);
UPDATE emp
SET sal = 1.15*sal
where empno=MAT;
ELSE
DBMS_OUTPUT.PUT_LINE('Salário antigo: ' || R_c1.sal || ' Percentual aplicado: 10%
' || 'Novo salário: ' || R_c1.sal*1.1);

```

```

UPDATE emp
SET sal = 1.1*sal
where empno=MAT;
END IF;
END FUNC_DADOS;

```

18:17:32 Matheus > select * from emp;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							

10	7839	KING	PRESIDENT		17/11/81	5500	
20	7566	JONES	MANAGER	7839	02/04/81	3272,5	
20	7788	SCOTT	ANALYST	7566	19/04/87	3300	
20	7876	ADAMS	CLERK	7788	23/05/87	1210	
20	7902	FORD	ANALYST	7566	03/12/81	3300	
20	7369	SMITH	CLERK	7902	17/12/80	18301,68	
30	7698	BLAKE	MANAGER	7839	01/05/81	3135	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1760	300
30	7521	WARD	SALESMAN	7698	22/02/81	1375	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1375	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1650	0
30	7900	JAMES	CLERK	7698	03/12/81	1045	
10	7782	CLARK	MANAGER	7839	09/06/81	2695	
10	7934	MILLER	CLERK	7782	23/01/82	1430	

14 linhas selecionadas.

```

18:17:43 Matheus > execute func_dados(7369);
Matrícula: 7369 Nome: SMITH Data de contratação: 17/12/1980 Nome do
departamento: RESEARCH
Salário antigo:18301,68 Percentual aplicado: 10% Novo salário: 20131,848

```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

4) Inserir um registro na tabela emp com o código igual ao maior código + 1 do menor departamento já cadastrado. Os

outros campos são livres.

MYLENA:>SELECT * FROM EMP ORDER BY EMPNO;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7369	SMITH	CLERK	7902	17/12/80	800	
20	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7782	CLARK	MANAGER	7839	09/06/81	2450	
10	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7839	KING	PRESIDENT		17/11/81	5000	
10	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7900	JAMES	CLERK	7698	03/12/81	950	
30	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7934	MILLER	CLERK	7782	23/01/82	1300	
10							

14 linhas selecionadas.

```
MYLENA:>  CREATE OR REPLACE PROCEDURE NEWEMP
2      (ENAME IN EMP.ENAME%TYPE,
3      JOB IN EMP.JOB%TYPE,
4      MGR IN EMP.MGR%TYPE,
5      HIREDATE IN EMP.HIREDATE%TYPE,
6      SAL IN EMP.SAL%TYPE,
7      COMM IN EMP.COMM%TYPE) AS
8      N_EMPNO EMP.EMPNO%TYPE;
9      N_DEPTNO EMP.DEPTNO%TYPE;
10     BEGIN
11         SELECT MAX(EMPNO)
12         INTO N_EMPNO
13         FROM EMP
14         WHERE DEPTNO = (SELECT MIN(DEPTNO)
15                        FROM EMP);
```

```

16      SELECT MIN(DEPTNO)
17      INTO N_DEPTNO
18      FROM EMP;
19      N_EMPNO := N_EMPNO + 1;
20      INSERT INTO EMP(EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)
21          VALUES(N_EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM,
N_DEPTNO);
22      END NEWEMP;
23  /

```

Procedimento criado.

MYLENA:>EXECUTE NEWEMP('LIAM', 'SALESMAN', 7900, SYSDATE, 3000, 500);

Procedimento PL/SQL concluído com sucesso.

MYLENA:>SELECT * FROM EMP ORDER BY EMPNO;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
-----	-----	-----	-----	-----	-----	-----	-----
20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
20	7566	JONES	MANAGER	7839	02/04/81	2975	
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
10	7782	CLARK	MANAGER	7839	09/06/81	2450	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
10	7839	KING	PRESIDENT		17/11/81	5000	
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
30	7900	JAMES	CLERK	7698	03/12/81	950	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
10	7934	MILLER	CLERK	7782	23/01/82	1300	
10	7935	LIAM	SALESMAN	7900	24/11/19	3000	500

15 linhas selecionadas.

////////////////////////////////////

5) Crie uma procedure que receba como parâmetro o número do departamento e conceda um aumento de 10% para os funcionários que trabalham nesse departamento.

MYLENA:>ED

Gravou file afiedt.buf

