NSHURO ARNAUD NELLIGAN

27960

PHASE 7

Objective:

Implement business rules, auditing, and advanced database features using PL/SQL triggers and functions to enforce security restrictions and record violations.
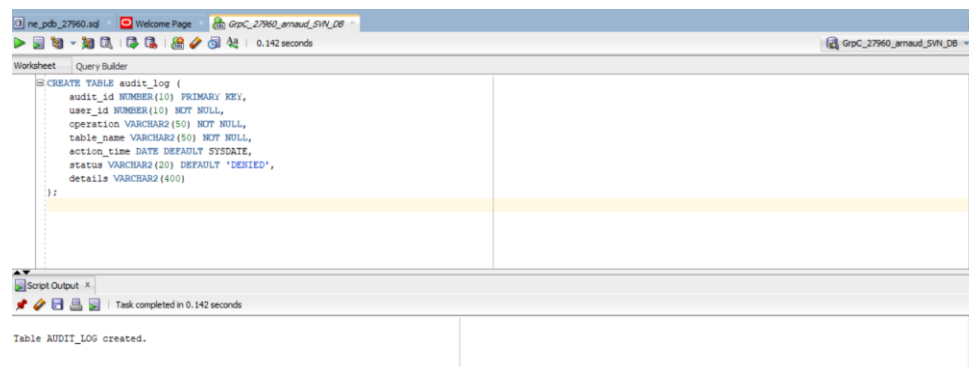
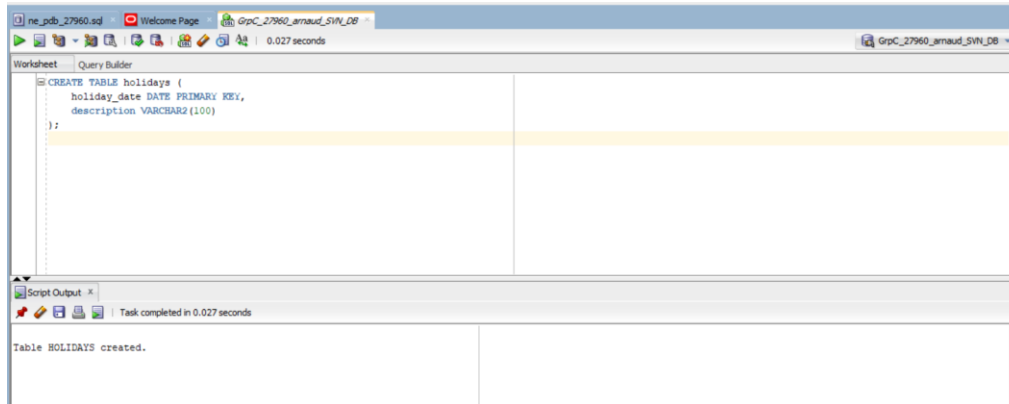**1. Business Rule**

Critical Rule:

- Employees **cannot INSERT/UPDATE/DELETE** on:

    o **Weekdays (Monday–Friday)**

    o **Public holidays (upcoming month only)**

- All attempted actions must be logged in an **audit table**.
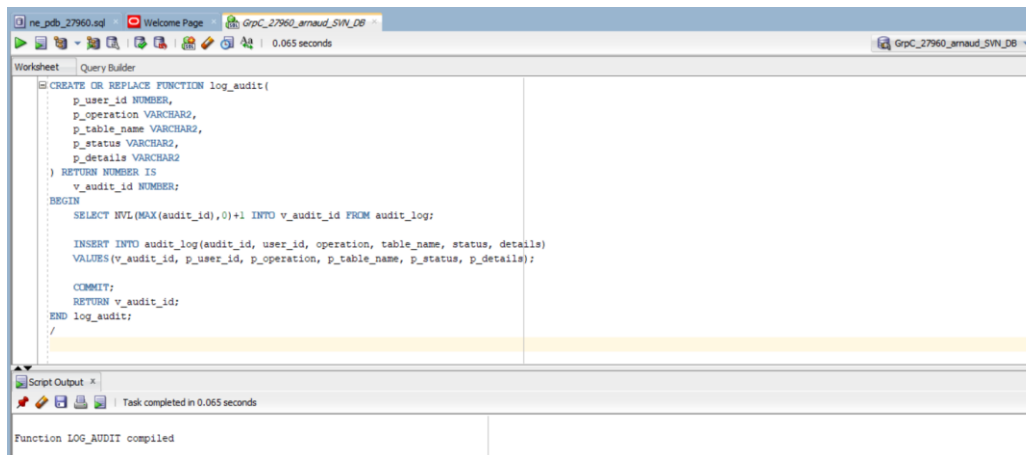
2. Tables for Auditing

Audit Log Table

Holiday Table

Table HOLIDAYS created.

3. Audit Logging Function


```
CREATE OR REPLACE FUNCTION log_audit(
    p_user_id NUMBER,
    p_operation VARCHAR2,
    p_table_name VARCHAR2,
    p_status VARCHAR2,
    p_details VARCHAR2
) RETURN NUMBER IS
    v_audit_id NUMBER;
BEGIN
    SELECT NVL(MAX(audit_id),0)+1 INTO v_audit_id FROM audit_log;

    INSERT INTO audit_log(audit_id, user_id, operation, table_name, status, details)
    VALUES(v_audit_id, p_user_id, p_operation, p_table_name, p_status, p_details);

    COMMIT;
    RETURN v_audit_id;
END log_audit;
/
```

Function LOG_AUDIT compiled

4. Restriction Check Function


```
CREATE OR REPLACE FUNCTION check_restriction RETURN BOOLEAN IS
    v_day VARCHAR2(10);
    v_count NUMBER;
BEGIN
    -- Get current day abbreviation
    SELECT TO_CHAR(SYSDATE,'DY','NLS_DATE_LANGUAGE=ENGLISH') INTO v_day FROM dual;

    -- Weekday restriction
    IF v_day IN ('MON','TUE','WED','THU','FRI') THEN
        RETURN FALSE;
    END IF;

    -- Public holiday restriction
    SELECT COUNT(*) INTO v_count
    FROM holidays
    WHERE holiday_date = TRUNC(SYSDATE);

    IF v_count > 0 THEN
        RETURN FALSE;
    END IF;

    -- Allowed
    RETURN TRUE;
END check_restriction;
/
```

Function CHECK_RESTRICTION compiled