**/\*1º.- Realizar un bloque Pl/Sql en el que introducimos una letra por teclado y visualizamos cuál es su número ASCII.\*/**

DECLARE

letra CHAR(1);

BEGIN

letra := '&letra';

dbms\_output.put\_line('la letra introducido es '

|| letra

|| ' su número ASCII es '

|| ASCII(letra));

END;

/

**/\*2º.- Realizar un bloque Pl/Sql en el que introducimos el valor de los catetos de un triángulo y visualizamos el valor de su hipotenusa. \*/**

DECLARE

CAT1 REAL(3);

CAT2 REAL(3);

BEGIN

CAT1 := '&CATETO1';

CAT2 := '&CATETO2';

Cat1 := power(cat1,2);

Cat2 := power(cat2,2);

dbms\_output.put\_line('Su hipotenusa es '

|| trunc(sqrt(cat1+cat2),3));

END;

/

**/\*3º.- Realizar un bloque Pl/Sql que nos permita visualizar el abecedario de forma que cada letra aparezca en una línea. \*/**

DECLARE

i REAL := 1;

abecedario real := ascii('a');

BEGIN

FOR i IN 1..27 LOOP

dbms\_output.put\_line(chr(abecedario) || ' ');

abecedario := abecedario+1;

END LOOP;

END;

/

**/\*4º.- Realizar un bloque Pl/Sql en el que introducimos una palabra por teclado y visualizamos cada una de sus letras en una línea. \*/**

DECLARE

palabra VARCHAR2(20);

BEGIN

palabra := '&Palabra';

FOR i IN 1..length(palabra) LOOP

dbms\_output.put\_line(substr(palabra, i, 1));

END LOOP;

END;

/

**/\*5º.- Realizar un bloque Pl/Sql en el que introducimos una palabra por teclado y visualizamos si esta es palíndroma. Una palabra es palíndroma si se lee igual de izquierda a derecha que de derecha a izquierda.\*/**

DECLARE

palabra VARCHAR2(20);

longitud REAL(2);

i REAL := 1;

ispalindromo BOOLEAN := true;

BEGIN

palabra := '&Palabra';

longitud := length(palabra);

WHILE

i < longitud

AND ispalindromo

LOOP

IF ascii(substr(palabra, i, 1)) != ascii(substr(palabra, longitud, 1)) THEN

dbms\_output.put\_line('La palabra no es palindromo.');

ispalindromo := false;

EXIT;

END IF;

longitud := longitud - 1;

i := i + 1;

END LOOP;

IF ispalindromo THEN

dbms\_output.put\_line('La palabra es palindromo.');

END IF;

END;

/

**/\*6º.- Realizar un bloque Pl/Sql en el que introducimos tres números por teclado y visualizamos el mayor de los números introducidos.\*/**

DECLARE

num1 REAL :='&num1';

num2 REAL :='&num2';

num3 REAL :='&num3';

BEGIN

IF num1>num2 and num1>num3 THEN

dbms\_output.put\_line('el mayor numero introducido es : '|| num1);

elsif num2>num1 and num2>num3 THEN

dbms\_output.put\_line('el mayor numero introducido es : '|| num2);

else

dbms\_output.put\_line('el mayor numero introducido es : '|| num3);

END IF;

END;

/

**/\*7º.- Realizar un bloque Pl/Sql en el que introducimos el numero de un empleado e incrementamos el salario de dicho empleado en función del número de empleados de los que es jefe.**

**• Si no es jefe de ningún empleado será 30 euros.**

**• Si es jefe de un empleado la subida será 50 euros.**

**• Si es jefe de 2 empleados la subida será 70 euros.**

**• Si es jefe de más de 2 empleados la subida será 100 euros.**

**• Además, si el empleado es el presidente se incrementará el salario en 30 euros.\*/**

DECLARE

id\_emplee REAL := '&id\_emplee';

id\_manager employee.manager\_id%TYPE;

salario employee.salary%TYPE;

count\_jefe REAL;

BEGIN

SELECT

salary,

nvl(manager\_id, 0)

INTO

salario,

id\_manager

FROM

employee

WHERE

employee\_id = id\_emplee;

SELECT

COUNT(\*)

INTO count\_jefe

FROM

employee

WHERE

id\_emplee = manager\_id

GROUP BY

manager\_id;

