

Exercici 1: (ESQUEMA PRODUCTES)

Realitza una funció anomenada “bonus_salesman_cachinero_marc”. Una funció per calcular el bonus a un venedor en funció de les seves comandes.

Els paràmetres de la funció son 2 i no poden ser null:

- un id de treballador, exemple: 103
- un any, exemple 2015

Cal validar els paràmetres:

- que la id correspongui a un treballador i que el treballador sigui ‘Sales Manager o Sales Representative’.
- que l’any sigui no més gran que l’any actual o sigui posterior al 2000.

Si els paràmetres no son correctes la funció retorna null,

Si els paràmetres son correctes, cal cercar comandes:

- comandes on el venedor sigui l’id que passa com a paràmetre.
- realitzades a l’any que indica el paràmetre.

El venedor obté un bonus de 150\$ per comanda, llavors el bonus total es el resultat de:

- $150 * \text{total_comandes}$

Com a resultat de la funció cal retornar el bonus obtingut per venedor.

SCRIPT DE PROVES:

```
set serveroutput on
```

```
declare
```

```
result number;
```

```
id_emp number;
```

```
any_vendes number;
```

```
begin
```

```
---paràmetre null
```

```
id_emp:=null;
```

```
any_vendes:=1995;
```

```
DBMS_OUTPUT.PUT_LINE('----- Crida amb paràmetre null');
```

```
result:=bonus_salesman(id_emp,any_vendes);
```

```
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador '||nvl(to_char(id_emp),'null')||' l'any '||any_vendes
```

```
||' és :'||nvl(to_char(result),'null'));
```

```
---any incorrecte
```

```
id_emp:=54;
```

```
any_vendes:=2025;
```

```
DBMS_OUTPUT.PUT_LINE('----- any incorrecte');
```

```
result:=bonus_salesman(id_emp,any_vendes);
```

```
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador '||nvl(to_char(id_emp),'null')||' l'any '||any_vendes
```

```
||' és :'||nvl(to_char(result),'null'));
```

```
---empleat incorrecte
```

```
id_emp:=107;
```

```

any_vendes:=2017;
DBMS_OUTPUT.PUT_LINE('----- empleat incorrecte');
result:=bonus_salesman(id_emp,any_vendes);
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador '||nvl(to_char(id_emp),'null')||' l'any
'||any_vendes
                                ||' és :'||nvl(to_char(result),'null'));

---empleat correcte
id_emp:=54;
any_vendes:=2017;
DBMS_OUTPUT.PUT_LINE('----- empleat correcte');
result:=bonus_salesman(id_emp,any_vendes);
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador '||nvl(to_char(id_emp),'null')||' l'any
'||any_vendes
                                ||' és :'||nvl(to_char(result),'null'));

end;

```

ahora falta este chucha ejercicio x q el 3 es jodido de cojones voy a intentar escribirlo: **CO**n esto apruebas??

No lo se porque el 1 no lo puedo comprobar si va no me va en el oracle live x q da demasiados errores la base. que si que va COÑO
que te los pruebo
te los estoy probando

Cross in the bar?

```

Statement processed.
----- Crida amb paràmetre null
-- el bonus del treballador null l'any 1995 és: null
----- any incorrecte
-- el bonus del treballador 54 l'any 2025 és: null
----- empleat incorrecte
-- el bonus del treballador 107 l'any 2017 és: null
----- empleat correcte
-- el bonus del treballador 54 l'any 2017 és: 0

```

es lo que tira el plsqli

Entonces esta bien o no con lo que te pase?
Esque no se como va

cuanto vale cada ej?

