Name: Vishal R

Super set: 6413500

**Scenario – 1:**

SET SERVEROUTPUT ON;

BEGIN

    EXECUTE IMMEDIATE 'DROP PROCEDURE ProcessMonthlyInterest';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -4043 THEN RAISE; END IF;

END;

/

BEGIN

    EXECUTE IMMEDIATE 'DROP TABLE ACCOUNTS';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -942 THEN RAISE; END IF;

END;

/

CREATE TABLE ACCOUNTS (

    ACCOUNT\_ID     NUMBER PRIMARY KEY,

    CUSTOMER\_NAME  VARCHAR2(100),

    ACCOUNT\_TYPE   VARCHAR2(20),

    BALANCE        NUMBER(12,2)

);

/

BEGIN

    INSERT INTO ACCOUNTS VALUES (1, 'John Smith', 'SAVINGS', 10000.00);

    INSERT INTO ACCOUNTS VALUES (2, 'Alice Johnson', 'CURRENT', 20000.00);

    INSERT INTO ACCOUNTS VALUES (3, 'Robert Lee', 'SAVINGS', 5000.00);

    INSERT INTO ACCOUNTS VALUES (4, 'Emily Clark', 'SAVINGS', 15000.00);

    COMMIT;

END;

/

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

    UPDATE ACCOUNTS

    SET BALANCE = BALANCE + (BALANCE \* 0.01)

    WHERE ACCOUNT\_TYPE = 'SAVINGS';

    DBMS\_OUTPUT.PUT\_LINE('Monthly interest applied to all savings accounts.');

END;

/

BEGIN

    ProcessMonthlyInterest;

END;

/

BEGIN

    FOR acc IN (SELECT \* FROM ACCOUNTS) LOOP

        DBMS\_OUTPUT.PUT\_LINE('ID: ' || acc.ACCOUNT\_ID ||

                             ', Name: ' || acc.CUSTOMER\_NAME ||

                             ', Type: ' || acc.ACCOUNT\_TYPE ||

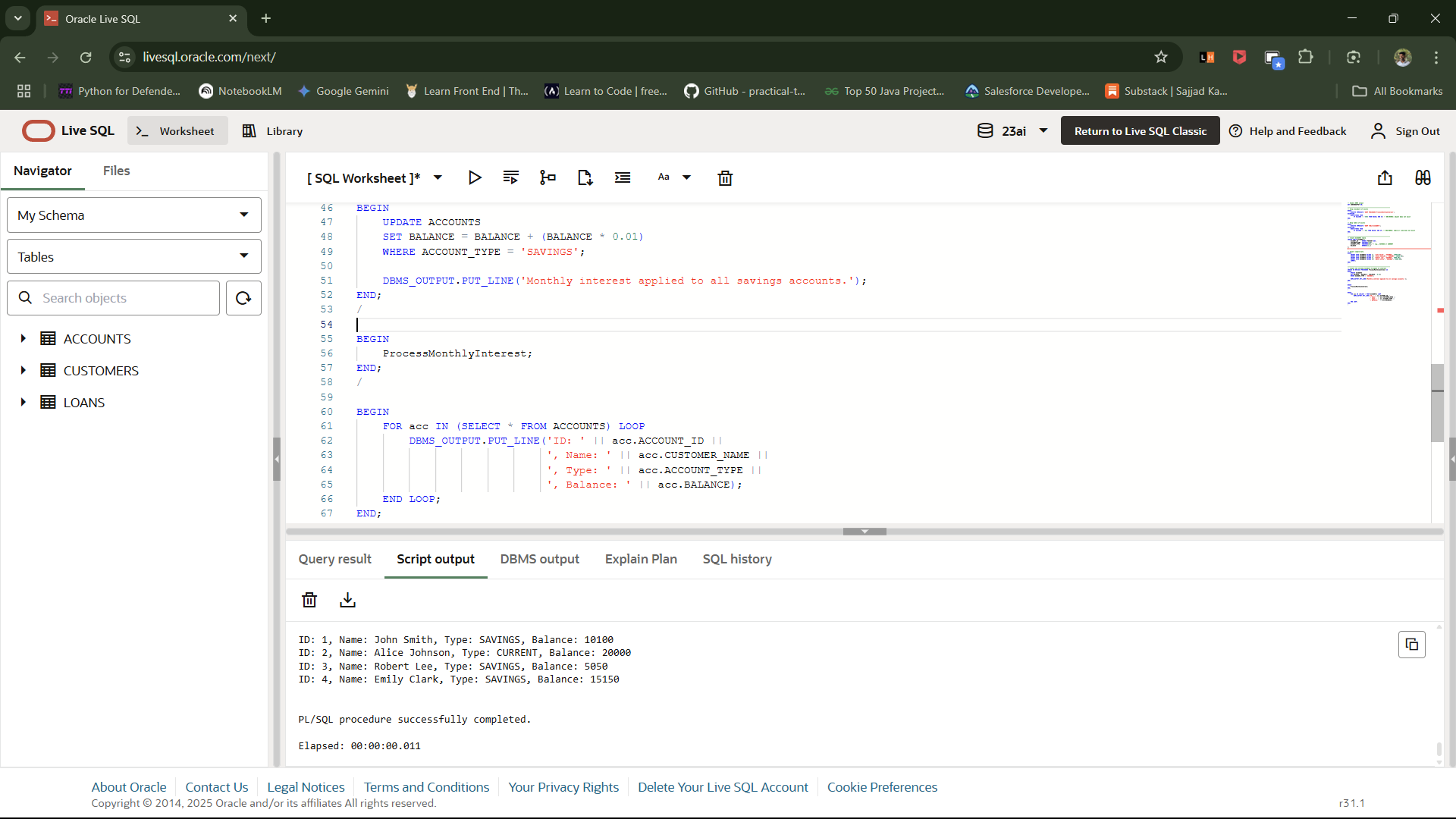
                             ', Balance: ' || acc.BALANCE);

