No siempre necesitas poner FOR UPDATE en un cursor cuando haces un UPDATE, pero hay situaciones en las que sí es necesario. Aquí te explico:

**🔹 Cursores implícitos**

* En SQL y PL/SQL, cuando haces un UPDATE, INSERT, DELETE o SELECT INTO, **se usa un cursor implícito automáticamente**, por lo que **no necesitas declarar uno explícito ni usar FOR UPDATE**.
* Ejemplo:
* UPDATE empleados SET salario = salario \* 1.1 WHERE departamento\_id = 10;

Aquí, Oracle maneja todo internamente sin que tengas que preocuparte por cursores.

**🔹 Cursores explícitos sin FOR UPDATE**

* Se usan cuando quieres recorrer un conjunto de filas **sin la intención de modificarlas directamente** dentro del cursor.
* Ejemplo:
* DECLARE CURSOR c\_empleados IS
* SELECT nombre, salario FROM empleados WHERE departamento\_id = 10;
  + Esto simplemente recupera datos sin bloquear ni preparar las filas para actualización.

**🔹 Cursores explícitos con FOR UPDATE**

* Se usa cuando necesitas **recorrer filas y modificarlas dentro del cursor**.
* **Bloquea** las filas seleccionadas para evitar que otros procesos las modifiquen mientras trabajas con ellas.
* Ejemplo:
* DECLARE CURSOR c\_empleados IS
* SELECT id\_empleado, salario FROM empleados WHERE departamento\_id = 10 FOR UPDATE;

Luego, al recorrer el cursor:

BEGIN

FOR empleado IN c\_empleados LOOP

UPDATE empleados

SET salario = empleado.salario \* 1.1

WHERE CURRENT OF c\_empleados;

END LOOP;

END;

🔹 **Importante:**

* + El WHERE CURRENT OF c\_empleados actualiza **solo la fila actual del cursor**, en lugar de usar WHERE id\_empleado = empleado.id\_empleado.
  + FOR UPDATE **bloquea las filas** hasta que hagas un COMMIT o ROLLBACK.

**🔹 Casos donde FOR UPDATE es necesario o recomendable**

1. **Evitar modificaciones concurrentes:**
   * Si varios procesos pueden modificar las mismas filas, FOR UPDATE asegura que no haya cambios inesperados mientras recorres el cursor.
2. **Actualizar filas dentro del cursor sin necesidad de una clave primaria explícita:**
   * Usando WHERE CURRENT OF, evitas escribir WHERE id = ..., lo que simplifica el código.
3. **Evitar inconsistencias en lecturas y modificaciones:**
   * FOR UPDATE bloquea las filas **solo hasta que hagas COMMIT**, evitando que otro usuario las cambie mientras tú las procesas.

🔹 **¿Cuándo NO usar FOR UPDATE?**

* Si solo necesitas leer los datos, porque bloquear filas innecesariamente puede afectar el rendimiento.
* Si vas a hacer el UPDATE de una sola vez sin recorrer el cursor.

Espero que esto te aclare la duda. ¿Quieres que te haga más ejemplos? 😊