1. 3,5 deberias de tener buena nota a mi los dos me devuelven cosas, pero tiene sentido con el enunciado y el script de pruebas?

le paso el script al gpt + el enunciado y le digo que lo adapte de parecer se parece.
yo mientras este bien me la pela

yo diria q esta bien,osea de cosas que parece que tienen sentido,me raya que devuelva cero,pero a lo mejor es que es cero,who knows

espera

espera

sisi espero me sigue dando 0

sii m salg es k viene la gorda (same,si te sales me salgo) quee va tt s no m ve estoy a penultima n chetao,y la foquita? la foca apañandoselas

yo tuve qque escribir todo a mano en un drive xd a mi me los ha psaso via moodle

mirame s el 1 esta biiiiien pero si ya te lo he mirao

por 1190834930p94'2

voy a escribir el 3 por si acaso

y el arque? chetado

2. 3.5
3. 3

CROSSBAR EL 1 ESTA BIEN??

```
CREATE OR REPLACE FUNCTION bonus_salesman_cachinero_marc (  
  p_employee_id NUMBER,  
  p_year NUMBER  
) RETURN NUMBER IS  
  v_bonus NUMBER;  
  v_total_orders NUMBER;  
  v_employee_job_title VARCHAR2(255);  
  v_current_year NUMBER := EXTRACT(YEAR FROM SYSDATE);  
BEGIN  
  -- Validate employee_id and get the job title  
  SELECT job_title  
  INTO v_employee_job_title  
  FROM employees  
  WHERE employee_id = p_employee_id;
```

```

-- Check if the job title is 'Sales Manager' or 'Sales Representative'
IF v_employee_job_title NOT IN ('Sales Manager', 'Sales Representative') THEN
    RETURN NULL;
END IF;

-- Validate the year
IF p_year > v_current_year OR p_year < 2000 THEN
    RETURN NULL;
END IF;

-- Calculate the total number of orders for the given employee and year
SELECT COUNT(*)
INTO v_total_orders
FROM orders
WHERE salesman_id = p_employee_id
    AND EXTRACT(YEAR FROM order_date) = p_year;

-- Calculate the bonus
v_bonus := 150 * v_total_orders;

RETURN v_bonus;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        -- If no employee is found with the given ID
        RETURN NULL;
    WHEN OTHERS THEN
        -- Handle other exceptions
        RETURN NULL;
END;
/

SCRIPT:
SET SERVEROUTPUT ON;

DECLARE
    result NUMBER;
    id_emp NUMBER;
    any_vendes NUMBER;
BEGIN
    -- Test case: Parameter null
    id_emp := NULL;
    any_vendes := 1995;
    DBMS_OUTPUT.PUT_LINE('----- Crida amb paràmetre null');
    result := bonus_salesman_cachinero_marc(id_emp, any_vendes);
    DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador ' || NVL(TO_CHAR(id_emp),
'null') || ' l''any ' || any_vendes || ' és: ' || NVL(TO_CHAR(result), 'null'));

```

```

-- Test case: Incorrect year
id_emp := 54;
any_vendes := 2025;
DBMS_OUTPUT.PUT_LINE('----- any incorrecte');
result := bonus_salesman_cachinero_marc(id_emp, any_vendes);
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador ' || NVL(TO_CHAR(id_emp),
'null') || ' l"any ' || any_vendes || ' és: ' || NVL(TO_CHAR(result), 'null'));

-- Test case: Incorrect employee
id_emp := 107;
any_vendes := 2017;
DBMS_OUTPUT.PUT_LINE('----- empleat incorrecte');
result := bonus_salesman_cachinero_marc(id_emp, any_vendes);
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador ' || NVL(TO_CHAR(id_emp),
'null') || ' l"any ' || any_vendes || ' és: ' || NVL(TO_CHAR(result), 'null'));

-- Test case: Correct employee
id_emp := 54;
any_vendes := 2017;
DBMS_OUTPUT.PUT_LINE('----- empleat correcte');
result := bonus_salesman_cachinero_marc(id_emp, any_vendes);
DBMS_OUTPUT.PUT_LINE('-- el bonus del treballador ' || NVL(TO_CHAR(id_emp),
'null') || ' l"any ' || any_vendes || ' és: ' || NVL(TO_CHAR(result), 'null'));
END;
/

```

Exercici 2: (ESQUEMA_HR)

Realitzar el procediment anomenat “emp_dep_cachinero_marc” que llisti els treballadors d’un departament.