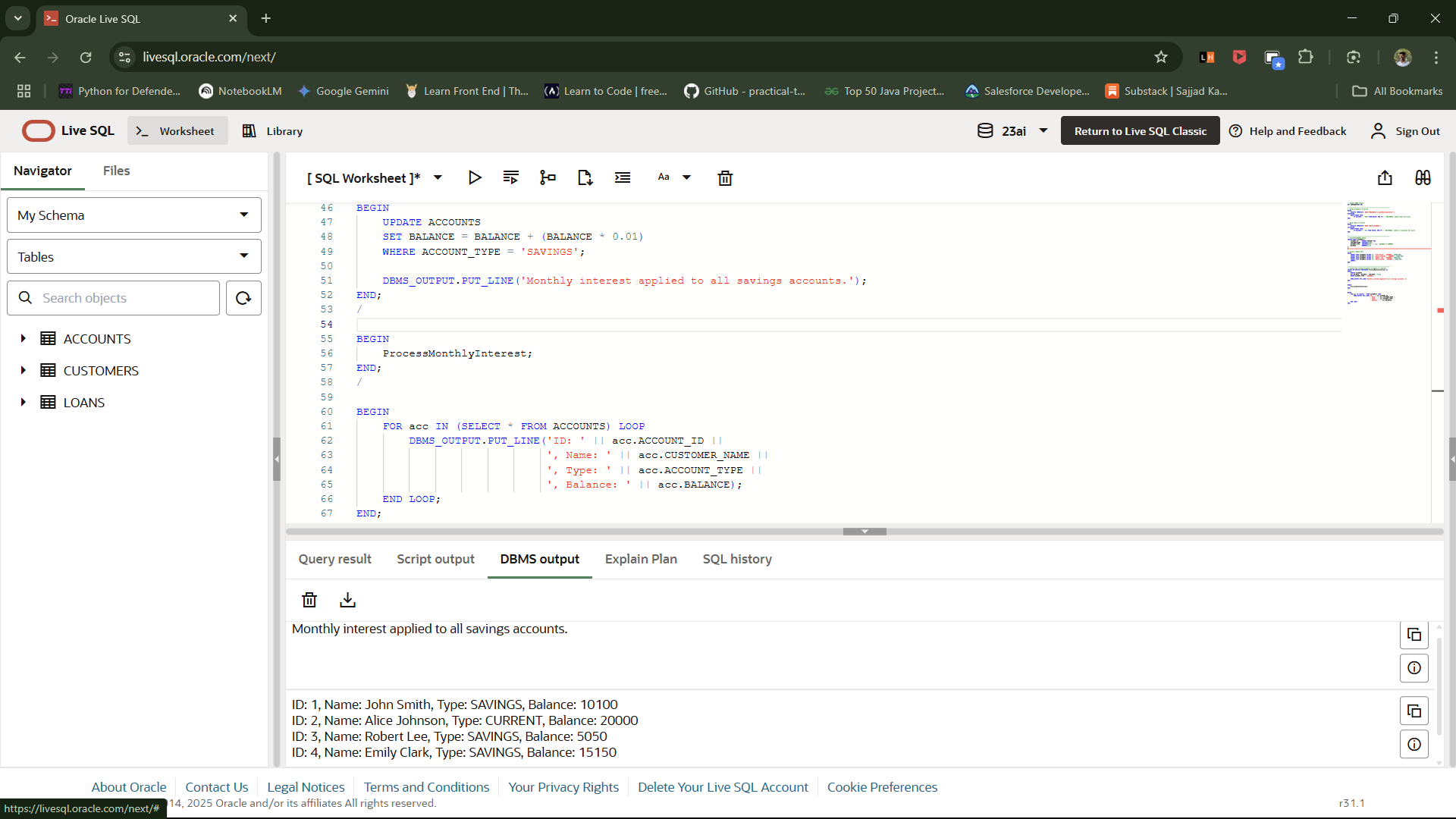
    END LOOP;