IF count\_jefe = 0 THEN

salario := salario + 30;

dbms\_output.put\_line('No eres jefe de nadie,aumenta 30e en el salario: ' ||(salario));

ELSIF count\_jefe = 1 THEN

salario := salario + 50;

dbms\_output.put\_line('Eres jefe de un empleado,aumenta 50e en el salario: ' ||(salario));

ELSIF count\_jefe = 2 THEN

salario := salario + 70;

dbms\_output.put\_line('Eres jefe de dos empleados,aumenta 70e en el salario: ' ||(salario));

ELSE

salario := salario + 100;

dbms\_output.put\_line('Eres jefe de más de 2 empleados,aumenta 100e en el salario: ' ||(salario));

END IF;

IF ( id\_manager = 0 ) THEN

salario := salario + 30;

dbms\_output.put\_line('Eres presidente aumenta 30e mas en el salario: ' ||(salario + 30));

END IF;

END;

/

**8º.- Realizar un bloque Pl/Sql en el que introducimos el número de un empleado y modificamos el salario o la comisión de dicho empleado en función de las siguientes premisas:**

**• Si el empleado pertenece al departamento 13 su comisión será la comisión media del departamento al que pertenece.**

**• Si el empleado pertenece al departamento 14 su comisión será la comisión mínima del departamento al que pertenece.**

**• Si el empleado pertenece al departamento 23 su comisión será la comisión máxima del departamento al que pertenece.**

**• Si el empleado pertenece al departamento 10 su salario será el salario mínimo del departamento al que pertenece.**

**• Si el empleado pertenece al departamento 12 su salario será el salario máximo del departamento al que pertenece**

**• Si el empleado pertenece a cualquier otro departamento su salario será el salario medio del departamento al que pertenece.**

DECLARE

nombre\_empleado employee.last\_name%TYPE := upper('&NombreEmpleado');

id\_departamiento employee.department\_id%TYPE;

BEGIN

SELECT

department\_id

INTO id\_departamiento

FROM

employee

WHERE

first\_name = nombre\_empleado;

IF id\_departamiento = 13 THEN

UPDATE employee

SET

commission = (

SELECT

AVG(nvl(commission, 0))

FROM

employee

WHERE

department\_id = 13

GROUP BY

department\_id

)

WHERE

department\_id = 13

AND first\_name = nombre\_empleado;

ELSIF id\_departamiento = 14 THEN

UPDATE employee

SET

commission = (

SELECT

MIN(nvl(commission, 0))

FROM

employee

WHERE

department\_id = 14

GROUP BY

department\_id

)

WHERE

department\_id = 14

AND first\_name = nombre\_empleado;

ELSIF id\_departamiento = 23 THEN

UPDATE employee

SET

commission = (

SELECT

MAX(commission)

FROM

employee

WHERE

department\_id = 23

GROUP BY

department\_id

)

WHERE

department\_id = 23

AND first\_name = nombre\_empleado;

ELSIF id\_departamiento = 10 THEN

UPDATE employee

SET

commission = (

SELECT

MIN(salary)

FROM

employee

WHERE

department\_id = 10

GROUP BY

department\_id

)

WHERE

department\_id = 10

AND first\_name = nombre\_empleado;

ELSIF id\_departamiento = 12 THEN

UPDATE employee

SET

commission = (

SELECT

MAX(commission)

FROM

employee

WHERE

department\_id = 12

GROUP BY

department\_id

)

WHERE

department\_id = 12

AND first\_name = nombre\_empleado;

ELSE

UPDATE employee

SET

commission = (

SELECT

AVG(salary)

FROM

employee

WHERE

department\_id = id\_departamiento

GROUP BY

department\_id

)

WHERE

department\_id = id\_departamiento

AND first\_name = nombre\_empleado;

END IF;

END;

/

**9º.- Crear una vista denominada salarios\_caros, con los mismos campos que la tabla empleados, pero en castellano, en donde vamos a guardar todos los empleados que tienen salarios superiores a 2500.**

CREATE VIEW salarios\_caros (

id\_empleado,

apellido,

nombre,

inicial,

id\_trabajo,

id\_jefe,

fecha\_contrato,

salario,

comision,

id\_departamiento

) AS

SELECT \* FROM employee WHERE salary > 2500;