- El parametre de entrada del procediment es:
 - id, identificador d’un departament, no pot ser null.

Cal validar el paràmetre

- que l’ ID no sigui null.
- que l’ ID correspongui a un departament que existeixi.

si el parametre no es correcte el procediment acabara sense llistar res.

Si el parametre es correcte, es treura una llista que tindra una capçalera amb el nom del departament a continuacio la llista de treballadors. Cal tenir present el cas d’un departament sense treballadors, llavors nomes es llistara la capçalera amb el nom del departament.

Les dades a llistar de cada empleat son les següents:

- FIRST_NAME, LAST_NAME, HIRE_DATE, JOB_TITLE.

Les dades s’han de mostrar en columnes de mida fixa,excepte la capçalera.

Exemple del que s’ha de mostrar:

Departament <nom del departament>

Bruce Ernst 21-05-2007 Programmer

SCRIPT DE PROBES:

```
declare
v_id number;
begin
-- prova departament null
DBMS_OUTPUT.PUT_LINE('-- prova departament null-----');
v_id:=null;
emp_dept(v_id);
DBMS_OUTPUT.PUT_LINE("");
-- prova departament que NO existeix
DBMS_OUTPUT.PUT_LINE('-- prova departament que NO
existeix-----');
v_id:=333;
emp_dept(v_id);
DBMS_OUTPUT.PUT_LINE("");
-- prova departament sense treballadors
DBMS_OUTPUT.PUT_LINE('-- prova departament sense
treballadors-----');
```

```

v_id:=180;
emp_dept(v_id);
DBMS_OUTPUT.PUT_LINE("");
-- prova departament amb treballadors
DBMS_OUTPUT.PUT_LINE('-- prova departament amb
treballadors-----');
v_id:=30;
emp_dept(v_id);
DBMS_OUTPUT.PUT_LINE("");
end;

```

```

CREATE OR REPLACE PROCEDURE emp_dep_cachinero_marc (p_id IN
departments.department_id%TYPE)
IS
-- Variables per emmagatzemar informació del departament
v_department_name departments.department_name%TYPE;
v_count          NUMBER;

-- Cursor per llistar els empleats del departament
CURSOR emp_cursor IS
SELECT e.first_name, e.last_name, TO_CHAR(e.hire_date, 'DD-MM-YYYY') AS
hire_date, j.job_title
FROM employees e
JOIN jobs j ON e.job_id = j.job_id
WHERE e.department_id = p_id;

BEGIN
-- Comprovar si el paràmetre d'entrada no és nul
IF p_id IS NULL THEN
DBMS_OUTPUT.PUT_LINE('El paràmetre id no pot ser null. ');
RETURN;
END IF;

-- Comprovar si el departament existeix
BEGIN
SELECT department_name INTO v_department_name
FROM departments
WHERE department_id = p_id;
EXCEPTION
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('El departament amb id ' || p_id || ' no existeix. ');
RETURN;
END;

```

```

-- Comprovar si el departament té treballadors
SELECT COUNT(*) INTO v_count
FROM employees
WHERE department_id = p_id;

-- Mostrar el nom del departament
DBMS_OUTPUT.PUT_LINE('Departament: ' || v_department_name);
DBMS_OUTPUT.PUT_LINE('-----');

-- Si el departament té treballadors, llistar-los
IF v_count > 0 THEN
  FOR emp IN emp_cursor LOOP
    DBMS_OUTPUT.PUT_LINE(RPAD(emp.first_name, 12) || RPAD(emp.last_name,
12) || RPAD(emp.hire_date, 12) || emp.job_title);
  END LOOP;
END IF;

END emp_dep_cachinero_marc;
/

```

Bargos pruebame si va que a mi no me carga el oracle live

y esto??
pero
eso lo hepuesto yo eh
y si que va
lo q te he pegao

con el script de probes y todo
funciona funciona

El script de pruebas me lo puedes editar?