```
1 CREATE OR REPLACE PROCEDURE UPDATESALDEPT(N_DEPT IN EMP.DEPTNO%TYPE) AS
2 BEGIN
3   UPDATE EMP
4   SET SAL = (SAL + SAL*0.1)
5   WHERE DEPTNO = N_DEPT;
6* END UPDATESALDEPT;
```

MYLENA:>/

Procedimento criado.

MYLENA:>SELECT * FROM EMP ORDER BY EMPNO;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7369	SMITH	CLERK	7902	17/12/80	800	
20	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7782	CLARK	MANAGER	7839	09/06/81	2450	
10	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7839	KING	PRESIDENT		17/11/81	5000	
10	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7900	JAMES	CLERK	7698	03/12/81	950	
30	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7934	MILLER	CLERK	7782	23/01/82	1300	
10							

10	7935	LIAM	SALESMAN	7900	24/11/19	3000	500
10	7936	LARA	CLERK	7876	24/11/19	100	0

16 linhas selecionadas.

MYLENA:>EXECUTE UPDATESALDEPT(10);

Procedimento PL/SQL concluÝdo com sucesso.

MYLENA:>SELECT * FROM EMP ORDER BY EMPNO;

DEPTNO	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM

20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
20	7566	JONES	MANAGER	7839	02/04/81	2975	
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
10	7782	CLARK	MANAGER	7839	09/06/81	2695	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
10	7839	KING	PRESIDENT		17/11/81	5500	
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
30	7900	JAMES	CLERK	7698	03/12/81	950	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
10	7934	MILLER	CLERK	7782	23/01/82	1430	
10	7935	LIAM	SALESMAN	7900	24/11/19	3300	500
10	7936	LARA	CLERK	7876	24/11/19	110	0

16 linhas selecionadas.

////////////////////////////////////

6) Criar uma procedure que atualize o salário dos funcionários. O número de departamento será passado por parâmetro. O valor a ser aumento também será passado por parâmetro.

MYLENA:>ED

Gravou file afiedt.buf

```

1 CREATE OR REPLACE PROCEDURE UPDATESAL
2 (N_EMPNO IN EMP.EMPNO%TYPE,
3  N_VALOR IN NUMBER) AS
4  SAL EMP.SAL%TYPE;
5  NEW_SAL EMP.SAL%TYPE;
6  BEGIN
7      SELECT SAL
8      INTO SAL
9      FROM EMP
10     WHERE EMPNO = N_EMPNO;
11     NEW_SAL := SAL + (SAL * N_VALOR/100);
12     UPDATE EMP
13     SET SAL = NEW_SAL
14     WHERE EMPNO = N_EMPNO;
15* END UPDATESAL;

```

MYLENA:>/

Procedimento criado.

MYLENA:>SELECT * FROM EMP;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
10	7839	KING	PRESIDENT		17/11/81	5000	
20	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0

30	7900 JAMES	CLERK	7698 03/12/81	950	
10	7782 CLARK	MANAGER	7839 09/06/81	2450	
10	7934 MILLER	CLERK	7782 23/01/82	1300	
20	7935 ALANA	MANAGER	7782 24/11/19	1000	0

15 linhas selecionadas.

MYLENA:>EXECUTE UPDATESAL(7935, 10);

Procedimento PL/SQL concluÍdo com sucesso.

MYLENA:>SELECT * FROM EMP;

DEPTNO	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
-----	-----	-----	-----	-----	-----	-----	-----
10	7839	KING	PRESIDENT		17/11/81	5000	
20	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7900	JAMES	CLERK	7698	03/12/81	950	
10	7782	CLARK	MANAGER	7839	09/06/81	2450	
10	7934	MILLER	CLERK	7782	23/01/82	1300	
20	7935	ALANA	MANAGER	7782	24/11/19	1100	0

15 linhas selecionadas.

MYLENA:>EXECUTE UPDATESAL(7934, 10);

Procedimento PL/SQL concluído com sucesso.

MYLENA:>SELECT * FROM EMP;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							

10	7839	KING	PRESIDENT		17/11/81	5000	
20	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7900	JAMES	CLERK	7698	03/12/81	950	
10	7782	CLARK	MANAGER	7839	09/06/81	2450	
10	7934	MILLER	CLERK	7782	23/01/82	1430	
20	7935	ALANA	MANAGER	7782	24/11/19	1100	0

15 linhas selecionadas.

////////////////////////////////////

7) Fazer uma procedure que receba como parâmetro o número do funcionário e mostrar na tela o número, nome e salário.

MYLENA:>ED

Gravou file afiedt.buf