END;

/

Output:  
  
**Scenario – 1:**





Scenario – 2:

SET SERVEROUTPUT ON;

BEGIN

    EXECUTE IMMEDIATE 'DROP PROCEDURE UpdateEmployeeBonus';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -4043 THEN RAISE; END IF;

END;

/

BEGIN

    EXECUTE IMMEDIATE 'DROP TABLE EMPLOYEES';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -942 THEN RAISE; END IF;

END;

/

CREATE TABLE EMPLOYEES (

    EMP\_ID        NUMBER PRIMARY KEY,

    NAME          VARCHAR2(100),

    DEPARTMENT    VARCHAR2(50),

    SALARY        NUMBER(10,2)

);

BEGIN

    INSERT INTO EMPLOYEES VALUES (1, 'Alice', 'SALES', 50000);

    INSERT INTO EMPLOYEES VALUES (2, 'Bob', 'HR', 45000);

    INSERT INTO EMPLOYEES VALUES (3, 'Charlie', 'SALES', 55000);

    INSERT INTO EMPLOYEES VALUES (4, 'Diana', 'IT', 60000);

    INSERT INTO EMPLOYEES VALUES (5, 'Ethan', 'SALES', 52000);

    COMMIT;

END;

/

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

    dept\_name IN VARCHAR2,

    bonus\_pct IN NUMBER

) IS

BEGIN

    UPDATE EMPLOYEES

    SET SALARY = SALARY + (SALARY \* bonus\_pct / 100)

    WHERE UPPER(DEPARTMENT) = UPPER(dept\_name);

    DBMS\_OUTPUT.PUT\_LINE('Bonus of ' || bonus\_pct || '% applied to department: ' || dept\_name);

END;

/

BEGIN

    UpdateEmployeeBonus('SALES', 10);

END;

