

Recorrer con un cursor una tabla de objetos

Ejemplo:

Se crea el Object tipe.....

```
SQL> CREATE OR REPLACE TYPE apartment AS OBJECT (
2     BldgName          VARCHAR2(40),
3     BldgAddress       address,
4     BldgMgr           INTEGER,
5     MEMBER PROCEDURE  ChangeMgr (NewMgr IN INTEGER),
6     ORDER MEMBER FUNCTION Compare (Otherapartment IN apartment)
7         RETURN INTEGER
8 );
9 /
```

Se crea la tabla para almacenar objetos.....

```
SQL> CREATE TABLE apartments OF apartment;
```

Insertamos varios apartamentos en la tabla....

```
SQL> INSERT INTO apartments values (apartment('Headquarters
apartment',address('1 J', '', 'Detroit', 'MI', '48226', ''),599));
```

Recorremos con un cursor la tabla de objetos

```
SQL> DECLARE
2     this_apartment    apartment;
3
4     CURSOR all_apartments IS
5         SELECT value (b) AS bldg
6             FROM apartments b
7             ORDER BY b.BldgName;
8
9     BEGIN
10        FOR one_apartment IN all_apartments LOOP
11            this_apartment := one_apartment.bldg;
12            dbms_output.put_line(this_apartment.BldgName || ' is located
in '
13                                || this_apartment.BldgAddress.city
14                                || ' ' ||
this_apartment.BldgAddress.state_abbr);
15        END LOOP;
16
17        COMMIT;
18    END;
19 /
```

Creación de secuencias (Sequence)

Sintaxis:

```
CREATE SEQUENCE secuencia
[INCREMENT BY n]
[START WITH n]
[{MAXVALUE n|NOMAXVALUE}]
[{MINVALUE n|NOMINVALUE}]
[{CYCLE|NOCYCLE}]
```

Donde:

- *secuencia*. Es el nombre que se le da al objeto de secuencia
- **INCREMENT BY**. Indica cuánto se incrementa la secuencia cada vez que se usa. Por defecto se incrementa de uno en uno
- **START WITH**. Indica el valor inicial de la secuencia (por defecto 1)
- **MAXVALUE**. Máximo valor que puede tomar la secuencia. Si no se toma NOMAXVALUE que permite llegar hasta el 1027
- **MINVALUE**. Mínimo valor que puede tomar la secuencia. Por defecto -1026

```
CREATE SEQUENCE numeroPlanta
INCREMENT 100
STARTS WITH 100
MAXVALUE 2000
```

Insertar directamente el valor obtenido de una secuencia en una tabla

```
INSERT INTO plantas(num, uso)
VALUES(numeroPlanta.NEXTVAL, 'Suites');
```

Insertar directamente el valor obtenido de una secuencia en una variable de PL/SQL

```
SELECT numeroPlanta.NEXTVAL INTO num_planta FROM dual;
```