```
1 CREATE OR REPLACE PROCEDURE SHOWEMP(X_EMPNO IN EMP.EMPNO%TYPE) AS
2 X_ENAME EMP.ENAME%TYPE;
3 X_SAL EMP.SAL%TYPE;
```

```

4 BEGIN
5   SELECT ENAME, SAL
6   INTO X_ENAME, X_SAL
7   FROM EMP
8   WHERE EMPNO = X_EMPNO;
9   DBMS_OUTPUT.PUT_LINE('EMPNO = ' || X_EMPNO);
10  DBMS_OUTPUT.PUT_LINE('ENAME = ' || X_ENAME);
11  DBMS_OUTPUT.PUT_LINE('SAL = ' || X_SAL);
12* END SHOWEMP;
MYLENA:>/

```

Procedimento criado.

```
MYLENA:>SELECT * FROM EMP;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7839	KING	PRESIDENT		17/11/81	5500	
10	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
20	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7900	JAMES	CLERK	7698	03/12/81	950	
30	7782	CLARK	MANAGER	7839	09/06/81	2695	
10	7934	MILLER	CLERK	7782	23/01/82	1430	
10	7935	LIAM	SALESMAN	7900	24/11/19	3300	500
10	7936	LARA	CLERK	7876	24/11/19	110	0

16 linhas selecionadas.

```
MYLENA:>EXECUTE SHOWEMP(7936);
EMPNO = 7936
ENAME = LARA
SAL = 110
```

Procedimento PL/SQL concluído com sucesso.

```
MYLENA:>EXECUTE SHOWEMP(7935);
EMPNO = 7935
ENAME = LIAM
SAL = 3300
```

Procedimento PL/SQL concluído com sucesso.

```
MYLENA:>EXECUTE SHOWEMP(7782);
EMPNO = 7782
ENAME = CLARK
SAL = 2695
```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

8) Fazer uma procedure que receba como parâmetro o número do funcionário e mostre o nome do funcionário.

```
MYLENA:>ED
Gravou file afiedt.buf
```

```
1 CREATE OR REPLACE PROCEDURE SHOWEMPNAME(X_EMPNO IN EMP.EMPNO%TYPE) AS
2 X_ENAME EMP.ENAME%TYPE;
3 BEGIN
4     SELECT ENAME
5     INTO X_ENAME
6     FROM EMP
7     WHERE EMPNO = X_EMPNO;
8     DBMS_OUTPUT.PUT_LINE('ENAME = ' || X_ENAME);
9* END SHOWEMPNAME;
```

```
MYLENA:>/
```

Procedimento criado.

```
MYLENA:>SELECT * FROM EMP;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
10	7839	KING	PRESIDENT		17/11/81	5500	
20	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	3000	

20	7876 ADAMS	CLERK	7788 23/05/87	1100	
20	7902 FORD	ANALYST	7566 03/12/81	3000	
20	7369 SMITH	CLERK	7902 17/12/80	800	
30	7698 BLAKE	MANAGER	7839 01/05/81	2850	
30	7499 ALLEN	SALESMAN	7698 20/02/81	1600	300
30	7521 WARD	SALESMAN	7698 22/02/81	1250	500
30	7654 MARTIN	SALESMAN	7698 28/09/81	1250	1400
30	7844 TURNER	SALESMAN	7698 08/09/81	1500	0
30	7900 JAMES	CLERK	7698 03/12/81	950	
10	7782 CLARK	MANAGER	7839 09/06/81	2695	
10	7934 MILLER	CLERK	7782 23/01/82	1430	
10	7935 LIAM	SALESMAN	7900 24/11/19	3300	500
10	7936 LARA	CLERK	7876 24/11/19	110	0

16 linhas selecionadas.

```
MYLENA:>EXECUTE SHOWEMPNAME(7369);
ENAME = SMITH
```

Procedimento PL/SQL concluído com sucesso.