D locos gracias

Pero el script de pruebas lo editaste? o no hace falta tiene que ir el script de pruebas con el codigo sbs

pa q
editar pa q
espera

DECLARE

v_id NUMBER;

BEGIN

-- prova departament null

DBMS_OUTPUT.PUT_LINE('-- prova departament null-----');

v_id := NULL;

emp_dep_cachinero_marc(v_id);

DBMS_OUTPUT.PUT_LINE("");


```

-- prova departament que NO existeix
DBMS_OUTPUT.PUT_LINE('-- prova departament que NO existeix-----');
v_id := 333;
emp_dep_cachinero_marc(v_id);
DBMS_OUTPUT.PUT_LINE("");

-- prova departament sense treballadors
DBMS_OUTPUT.PUT_LINE('-- prova departament sense treballadors-----');
v_id := 180;
emp_dep_cachinero_marc(v_id);
DBMS_OUTPUT.PUT_LINE("");

-- prova departament amb treballadors
DBMS_OUTPUT.PUT_LINE('-- prova departament amb treballadors-----');
v_id := 30;
emp_dep_cachinero_marc(v_id);
DBMS_OUTPUT.PUT_LINE("");
END;
/

```

Exercici 3: (esquema HR)

Crea tres paquets amb el conjunt de funcions que trobaràs en el fitxer que acompanya a l'examen:

- funcions_employees0523.txt

En el fitxer funcions_employees0523.txt hi ha funcions independents que s'han de agrupar dins de tres paquets que s'han de dir:

- pkg_employees0523_cachinero_marc
- pkg_jobs0523_cachinero_marc
- pkg_hr_utilitats0523_cachinero_marc

Al paquet pkg_employees0523_cachinero_marc s'han de posar les funcions <<f_empleat....>>.

Al paquet pkg_jobs0523_cachinero_marc s'han de posar les funcions <<f_job...>>

Al paquet pkg_hr_utilitats0523_cachinero_marc s'ha de posar la funció <<normalizar_string>>.

Cal crear les dues parts del paquet: especificació i cos. A part d'especificació només s'han de declarar les funcions que estan definides com a públiques, les que estan encapçalades amb el comentari <<--FUNCIO PÚBLICA>>.

Una vegada compilats els paquets cal fer proves, per això tens l'altre fitxer:

- funcions_script_proves0523.txt

Hay que adaptar el script para llamar las funciones que creaste

**vale-----no entiendo el 3,el g me
hace cosas raras**

**a partir de lo d la mariina hay que
crear 3 scripts es de hacer paquetes.
Dile que intente hacer el ultimo que
ese creo que es mas facil que
comprrenda el chpt —a partir de q?**

d LOCOS

tee marque

pasale el archivo ese y qqu te haga 3 archivos llamados como dice ella ns. Csas de marina.

**solo se qquee tienes que pasar la bd d HR y el script d la marrina el
funcions_employees0523.txt**

aaa coño

ahora entiendes?

sisi

d locoos

no entiendo,me peta.

por

ns si se tienen que hacer archiivos a parte le ppregunto a marina espera

Son como s fueeeran 3 scripts en el mismo (3 paquetes) CACHIIIIII!!!!!!

dime

me lo paso el arque pero no tengo ni idea de que tieene qque devoolveer si el script d
pruuebas va biieen ya esta

el arrrques me paso esto pero no que cambiar

Falta una funcion pero ns cual y no meee va mrame de arreglarlooo plz

entiendes algo? nop estoy probando

Ni que sea haz que funcione esto xd

me devuelve esto:
Unsupported Command

Statement processed.
treballador: 110
treballador: 110
treballador: 148
treballador: -99

Unsupported Command
treballador: 110
treballador: 110
treballador: 148
treballador: -99