/

BEGIN

    FOR emp IN (SELECT \* FROM EMPLOYEES) LOOP

        DBMS\_OUTPUT.PUT\_LINE('ID: ' || emp.EMP\_ID ||

                             ', Name: ' || emp.NAME ||

                             ', Dept: ' || emp.DEPARTMENT ||

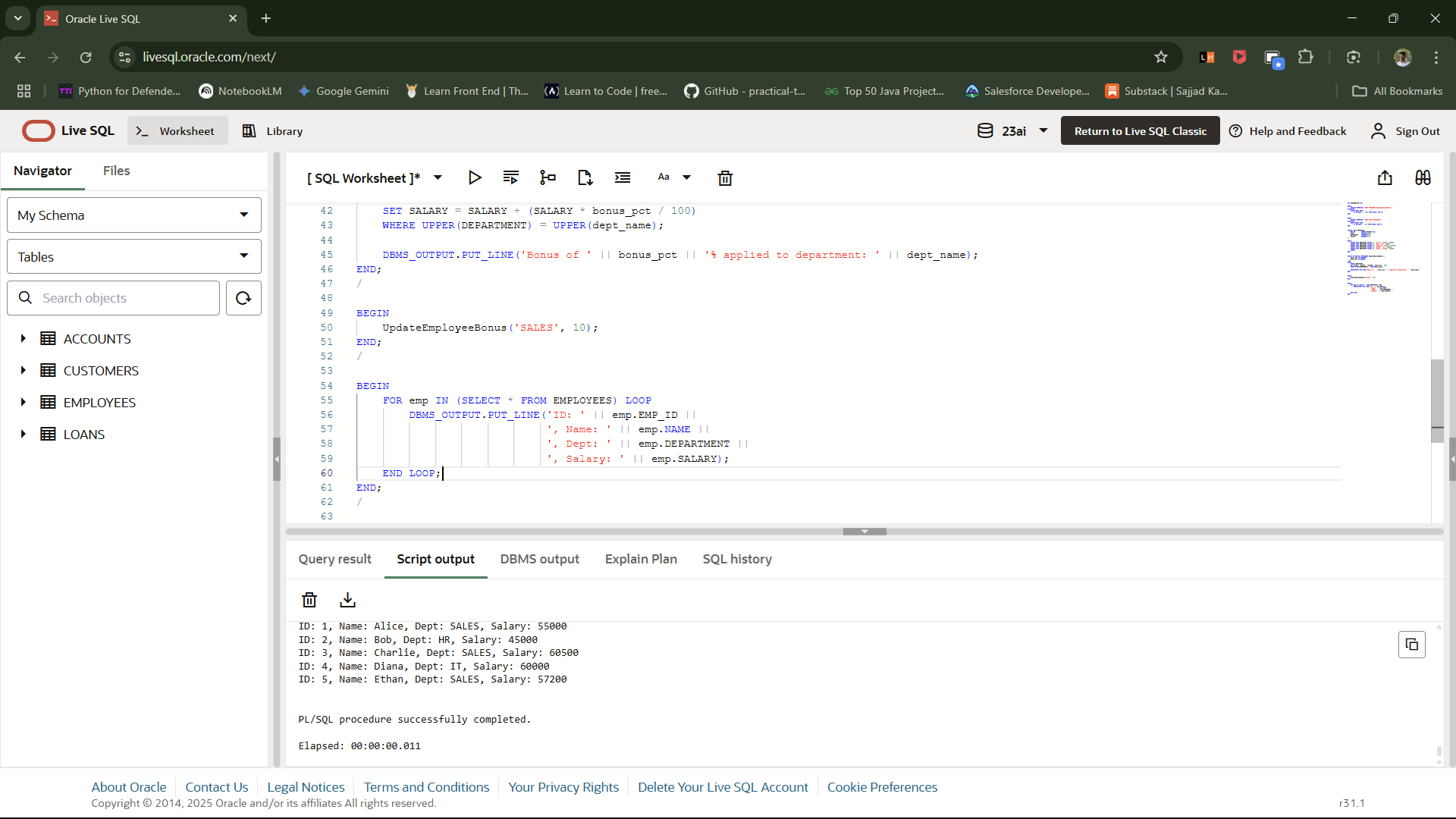
                             ', Salary: ' || emp.SALARY);

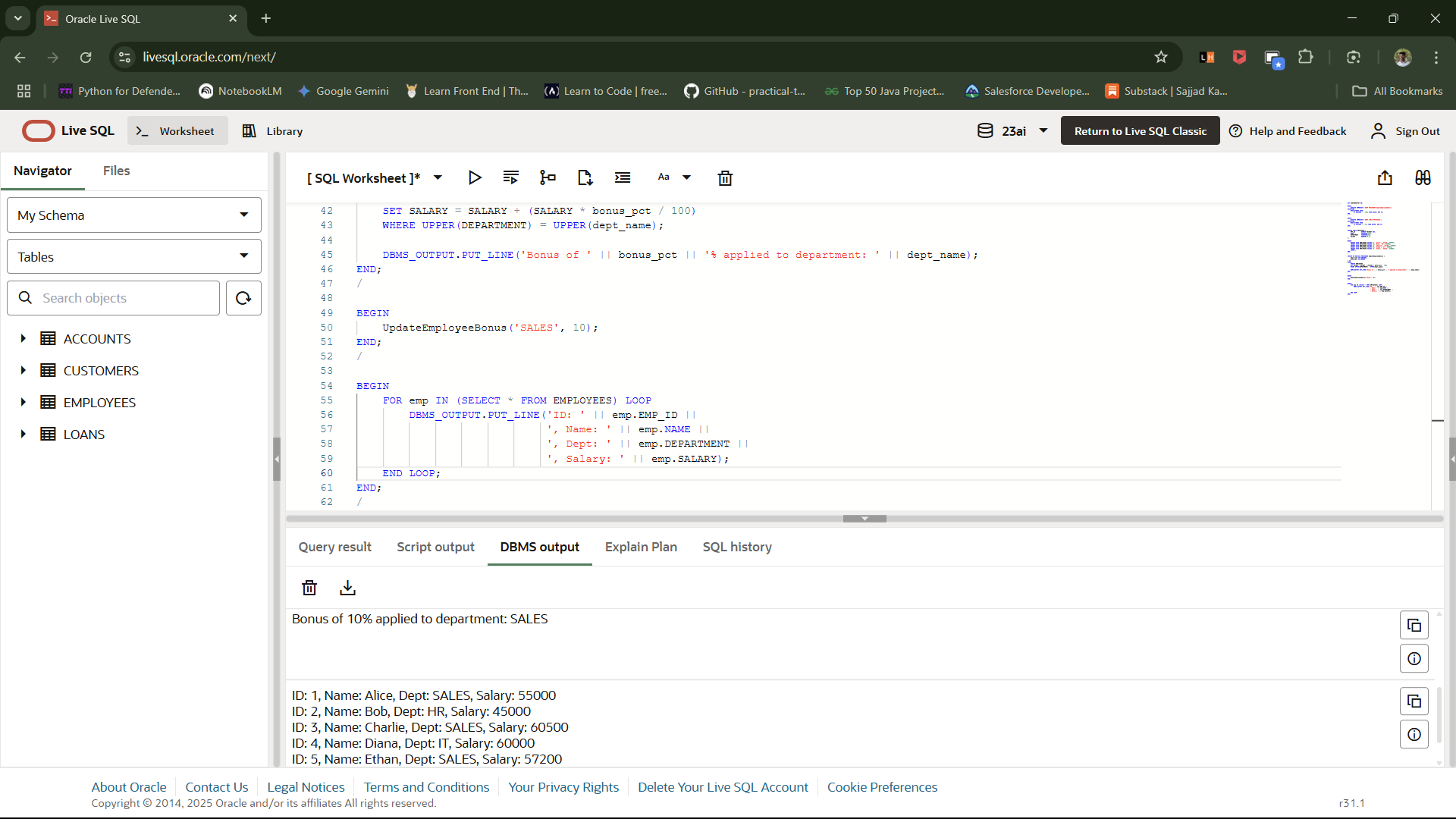
    END LOOP;