```
MYLENA:>EXECUTE SHOWEMPNAME(7499);
ENAME = ALLEN
```

Procedimento PL/SQL concluído com sucesso.

```
MYLENA:>EXECUTE SHOWEMPNAME(7935);
ENAME = LIAM
```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

9) Fazer uma procedure que leia todos os empregados e chame outra procedure que retorne a quantidade de dias que cada empregado trabalha na empresa.

Mostre na tela: código, nome e quantidade de dias.

OBS.: Ordenar os empregados por departamento.

```
MYLENA:>ed
Gravou file afiedt.buf
```

```

1  create or replace procedure qtddias(hiredate in emp.hiredate%type) as
2  new number;
3  begin
4      new:= sysdate - hiredate;
5      dbms_output.put_line(' Quantidade de dias = ' || round(new, 1));
6* end qtddias;
MYLENA:>/

```

```

MYLENA:>execute qtddias('22/11/2019');
Quantidade de dias = 7,5

```

Procedimento PL/SQL concluído com sucesso.

```

MYLENA:>ed
Gravou file afiedt.buf

```

```

1  create or replace procedure reademp as
2  cursor c1 is select empno, ename, hiredate
3      from emp
4      order by deptno, empno;
5  begin
6      for rc1 in c1 loop
7          DBMS_OUTPUT.PUT('Codigo = ' || rc1.empno || ' Nome = ' ||
rc1.ename);
8          qtddias(rc1.hiredate);
9      end loop;
10* end reademp;
MYLENA:>/

```

Procedimento criado.

```

MYLENA:>execute reademp;
Codigo = 7782 Nome = CLARK Quantidade de dias = 14052,7
Codigo = 7839 Nome = KING Quantidade de dias = 13891,7
Codigo = 7934 Nome = MILLER Quantidade de dias = 13824,7
Codigo = 7369 Nome = SMITH Quantidade de dias = 14226,7
Codigo = 7566 Nome = JONES Quantidade de dias = 14120,7
Codigo = 7788 Nome = SCOTT Quantidade de dias = 11912,7
Codigo = 7876 Nome = ADAMS Quantidade de dias = 11878,7
Codigo = 7902 Nome = FORD Quantidade de dias = 13875,7
Codigo = 7935 Nome = LIAM Quantidade de dias = 0
Codigo = 7499 Nome = ALLEN Quantidade de dias = 14161,7
Codigo = 7521 Nome = WARD Quantidade de dias = 14159,7
Codigo = 7654 Nome = MARTIN Quantidade de dias = 13941,7
Codigo = 7698 Nome = BLAKE Quantidade de dias = 14091,7
Codigo = 7844 Nome = TURNER Quantidade de dias = 13961,7
Codigo = 7900 Nome = JAMES Quantidade de dias = 13875,7

```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

10) Fazer uma procedure que leia todos os departamentos e, para

cada departamento lido, mostrar na tela: Empregados do XXXXXXXX e depois chamar outra procedure passando o número do departamento. Esta procedure mostrará na tela o nome do funcionário e o nome do gerente.

OBS.: Se algum departamento não possuir empregados, mostrar: O departamento XXXXXXXX não possui empregados.

MYLENA:>ed

Gravou file afiedt.buf

```
1  create or replace procedure empfromdept(numdept emp.deptno%type) as
2  gerente emp.ename%type;
3  type treg is record( ename emp.ename%type,
4                      mgr emp.mgr%type);
5  vreg treg;
6  cursor c1 is select ename, mgr
7  from emp
8  where deptno = numdept
9  order by ename;
10 begin
11  open c1;
12  loop
13      fetch c1 into vreg;
14      if c1%notfound = true and c1%rowcount = 0 then
15          dbms_output.put_line('Nao ha funcionarios');
16      dbms_output.put_line(' -- ');
17      exit;
18      end if;
19      if c1%notfound = true then
20          dbms_output.put_line('Nao ha mais funcionarios');
21      dbms_output.put_line(' -- ');
22      exit;
23      end if;
24      vreg.mgr := nvl(vreg.mgr,0);
25      If vreg.mgr = 0 then
26          dbms_output.put_line('Funcionario: ' || vreg.ename);
27      Else
28          select ename
29          into gerente
30          from emp
31          where empno = vreg.mgr
32          order by ename desc;
33          dbms_output.put_line('Funcionario: ' || vreg.ename || ' |
Gerente: ' || gerente);
34      End If ;
35  end loop;
36* end empfromdept;
MYLENA:>/
```

MYLENA:>ed

Gravou file afiedt.buf

