# PL/SQL Mandatory Hands On Exercise 1

Question:

Exercise 1: Control Structures

Scenario 1: The bank wants to apply a discount to loan interest rates for customers above 60 years old.

o Question: Write a PL/SQL block that loops through all customers, checks their age, and if they are above 60, apply a 1% discount to their current loan interest rates.

Scenario 2: A customer can be promoted to VIP status based on their balance.

o Question: Write a PL/SQL block that iterates through all customers and sets a flag IsVIP to TRUE for those with a balance over $10,000.

Scenario 3: The bank wants to send reminders to customers whose loans are due within the next 30 days.

o Question: Write a PL/SQL block that fetches all loans due in the next 30 days and prints a reminder message for each customer.

Code:

**-- Create customers table**

**CREATE TABLE customers (**

**customer\_id NUMBER PRIMARY KEY,**

**name VARCHAR2(30),**

**age NUMBER,**

**balance NUMBER(12, 2),**

**interest\_rate NUMBER(5, 2),**

**isVIP VARCHAR2(5) DEFAULT 'FALSE'**

**);**

**-- Create loans table**

**CREATE TABLE loans (**

**loan\_id NUMBER PRIMARY KEY,**

**customer\_id NUMBER,**

**loan\_amount NUMBER(12, 2),**

**due\_date DATE,**

**FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)**

**);**

**set linesize 150;**

**INSERT INTO customers VALUES (1, 'Alice', 65, 12000.00, 6.5, 'FALSE');**

**INSERT INTO customers VALUES (2, 'Bob', 45, 8000.00, 5.5, 'FALSE');**

**INSERT INTO customers VALUES (3, 'Charlie', 70, 15000.00, 7.0, 'FALSE');**

**INSERT INTO customers VALUES (4, 'Diana', 61, 4000.00, 5.0, 'FALSE');**

**INSERT INTO customers VALUES (5, 'Eve', 30, 10500.00, 6.2, 'FALSE');**

**INSERT INTO customers VALUES (6, 'Frank', 55, 6000.00, 6.8, 'FALSE');**

**INSERT INTO customers VALUES (7, 'Grace', 62, 3000.00, 5.3, 'FALSE');**

**INSERT INTO customers VALUES (8, 'Hank', 28, 9500.00, 6.0, 'FALSE');**

**INSERT INTO customers VALUES (9, 'Ivy', 67, 13000.00, 7.1, 'FALSE');**

**INSERT INTO customers VALUES (10, 'Jack', 40, 11000.00, 5.9, 'FALSE');**

**INSERT INTO loans VALUES (101, 1, 5000.00, SYSDATE + 15);**

**INSERT INTO loans VALUES (102, 2, 3000.00, SYSDATE + 45);**

**INSERT INTO loans VALUES (103, 3, 7000.00, SYSDATE + 10);**

**INSERT INTO loans VALUES (104, 4, 4000.00, SYSDATE + 25);**

**INSERT INTO loans VALUES (105, 5, 2000.00, SYSDATE + 5);**

**INSERT INTO loans VALUES (106, 6, 3500.00, SYSDATE + 35);**

**INSERT INTO loans VALUES (107, 7, 2500.00, SYSDATE + 28);**

**INSERT INTO loans VALUES (108, 8, 1500.00, SYSDATE + 50);**

**INSERT INTO loans VALUES (109, 9, 8000.00, SYSDATE + 18);**

**INSERT INTO loans VALUES (110, 10, 4500.00, SYSDATE + 3);**

**select \* from customers;**

**select \* from loans;**

**commit;**

**BEGIN**

**FOR rec IN (SELECT customer\_id, interest\_rate, age**

**FROM customers**

**WHERE age > 60)**

**LOOP**

**UPDATE customers**

**SET interest\_rate = interest\_rate - 1**

**WHERE customer\_id = rec.customer\_id;**

**END LOOP;**

**COMMIT;**

**END;**

**/**

**select \* from customers;**

**commit;**

**BEGIN**

**FOR rec IN (SELECT customer\_id, balance**

**FROM customers**

**WHERE balance > 10000)**

**LOOP**

**UPDATE customers**

**SET isVIP = 'TRUE'**

**WHERE customer\_id = rec.customer\_id;**

**END LOOP;**

**COMMIT;**

**END;**

**/**

**select \* from customers;**

**commit;**

**DECLARE**

**v\_due\_date loans.due\_date%TYPE;**

**v\_customer\_id loans.customer\_id%TYPE;**

**BEGIN**

**FOR rec IN (**

**SELECT customer\_id, due\_date**

**FROM loans**

**WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30**

**)**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan for Customer ID ' || rec.customer\_id ||**

**' is due on ' || TO\_CHAR(rec.due\_date, 'DD-MON-YYYY'));**

**END LOOP;**

**END;**

**/**

Output:



