**WEEK-02 HANDS ON SOLUTIONS**

**PL/SQL EXERCISE SOLUTIONS**

**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.

**EXPLANATION**

This PL/SQL program is designed to demonstrate the use of control structures in Oracle by simulating a real-world banking scenario.

Two tables are created: customers and loans.

The customers table holds details about each customer, such as their customer\_id, name, age, balance, and a flag isvip which denotes if they are considered a VIP.

The loans table stores loan information including loan\_id, customer\_id (foreign key), interest\_rate, and due\_date.

These two tables are linked via a foreign key relationship to represent which customer owns which loan.

The script proceeds with three main procedures (PL/SQL blocks), each addressing a specific business rule using control structures.

In the first scenario, a FOR loop and an IF condition are used to iterate through customers and apply a 1% interest rate discount to all customers over the age of 60.

In the second block, the script directly updates the isvip status of all customers with balances exceeding $10,000 by setting the flag to 'TRUE'.

This step is useful for classifying high-value customers. The third block is a reporting operation that finds all loans due within the next 30 days.

For each such loan, a reminder message is printed using DBMS\_OUTPUT.PUT\_LINE, including the customer’s name, loan ID, and due date.

When this PL/SQL code is executed, it modifies the loan interest rates and customer VIP flags according to the given conditions and displays reminder messages for loans due soon.

The procedures demonstrate key PL/SQL features such as cursor-based loops, conditional logic, data manipulation (UPDATE), and output statements.

**OUTPUT**





