PLSQL Exercises Solved

Exercise 1: Control Structures

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.  
Code:

BEGIN

FOR rec IN (SELECT customer\_id, age FROM Customers) LOOP

IF rec.age > 60 THEN

UPDATE Loans

SET interest\_rate = interest\_rate - 1

WHERE customer\_id = rec.customer\_id;

END IF;

END LOOP;

END;

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.

Code:

BEGIN

FOR rec IN (SELECT customer\_id, balance FROM Customers) LOOP

IF rec.balance > 10000 THEN

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE customer\_id = rec.customer\_id;

END IF;

END LOOP;

END;

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:

BEGIN

FOR rec IN (

SELECT loan\_id, customer\_id, due\_date

FROM Loans

WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30

) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ' || rec.loan\_id || ' is due soon for customer ' || rec.customer\_id);

END LOOP;

END;

Exercise 3: Stored Procedures

Question: Write a stored procedure ProcessMonthlyInterest that calculates and updates the balance of all savings accounts by applying an interest rate of 1% to the current balance.

Code:

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE SavingsAccounts

SET balance = balance + (balance \* 0.01);

COMMIT;

END;

Question: Write a stored procedure UpdateEmployeeBonus that updates the salary of employees in a given department by adding a bonus percentage passed as a parameter.

Code:

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(dept\_id NUMBER, bonus\_percent NUMBER) IS

BEGIN

UPDATE Employees

SET salary = salary + (salary \* bonus\_percent / 100)

WHERE department\_id = dept\_id;

COMMIT;

END;

Question: Write a stored procedure **TransferFunds** that transfers a specified amount from one account to another, checking that the source account has sufficient balance before making the transfer.

Code:

CREATE OR REPLACE PROCEDURE TransferFunds(from\_id NUMBER, to\_id NUMBER, amount NUMBER) IS

from\_balance NUMBER;

BEGIN

SELECT balance INTO from\_balance FROM Accounts WHERE account\_id = from\_id FOR UPDATE;

IF from\_balance >= amount THEN

UPDATE Accounts SET balance = balance - amount WHERE account\_id = from\_id;

UPDATE Accounts SET balance = balance + amount WHERE account\_id = to\_id;

COMMIT;

ELSE

RAISE\_APPLICATION\_ERROR(-20004, 'Insufficient balance');

END IF;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE(SQLERRM);

END;