Statement processed.
job id és: AD_PRES
job id és: AD_PRES
job id és: AC_MGR

TUUUUUUUUU
CACHI
GILIPOLLAS
HERMANO
SI TE HABIA DICHO ESO HACE COMO 30 MINS
TE HABIA PREGUNTADO ABAJO
siisi esta bien entonces sii devuelve esta bien creo
Tiooo estoy eestreesado cjoones y eel aques preeguuntandomeee xd jajsjsjaajaj
si se acaba de ir sii
el scrip de probas tal cual

pasamee eesto con eel scrriipppt prroovas
lo he pasado
y devuelve lp de arriba
con el script de proves
ppasme eeeel que veas qqu esteee mejor el del arrrrques o el tuuuuuyyoo

Crossbar?

```

CREATE OR REPLACE PACKAGE pkg_ht_utilitats0523_Cachinero_Marc AS
function normalitzar_string
(p_string varchar2,p_convertir boolean)
return varchar2;
END pkg_ht_utilitats0523_Cachinero_Marc;
/
CREATE OR REPLACE PACKAGE BODY pkg_hr_utilitats0523_Cachinero_Marc AS
function normalitzar_string
(p_string varchar2,p_convertir boolean)
return varchar2
is
v_string varchar2(50);
v_string_out varchar2(50);
v_paraula varchar2(50);
v_paraula_aux varchar2(50);
v_pos number;
primera_vegada boolean;
-- agafa l'string d'entrada, elimina blancs a dreta i esquerra
-- i posa el primer caracter e majúscula i la resta en minúscula
begin
-- comprovar que no sigui null
if p_string is null then return null;
end if;
-- comprovar que la longitud no sigui més de 50
if length(trim(p_string))>50 then return null;
end if;

-- elimina blancs a dreta i esquerra
v_string:=trim(p_string);

-- agafar cada paraula i convertir-la
primera_vegada:=true;
while v_string is not null loop
-- busquem la posició del blanc, si no n'hi ha la posició és el final de l'string
v_pos:= instr(v_string,' ',1,1);
if v_pos=0 then v_pos:=length(v_string)+1; end if;

-- fem substring per trobar la paraula
v_paraula:=substr(v_string,1,v_pos-1);

-- CONVERTIM la primera lletra a majúscula i la resta a minúscula
-- si l'opció convertir és true
if p_convertir then
if length(v_paraula)>1 then

```

```

v_paraula_aux:=upper(substr(v_paraula,1,1))||lower(substr(v_paraula,2));
else
v_paraula_aux:=upper(v_paraula);
end if;
else
v_paraula_aux:=v_paraula;
end if;

```

```

-- concatenem la paraula a l'string de sortida
if primera_vegada then
v_string_out:= v_paraula_aux;
primera_vegada:=false;
else
v_string_out:= v_string_out||' '||v_paraula_aux;
end if;

```

```

-- escurcem l'string per continuar buscant paraules a convertir,
-- aprofitem per eliminar blancs a l'esquerra
v_string:=ltrim(substr(v_string,v_pos+1));
end loop;

```

```

return v_string_out;
end normalitzar_string;
END pkg_hr_utilitats0523_Cachinero_Marc;
/

```

```

CREATE OR REPLACE PACKAGE pkg_employees0523_Cachinero_Marc AS

```

```

function f_job(nom jobs.job_title%type)
return jobs.job_id%type;

```

```

function f_empleat_per_cognom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type)
return employees%rowtype;

```

```

function f_empleat_per_cognom_i_nom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type,
p_nom employees.first_name%type)
return employees%rowtype;

```

```

END pkg_employees0523_Cachinero_Marc;
/

```

```

CREATE OR REPLACE PACKAGE BODY pkg_employees0523_Cachinero_Marc AS

```

```

function f_job_excep(nom jobs.job_title%type)
return jobs.job_id%type

```

```

is
v_id jobs.job_id%type;
begin
select job_id into v_id
from jobs where job_title=nom;
return v_id;
exception
when no_data_found then
return null;
when too_many_rows then
return null;
when others then
return -1;
end f_job_excep;

function f_job(nom jobs.job_title%type)
return jobs.job_id%type
is
begin
return
f_job_excep(pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(nom,true));
end f_job;

function f_empleat_per_cognom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type)
return employees%rowtype
is
v_emp employees%rowtype;
begin
v_emp:= f_empleat_per_cognom_i_nom(p_dept,p_cognom,null);
return v_emp;
end f_empleat_per_cognom;

function f_empleat_per_cognom_i_nom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type,
p_nom employees.first_name%type)
return employees%rowtype
is
v_emp employees%rowtype;
v_convertir boolean:=true;
begin
begin
select * into v_emp
from employees where department_id=p_dept

