

TP paquetage

Compte-rendu de bd

BOURG / LAYADI
26/09/2016

Package:

create or replace package pksorties

is

function les_minibus return number;

function affectations_minibus(numero sortie.nosortie%type) return number;

end pksorties;

Corps du package:

create or replace PACKAGE BODY PKSORTIES AS

type type_tabminibus is table of

minibus%rowtype index by

binary_integer;

v_tabminibus type_tabminibus;

function les_minibus return number AS

 v_nbminibus number(2) := 0;

 v_message varchar2(255);

 v_sqlcode number;

BEGIN

 for enr_minibus in (select * from minibus order by capacite desc)

 loop

 v_nbminibus := v_nbminibus + 1;

 v_tabminibus(v_nbminibus).nominibus := enr_minibus.nominibus;

 v_tabminibus(v_nbminibus).capacite := enr_minibus.capacite;

 end loop;

 RETURN v_nbminibus;

EXCEPTION

```

when others then
    v_message := SQLERRM;
    v_sqlcode := SQLCODE;
    insert into erreurs values (v_message || v_sqlcode || TO_CHAR(SYSDATE, 'YYYY-MM-DD
HH24:MI:SS'));
    return -1;
END les_minibus;

```

function affectations_minibus(numero sortie.nosortie%type) return number AS

```

v_message varchar2(255);
v_sqlcode number;
v_denomMinibus number(2);
v_nombMinibus number(2) := 1;
cursor myCursor is select noadherent from inscription_sortie where nosortie = numero order
by date_inscription FOR UPDATE OF nominibus;
strix myCursor%rowtype;

BEGIN
v_denomMinibus := les_minibus;

OPEN myCursor;
FETCH myCursor into strix;
while myCursor%found and v_nombMinibus <= v_denomMinibus
LOOP
    IF v_tabminibus(v_nombMinibus).capacite = 0 THEN
        v_nombMinibus := v_nombMinibus +1;
    END IF;

```

```
IF v_denomMinibus > 0 THEN
    UPDATE inscription_sortie set nominibus = v_tabminibus(v_nombMinibus).nominibus
    where current of myCursor;
    END IF;

    v_tabminibus(v_nombMinibus).capacite := v_tabminibus(v_nombMinibus).capacite - 1;
    FETCH myCursor into strix;
END LOOP;
CLOSE myCursor;

commit;
return v_nombMinibus;

EXCEPTION
    when others then
        v_message := SQLERRM;
        v_sqlcode := SQLCODE;
        rollback;
        insert into erreurs values (v_message || v_sqlcode || TO_CHAR(SYSDATE, 'YYYY-MM-DD
        HH24:MI:SS'));
        return -1;

END affectations_minibus;

END PKSORTIES;
```

Exécution:

```
variable nb_minibus number;
exec :nb_minibus :=affectations_minibus(2);
print nb_minibus;
```

Résultat:

	NOADHERENT	NOSORTIE	DATE_INSCRIPTION	NOMINIBUS
1	1	1	10/08/14	(null)
2	3	1	09/08/14	(null)
3	5	1	15/08/14	(null)
4	7	1	01/08/14	(null)
5	8	1	11/08/14	(null)
6	10	1	16/08/14	(null)
7	11	1	04/08/14	(null)
8	12	1	10/08/14	(null)
9	13	1	10/08/14	(null)
10	14	1	05/08/14	(null)
11	15	1	20/08/14	(null)
12	16	1	20/08/14	(null)
13	4	1	21/08/14	(null)
14	6	1	01/08/14	(null)
15	9	1	14/08/14	(null)
16	2	2	10/08/14	2
17	4	2	15/08/14	2
18	10	2	18/08/14	2
19	6	2	16/08/14	2
20	8	2	14/08/14	2