```
1  create or replace procedure departamentos as
2  cursor c1 is select deptno
```



```

3      from dept;
4  begin
5      for rc1 in c1 loop
6          dbms_output.put_line('Empregados do departamento ' ||
rc1.deptno);
7          empfromdept(rc1.deptno);
8      end loop;
9* end departamentos;
MYLENA:>/

```

Procedimento criado.

```

MYLENA:>execute departamentos;
Empregados do departamento 10
Funcionario: CLARK | Gerente: KING
Funcionario: KING
Funcionario: Mariana | Gerente: LIAM
Funcionario: MILLER | Gerente: CLARK
Nao ha mais funcionarios
--
Empregados do departamento 20
Funcionario: ADAMS | Gerente: SCOTT
Funcionario: FORD | Gerente: JONES
Funcionario: JONES | Gerente: KING
Funcionario: LIAM | Gerente: CLARK
Funcionario: SCOTT | Gerente: JONES
Funcionario: SMITH | Gerente: FORD
Nao ha mais funcionarios
--
Empregados do departamento 30
Funcionario: ALLEN | Gerente: BLAKE
Funcionario: BLAKE | Gerente: KING
Funcionario: JAMES | Gerente: BLAKE
Funcionario: MARTIN | Gerente: BLAKE
Funcionario: TURNER | Gerente: BLAKE
Funcionario: WARD | Gerente: BLAKE
Nao ha mais funcionarios
--
Empregados do departamento 40
Nao ha funcionarios
--

```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

11) Fazer uma procedure que leia todos os departamentos. Para cada departamento lido, chamar uma procedure (passando o número do departamento) que selecionará todos os seus empregados. Para cada empregado lido, chamar outra procedure (passando o salário) que calculará o novo salário retornando no mesmo parâmetro. Depois mostrar o código, nome, salário antigo e salário novo. Quando terminar o departamento, retornar a quantidade de empregados e o total de salários do departamento

e mostrar o código, nome, quantidade e total.

Se sal < 1000, aumentar 15%
Se sal >= 1000 e < 5000, aumentar 6%
Se sal >= 5000, aumentar 5%./

```
create or replace procedure departamentos as
    qtdemp number;
    totalsal emp.sal%type;
    cursor c1 is select deptno, dname
                  from dept;
begin
    for rc1 in c1 loop
        dbms_output.put_line('Empregados do departamento ' ||
rc1.deptno);
        empfromdept(rc1.deptno, qtdemp, totalsal);
        dbms_output.put('Codigo: ' || rc1.deptno);
        dbms_output.put(' | Nome: ' || rc1.dname);
        dbms_output.put(' | Quantidade de empregados: ' || qtdemp);
        dbms_output.put_line(' | Total de Salarios: ' || totalsal);
    end loop;
end departamentos;
/
```

```
create or replace procedure novosalario(sal in out emp.sal%type) as
    newsal emp.sal%type;
begin
    If sal < 1000 then
        newsal:= sal + sal * 15/100;
    ElseIf sal >= 5000 then
        newsal:= sal + sal * 5/100;
    else
        newsal:= sal + sal * 6/100;
    End If;
    sal:= newsal;
end novosalario;
/
```

```
create or replace procedure empfromdept(numdept in emp.deptno%type, qtdemp out
number, totalsal out emp.sal%type) as
type treg is record( empno emp.empno%type,
                    ename emp.ename%type,
                    sal_antigo emp.sal%type,
                    sal_novo emp.sal%type);
vreg treg;
cursor c1 is select empno, ename, sal, sal
              from emp
              where deptno = numdept
              order by empno;
begin
    qtdemp:= 0;
    totalsal:= 0;
    open c1;
    loop
```

