

== Trigger ==

تعريفه : هو نوع خاص من الإجراءات المخزنة (stored procedure) التي تشتغل تلقائي وقت ما يصير عندي حدث (event) في خادم قاعدة البيانات.

الهيكل الأساسي:

```
CREATE TRIGGER trigger_name
{BEFORE | AFTER | INSTEAD OF}
{INSERT | UPDATE | DELETE}
ON table_name
[FOR EACH ROW]
[WHEN (condition)]
BEGIN
    -- Trigger امر تنفيذ ال
    -- يمكن أن يشمل إجراءات مثل تحديث جداول أخرى أو إدراج بيانات
END;
```

أنواعه:

1. DML Trigger : قائم على كل من INSERT و UPDATE و DELETE و

يقسم على جزئين:

-1 Before event : نقدر نستخدمها مثلاً في اني اقدر اني اقولك انه مو مسموح

لتعديل معين مثلاً أو اني مثلاً اغير قيم قبل حصول الحدث.

-2 After event : نستخدمه في حاله اني احتاج اسوي اجراء معين بعد ما يتم الحدث

بنجاح مثل : اذا تمت عملية شراء أو دفع يتم تحديث رصيد المستخدم

3- تلقائياً.

```
CREATE OR REPLACE TRIGGER log_dml_operations
AFTER INSERT OR UPDATE OR DELETE
ON employees
FOR EACH ROW
BEGIN
    INSERT INTO log_table (
        operation_type,
        table_name,
        affected_row_id,
        operation_time
    )
    VALUES (
        CASE
            WHEN INSERTING THEN 'INSERT'
            WHEN UPDATING THEN 'UPDATE'
            WHEN DELETING THEN 'DELETE'
        END,
        'employees', -- اسم الجدول الذي يتم التعديل عليه
        NVL(:OLD.emp_id, :NEW.emp_id), -- معرف الصف المتأثر
        SYSDATE -- وقت العملية
    );
END;
/
```

عند كل عملية إدراج، تعديل، أو حذف على جدول employees، يتم تسجيل العملية في جدول log_table مع نوع العملية (INSERT, UPDATE, DELETE) و اسم الجدول (employees) و معرف الصف المتأثر وقت العملية.

2. **DDL Trigger**: يُستخدم لضمان سلامة البيانات، حماية الجداول، والتأكد من عدم وجود تغييرات غير مصرح بها قائم على كلا من CREATE, DROP AND ALTER.

نقدر نستخدمه مثلاً في اني اقدر امنع احد انه يسوي DROP لجدول معين أو تسجيل أي عملية من العمليات السابقة مع اسم المنفذ لها.

```
CREATE OR REPLACE TRIGGER ddl_table_monitor
AFTER CREATE OR ALTER OR DROP
ON DATABASE
BEGIN
    INSERT INTO ddl_audit_log (
        event_type,
        object_name,
        operation_date,
        user_name
    )
    VALUES (
        ORA_SYSEVENT, -- نوع العملية (CREATE, ALTER, DROP)
        ORA_DICT_OBJ_NAME, -- اسم الكائن (الجدول، العرض، إلخ)
        SYSDATE, -- تاريخ ووقت العملية
        USER -- اسم المستخدم الذي قام بالعملية
    );

    DBMS_OUTPUT.PUT_LINE('DDL Operation Logged: ' || ORA_SYSEVENT || ' on ' || ORA_DICT_OBJ_NAME);
END;
```

المثال الموضح يسجل الوقت الحالي عند تنفيذ عملية CREATE أو DROP.

3. **Log on Trigger**: ينفذ وقت ما المستخدم يسجل دخول لقاعدة البيانات. نقدر نستفيد من توظيفه في الوقت الي المستخدم يسجل دخول، مثل تهيئة الجلسة أو التحقق من الحقوق أو تسجيل الأنشطة.

```
CREATE OR REPLACE TRIGGER logon_audit_trigger
AFTER LOGON
ON DATABASE
BEGIN
    INSERT INTO logon_audit (
        username,
        userhost,
        logon_time
    )
    VALUES (
        USER, -- اسم المستخدم الذي سجل الدخول
        SYS_CONTEXT('USERENV', 'HOST'), -- اسم الجهاز أو المضيف
        SYSDATE -- وقت تسجيل الدخول
    );
END;
```

عند تسجيل الدخول من أي مستخدم، يضاف سجل جديد في جدول logon_audit مكون من اسم المستخدم و اسم الجهاز أو المضيف و وقت تسجيل الدخول.