```
DECLARE
    CURSOR trans_cursor IS
        SELECT c.CustomerID, c.Name, t.TransactionDate, t.Amount, t.TransactionType
        FROM Customers c
        JOIN Accounts a ON c.CustomerID = a.CustomerID
        JOIN Transactions t ON a.AccountID = t.AccountID
        WHERE t.TransactionDate BETWEEN TRUNC(SYSDATE, 'MM') AND LAST_DAY(SYSDATE);

    v_customerID Customers.CustomerID%TYPE;
    v_name Customers.Name%TYPE;
    v_transactionDate Transactions.TransactionDate%TYPE;
    v_amount Transactions.Amount%TYPE;
    v_transactionType Transactions.TransactionType%TYPE;

BEGIN
    OPEN trans_cursor;
    LOOP
        FETCH trans_cursor INTO v_customerID, v_name, v_transactionDate, v_amount,
v_transactionType;
        EXIT WHEN trans_cursor%NOTFOUND;

        DBMS_OUTPUT.PUT_LINE('Customer ID: ' || v_customerID);
        DBMS_OUTPUT.PUT_LINE('Customer Name: ' || v_name);
        DBMS_OUTPUT.PUT_LINE('Transaction Date: ' || TO_CHAR(v_transactionDate, 'YYYY-MM-DD'));
        DBMS_OUTPUT.PUT_LINE('Amount: ' || v_amount);
        DBMS_OUTPUT.PUT_LINE('Transaction Type: ' || v_transactionType);
        DBMS_OUTPUT.PUT_LINE('----------------------------------');
    END LOOP;
    CLOSE trans_cursor;
END;
DECLARE
```

```
  CURSOR acc_cursor IS

    SELECT AccountID, Balance

    FROM Accounts;


  v_accountID Accounts.AccountID%TYPE;

  v_balance Accounts.Balance%TYPE;

  v_annual_fee CONSTANT NUMBER := 50; -- Assume annual fee is 50


BEGIN

  OPEN acc_cursor;

  LOOP

    FETCH acc_cursor INTO v_accountID, v_balance;

    EXIT WHEN acc_cursor%NOTFOUND;


    IF v_balance > v_annual_fee THEN

      UPDATE Accounts

      SET Balance = Balance - v_annual_fee

      WHERE AccountID = v_accountID;

    END IF;

  END LOOP;

  CLOSE acc_cursor;


  COMMIT;

END;

DECLARE

  CURSOR loan_cursor IS

    SELECT LoanID, InterestRate

    FROM Loans;


  v_loanID Loans.LoanID%TYPE;

  v_interestRate Loans.InterestRate%TYPE;
```

```
    v_new_interestRate Loans.InterestRate%TYPE;


    -- Define the new policy here
    PROCEDURE NewInterestPolicy(old_rate IN NUMBER, new_rate OUT NUMBER) IS
    BEGIN
        -- Example policy: reduce interest rate by 0.5% if the current rate is above 5%
        IF old_rate > 5 THEN
            new_rate := old_rate - 0.5;
        ELSE
            new_rate := old_rate;
        END IF;
    END NewInterestPolicy;


BEGIN
    OPEN loan_cursor;
    LOOP
        FETCH loan_cursor INTO v_loanID, v_interestRate;
        EXIT WHEN loan_cursor%NOTFOUND;


        NewInterestPolicy(v_interestRate, v_new_interestRate);


        UPDATE Loans
        SET InterestRate = v_new_interestRate
        WHERE LoanID = v_loanID;
    END LOOP;
    CLOSE loan_cursor;


    COMMIT;
END;
```