```

```

and
last_name=pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(p_cognom,v_con
vertir)
and
first_name=nvl(pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(p_nom,v_co
nvertir),first_name);
exception
when no_data_found then v_emp.employee_id:=-1;
when too_many_rows then v_emp.employee_id:=-99;
when others then v_emp.employee_id:=-10;
end;
return v_emp;
end f_empleat_per_cognom_i_nom;

END pkg_employees0523_Cachinero_Marc;

```

#####

tiene que devolver algo así?

Unsupported Command

Statement processed.

treballador: 110

treballador: 110

treballador: 148

treballador: -99

Unsupported Command

treballador: 110

treballador: 110

treballador: 148

treballador: -99

Statement processed.

job id és: AD_PRES

job id és: AD_PRES

job id és: AC_MGR

#####

#####

Peeegaloo aquuii

que pegue que? chetaose ha ido la foca

el código primero a

y scii pero si te lo pasé¿ktpioehju24ltjgubjrfgmdwppt provas

¿dnde esta marca en ropero de que ej??_¿?¿?¿?¿?¿? coño lo que has pasado tu y el**

sceipt el de la marina.¿* sa

3 ej 3

yo nada me paso eso el arque y eel script d pruebas me lo puedes cambiar? PERO

QUE ES IGUJALMKLÑE.NGJ+OE2ÑVJLKDNV JEFMBNMFJKLBFM

JOKLVMEFOKÑGEF2M+L`VFEL

ES IGUAL ES EL

MISMO SCRIPT

COÑO

QUE ES EL MISMO

SCRIPT

a no tengo que cambiar nada ?

no tengo que poner lo del apquete cambiado d nombre ni na?

Vale tt

chillea

entonces entrego lo del arque y el script de pruebas no?

SISISISISISISISISISIS

chetado

entonces entrego est noo?

David Bargados Gómez

script normal:

CREATE OR REPLACE PACKAGE pkg_ht_utilitats0523_Cachinero_Marc AS

function normalitzar_string

(p_string varchar2,p_convertir boolean)

return varchar2;

END pkg_ht_utilitats0523_Cachinero_Marc;

/

CREATE OR REPLACE PACKAGE BODY pkg_hr_utilitats0523_Cachinero_Marc AS

function normalitzar_string

(p_string varchar2,p_convertir boolean)

return varchar2

is

v_string varchar2(50);

v_string_out varchar2(50);

v_paraula varchar2(50);

v_paraula_aux varchar2(50);

v_pos number;

primera_vegada boolean;

-- agafa l'string d'entrada, elimina blancs a dreta i esquerra

-- i posa el primer caracter e majúscula i la resta en minúscula

begin

-- comprovar que no sigui null

if p_string is null then return null;

end if;

-- comprovar que la longitud no sigui més de 50

if length(trim(p_string))>50 then return null;

end if;

-- elimina blancs a dreta i esquerra

v_string:=trim(p_string);

-- agafar cada paraula i convertir-la

primera_vegada:=true;

while v_string is not null loop

-- busquem la posició del blanc, si no n'hi ha la posició és el final de l'string

v_pos:= instr(v_string,' ',1,1);

if v_pos=0 then v_pos:=length(v_string)+1; end if;

-- fem subtring per trobar la paraula

v_paraula:=substr(v_string,1,v_pos-1);