```

    fetch c1 into vreg;
    if c1%notfound = true and c1%rowcount = 0 then
        dbms_output.put_line('Nao ha funcionarios neste departamento');
        exit;
    end if;
if c1%notfound = true then
    dbms_output.put_line('Nao ha mais funcionarios');
    exit;
    end if;
novosalario(vreg.sal_novo);
    dbms_output.put_line('Codigo: ' || vreg.empno || ' | Nome: ' ||
vreg.ename || ' | Salario antigo: ' || vreg.sal_antigo || ' | Salario novo: ' ||
vreg.sal_novo);
totalsal:= totalsal + vreg.sal_novo;
qtdemp:= qtdemp + 1;
    update emp
    set sal=vreg.sal_novo
    where empno=vreg.empno;
    end loop;
end empfromdept;
/

```

MYLENA:>select * from emp;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
-----	-----	-----	-----	-----	-----	-----	-----
10	7839	KING	PRESIDENT		17/11/81	5000	
20	7566	JONES	MANAGER	7839	02/04/81	2975	
20	7788	SCOTT	ANALYST	7566	19/04/87	2000	
20	7876	ADAMS	CLERK	7788	23/05/87	1100	
20	7902	FORD	ANALYST	7566	03/12/81	3000	
20	7369	SMITH	CLERK	7902	17/12/80	800	
30	7698	BLAKE	MANAGER	7839	01/05/81	2850	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1600	300
30	7521	WARD	SALESMAN	7698	22/02/81	1250	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1500	0
30	7900	JAMES	CLERK	7698	03/12/81	950	
30	7782	CLARK	MANAGER	7839	09/06/81	2450	

10	7934 MILLER	CLERK	7782 23/01/82	1300	
10	7935 LIAM	MANAGER	7782 29/11/19	5000	0
20	7936 Emily			2000	
	7937 Mariana	Manager	7935 06/12/19	3000	0
10					

17 linhas selecionadas.

MYLENA:>execute departamentos();

***** Empregados do departamento 10 *****

Codigo: 7782 | Nome: CLARK | Salario antigo: 2450 | Salario novo: 2597
 Codigo: 7839 | Nome: KING | Salario antigo: 5000 | Salario novo: 5250
 Codigo: 7934 | Nome: MILLER | Salario antigo: 1300 | Salario novo: 1378
 Codigo: 7937 | Nome: Mariana | Salario antigo: 3000 | Salario novo: 3180

Nao ha mais funcionarios

Codigo: 10 | Nome: ACCOUNTING | Quantidade de empregados: 4 | Total de Salarios: 12405

.

 .

***** Empregados do departamento 20 *****

Codigo: 7369 | Nome: SMITH | Salario antigo: 800 | Salario novo: 920
 Codigo: 7566 | Nome: JONES | Salario antigo: 2975 | Salario novo: 3153,5
 Codigo: 7788 | Nome: SCOTT | Salario antigo: 2000 | Salario novo: 2120
 Codigo: 7876 | Nome: ADAMS | Salario antigo: 1100 | Salario novo: 1166
 Codigo: 7902 | Nome: FORD | Salario antigo: 3000 | Salario novo: 3180
 Codigo: 7935 | Nome: LIAM | Salario antigo: 5000 | Salario novo: 5250

Nao ha mais funcionarios

Codigo: 20 | Nome: RESEARCH | Quantidade de empregados: 6 | Total de Salarios: 15789,5

.

 .

***** Empregados do departamento 30 *****

Codigo: 7499 | Nome: ALLEN | Salario antigo: 1600 | Salario novo: 1696
 Codigo: 7521 | Nome: WARD | Salario antigo: 1250 | Salario novo: 1325
 Codigo: 7654 | Nome: MARTIN | Salario antigo: 1250 | Salario novo: 1325
 Codigo: 7698 | Nome: BLAKE | Salario antigo: 2850 | Salario novo: 3021
 Codigo: 7844 | Nome: TURNER | Salario antigo: 1500 | Salario novo: 1590
 Codigo: 7900 | Nome: JAMES | Salario antigo: 950 | Salario novo: 1092,5

Nao ha mais funcionarios

Codigo: 30 | Nome: SALES | Quantidade de empregados: 6 | Total de Salarios: 10049,5

.

 .

***** Empregados do departamento 40 *****

Nao ha funcionarios neste departamento

Codigo: 40 | Nome: OPERATIONS | Quantidade de empregados: 0 | Total de Salarios: 0

.

.....
.

Procedimento PL/SQL concluído com sucesso.

MYLENA:>select * from emp;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM
DEPTNO							
	7839	KING	PRESIDENT		17/11/81	5250	
10	7566	JONES	MANAGER	7839	02/04/81	3153,5	
20	7788	SCOTT	ANALYST	7566	19/04/87	2120	
20	7876	ADAMS	CLERK	7788	23/05/87	1166	
20	7902	FORD	ANALYST	7566	03/12/81	3180	
20	7369	SMITH	CLERK	7902	17/12/80	920	
20	7698	BLAKE	MANAGER	7839	01/05/81	3021	
30	7499	ALLEN	SALESMAN	7698	20/02/81	1696	300
30	7521	WARD	SALESMAN	7698	22/02/81	1325	500
30	7654	MARTIN	SALESMAN	7698	28/09/81	1325	1400
30	7844	TURNER	SALESMAN	7698	08/09/81	1590	0
30	7900	JAMES	CLERK	7698	03/12/81	1092,5	
10	7782	CLARK	MANAGER	7839	09/06/81	2597	
10	7934	MILLER	CLERK	7782	23/01/82	1378	
20	7935	LIAM	MANAGER	7782	29/11/19	5250	0
	7936	Emily				2000	
10	7937	Mariana	Manager	7935	06/12/19	3180	0

17 linhas selecionadas.

////////////////////////////////////

12) Criar uma procedure que receba como parâmetro o número do
funcionário e retornar seu salário total(sal + comm).

SQL> ED

Gravou file afiedt.buf

