**Exercise 1: Control Structures**

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

**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

DECLARE

CURSOR customer\_cur IS

SELECT customer\_id, age, interest\_rate

FROM loan\_customers;

BEGIN

FOR cust\_rec IN customer\_cur LOOP

IF cust\_rec.age > 60 THEN

UPDATE loan\_customers

SET interest\_rate = interest\_rate - (interest\_rate \* 0.01)

WHERE customer\_id = cust\_rec.customer\_id;

END IF;

END LOOP;

COMMIT;

dbms\_output.put\_line('Discount applied ');

EXCEPTION

WHEN OTHERS THEN

dbms\_output.put\_line('An error occurred: ' || SQLERRM);

ROLLBACK;

END;

A screenshot of a computer

AI-generated content may be incorrect.

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

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

**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.

DECLARE

CURSOR cust\_cur IS

SELECT customer\_id, balance FROM bank\_customers;

BEGIN

FOR cust\_rec IN cust\_cur LOOP

IF cust\_rec.balance > 10000 THEN

UPDATE bank\_customers

SET is\_vip = 'Y'

WHERE customer\_id = cust\_rec.customer\_id;

END IF;

END LOOP;

COMMIT;

dbms\_output.put\_line('VIP status updated for eligible customers.');

EXCEPTION

WHEN OTHERS THEN

dbms\_output.put\_line('Error: ' || SQLERRM);

ROLLBACK;

END;

SELECT customer\_id, f\_name, balance,

CASE

WHEN is\_vip = 'Y' THEN 'VIP Customer'

ELSE 'Regular Customer'

END AS status

FROM bank\_customers;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

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

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

DECLARE

CURSOR due\_soon\_cur IS

SELECT customer\_name, due\_date

FROM loan\_details

WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30;

BEGIN

FOR loan\_rec IN due\_soon\_cur LOOP

dbms\_output.put\_line(

'Reminder: ' || loan\_rec.customer\_name ||

', your loan is due on ' || TO\_CHAR(loan\_rec.due\_date, 'DD-Mon-YYYY')

);

END LOOP;

END;

/

A screenshot of a computer

AI-generated content may be incorrect.