-- CONVERTIM la primera lletra a majúscula i la resta a minúscula

-- si l'opció convertir és true

if p_convertir then

if length(v_paraula)>1 then

v_paraula_aux:=upper(substr(v_paraula,1,1))||lower(substr(v_paraula,2));

else

v_paraula_aux:=upper(v_paraula);

```

end if;
else
v_paraula_aux:=v_paraula;
end if;

-- concatenem la paraula a l'string de sortida
if primera_vegada then
v_string_out:= v_paraula_aux;
primera_vegada:=false;
else
v_string_out:= v_string_out||' '|v_paraula_aux;
end if;

-- escurcem l'string per continuar buscant paraules a convertir,
-- aprofitem per eliminar blancs a l'esquerra
v_string:=ltrim(substr(v_string,v_pos+1));
end loop;

return v_string_out;
end normalitzar_string;
END pkg_hr_utilitats0523_Cachinero_Marc;
/

```

```

CREATE OR REPLACE PACKAGE pkg_employees0523_Cachinero_Marc AS

```

```

function f_job(nom jobs.job_title%type)
return jobs.job_id%type;

```

```

function f_empleat_per_cognom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type)
return employees%rowtype;

```

```

function f_empleat_per_cognom_i_nom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type,
p_nom employees.first_name%type)
return employees%rowtype;

```

```

END pkg_employees0523_Cachinero_Marc;
/

```

```

CREATE OR REPLACE PACKAGE BODY pkg_employees0523_Cachinero_Marc AS

```

```

function f_job_excep(nom jobs.job_title%type)
return jobs.job_id%type
is
v_id jobs.job_id%type;
begin

```

```

select job_id into v_id
from jobs where job_title=nom;
return v_id;
exception
when no_data_found then
return null;
when too_many_rows then
return null;
when others then
return -1;
end f_job_excep;

```

```

function f_job(nom jobs.job_title%type)
return jobs.job_id%type
is
begin
return
f_job_excep(pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(nom,true));
end f_job;

```

```

function f_empleat_per_cognom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type)
return employees%rowtype
is
v_emp employees%rowtype;
begin
v_emp:= f_empleat_per_cognom_i_nom(p_dept,p_cognom,null);
return v_emp;
end f_empleat_per_cognom;

```

```

function f_empleat_per_cognom_i_nom
(p_dept departments.department_id%type,
p_cognom employees.last_name%type,
p_nom employees.first_name%type)
return employees%rowtype
is
v_emp employees%rowtype;
v_convertir boolean:=true;
begin
begin
select * into v_emp
from employees where department_id=p_dept
and
last_name=pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(p_cognom,v_convertir)

```

```

and
first_name=nvl(pkg_hr_utilitats0523_Cachinero_Marc.normalitzar_string(p_nom,v_convertir),first_name);
exception
when no_data_found then v_emp.employee_id:=-1;
when too_many_rows then v_emp.employee_id:=-99;
when others then v_emp.employee_id:=-10;
end;
return v_emp;
end f_empleat_per_cognom_i_nom;

END pkg_employees0523_Cachinero_Marc;

```

SCRIPTPROVAS:

```

--SCRIPT DE PROVES f_empleat_per_cognom_i_nom
set SERVEROUTPUT on
declare
v_emp employees%rowtype;
begin

v_emp:=f_empleat_per_cognom_i_nom(100,'      CHEN','JOHN      ');
dbms_output.put_line('treballador: '||v_emp.employee_id);

v_emp:=f_empleat_per_cognom(100,'Chen      ');
dbms_output.put_line('treballador: '||v_emp.employee_id);

v_emp:=f_empleat_per_cognom_i_nom(80,'Cambrault','Gerald');
dbms_output.put_line('treballador: '||v_emp.employee_id);

v_emp:=f_empleat_per_cognom(80,'Cambrault');
dbms_output.put_line('treballador: '||v_emp.employee_id);
end;

/

```

--SCRIPT DE PROVES f_job

```

set SERVEROUTPUT on
declare
v_id jobs.job_id%type;
begin

v_id:=f_job('President');
dbms_output.put_line('job id és: '||v_id);

v_id:=f_job(' President ');
dbms_output.put_line('job id és: '||v_id);

```

```
v_id:=f_job(' accounting    manager');  
dbms_output.put_line('job id és: '||v_id);  
end;
```