```

1 CREATE OR REPLACE PROCEDURE SHOWSALCOMM
2 (X_EMPNO IN EMP.EMPNO%TYPE) AS
3 X_ENAME EMP.ENAME%TYPE;
4 X_SAL EMP.SAL%TYPE;
5 X_COMM EMP.COMM%TYPE;
6 SALCOMM EMP.SAL%TYPE;
7 BEGIN
8     SELECT ename, SAL, COMM
9     INTO X_ENAME, X_SAL, X_COMM
10    FROM EMP
11   WHERE EMPNO = X_EMPNO;
12 SALCOMM := X_SAL + nvl(X_COMM,0);
13 DBMS_OUTPUT.PUT_LINE('EMPNO = ' || X_EMPNO);
14 DBMS_OUTPUT.PUT_LINE('ENAME = ' || X_ENAME);
15 DBMS_OUTPUT.PUT_LINE('SAL = ' || X_SAL);
16 DBMS_OUTPUT.PUT_LINE('COMM = ' || nvl(X_COMM,0));
17 DBMS_OUTPUT.PUT_LINE('SAL+COMM = ' || SALCOMM);
18* END SHOWSALCOMM;
SQL> /

```

Procedimento criado.

```

SQL> execute showsalcomm(5000);
EMPNO = 5000
ENAME = MYLENA
SAL = 1100
COMM = 0
SAL+COMM = 1100

```

Procedimento PL/SQL concluído com sucesso.

```

SQL> execute showsalcomm(7499);
EMPNO = 7499
ENAME = ALLEN
SAL = 1760
COMM = 300
SAL+COMM = 2060

```

Procedimento PL/SQL concluído com sucesso.

////////////////////////////////////

13) Criar uma função que receba uma string e retorne a string com espaços em branco.
Ex.: JOSE --> J O S E

MYLENA:>ed
Gravou file afiedt.buf

```

1 create or replace function space(palavra in varchar2) return varchar2 is
2     total varchar2(300);
3     letra varchar2(300);
4     begin

```

```

5      total := '';
6      for cont in 1..length(palavra)
7          loop
8              letra:= '';
9              letra:= substr(palavra,cont,1); --retorna uma substring da posicao cont
ate n(no caso 1 letra)
10         total := total || letra || ' ';
11     end loop;
12     return(total);
13* end;
MYLENA:>/

```

```

MYLENA:>ed
Gravou file afiedt.buf

```

```

1  declare
2      nome varchar2(300):='&nome';
3  begin
4      nome := space(nome);
5      dbms_output.put_line(nome);
6* end;
MYLENA:>/
Informe o valor para nome: mylena
antigo  2:      nome varchar2(300):='&nome';
novo    2:      nome varchar2(300):='mylena';
m y l e n a

```

Procedimento PL/SQL concluído com sucesso.