END;

/

OUTPUT:





Scenario – 3:

SET SERVEROUTPUT ON;

BEGIN

    EXECUTE IMMEDIATE 'DROP PROCEDURE TransferFunds';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -4043 THEN RAISE; END IF;

END;

/

BEGIN

    EXECUTE IMMEDIATE 'DROP TABLE ACCOUNTS';

EXCEPTION

    WHEN OTHERS THEN

        IF SQLCODE != -942 THEN RAISE; END IF;

END;

/

CREATE TABLE ACCOUNTS (

    ACCOUNT\_ID     NUMBER PRIMARY KEY,

    CUSTOMER\_NAME  VARCHAR2(100),

    BALANCE        NUMBER(12,2)

);

BEGIN

    INSERT INTO ACCOUNTS VALUES (1, 'John Smith', 10000.00);

    INSERT INTO ACCOUNTS VALUES (2, 'Alice Johnson', 5000.00);

    COMMIT;

END;

/

CREATE OR REPLACE PROCEDURE TransferFunds (

    p\_from\_account   IN NUMBER,

    p\_to\_account     IN NUMBER,

    p\_amount         IN NUMBER

) IS

    v\_from\_balance NUMBER;

BEGIN

    SELECT BALANCE INTO v\_from\_balance

    FROM ACCOUNTS

    WHERE ACCOUNT\_ID = p\_from\_account;

    IF v\_from\_balance < p\_amount THEN

        RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient funds in source account.');

    END IF;

    UPDATE ACCOUNTS

    SET BALANCE = BALANCE - p\_amount

    WHERE ACCOUNT\_ID = p\_from\_account;

    UPDATE ACCOUNTS

    SET BALANCE = BALANCE + p\_amount

    WHERE ACCOUNT\_ID = p\_to\_account;

    DBMS\_OUTPUT.PUT\_LINE('Transfer of $' || p\_amount || ' from Account ' || p\_from\_account || ' to Account ' || p\_to\_account || ' successful.');

    COMMIT;

EXCEPTION

    WHEN NO\_DATA\_FOUND THEN

        DBMS\_OUTPUT.PUT\_LINE('Error: One of the accounts does not exist.');

    WHEN OTHERS THEN

        DBMS\_OUTPUT.PUT\_LINE('Transfer failed: ' || SQLERRM);

        ROLLBACK;

END;

/

BEGIN

    TransferFunds(1, 2, 2000);

END;

/

BEGIN

    FOR acc IN (SELECT \* FROM ACCOUNTS ORDER BY ACCOUNT\_ID) LOOP

        DBMS\_OUTPUT.PUT\_LINE('Account ID: ' || acc.ACCOUNT\_ID ||

                             ', Name: ' || acc.CUSTOMER\_NAME ||

                             ', Balance: ' || acc.BALANCE);

    END LOOP;

END;

/

OUTPUT:

