Exercise 6:

-- Customers

CREATE TABLE Customers (

CustomerID NUMBER PRIMARY KEY,

Name VARCHAR2(100),

DOB DATE,

Balance NUMBER,

LastModified DATE

);

-- Accounts

CREATE TABLE Accounts (

AccountID NUMBER PRIMARY KEY,

CustomerID NUMBER,

AccountType VARCHAR2(20),

Balance NUMBER,

LastModified DATE,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

-- Transactions

CREATE TABLE Transactions (

TransactionID NUMBER PRIMARY KEY,

AccountID NUMBER,

TransactionDate DATE,

Amount NUMBER,

TransactionType VARCHAR2(10),

FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID)

);

-- Loans

CREATE TABLE Loans (

LoanID NUMBER PRIMARY KEY,

CustomerID NUMBER,

LoanAmount NUMBER,

InterestRate NUMBER,

StartDate DATE,

EndDate DATE,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

-- Customers

INSERT INTO Customers VALUES (1, 'John Doe', TO\_DATE('1980-01-01','YYYY-MM-DD'), 2000, SYSDATE);

INSERT INTO Customers VALUES (2, 'Jane Smith', TO\_DATE('1990-07-10','YYYY-MM-DD'), 5000, SYSDATE);

-- Accounts

INSERT INTO Accounts VALUES (101, 1, 'Savings', 3000, SYSDATE);

INSERT INTO Accounts VALUES (102, 2, 'Checking', 7000, SYSDATE);

-- Transactions for current month

INSERT INTO Transactions VALUES (1, 101, SYSDATE, 500, 'Deposit');

INSERT INTO Transactions VALUES (2, 102, SYSDATE, 200, 'Withdrawal');

-- Transactions from previous month

INSERT INTO Transactions VALUES (3, 101, ADD\_MONTHS(SYSDATE, -1), 1000, 'Deposit');

-- Loans

INSERT INTO Loans VALUES (201, 1, 10000, 5, SYSDATE, ADD\_MONTHS(SYSDATE, 60));

INSERT INTO Loans VALUES (202, 2, 15000, 6, SYSDATE, ADD\_MONTHS(SYSDATE, 60));

BEGIN

DBMS\_OUTPUT.PUT\_LINE('--- Monthly Statements ---');

FOR txn IN (

SELECT c.Name, a.AccountID, t.TransactionDate, t.Amount, t.TransactionType

FROM Transactions t

JOIN Accounts a ON t.AccountID = a.AccountID

JOIN Customers c ON a.CustomerID = c.CustomerID

WHERE EXTRACT(MONTH FROM t.TransactionDate) = EXTRACT(MONTH FROM SYSDATE)

AND EXTRACT(YEAR FROM t.TransactionDate) = EXTRACT(YEAR FROM SYSDATE)

ORDER BY c.CustomerID

) LOOP

DBMS\_OUTPUT.PUT\_LINE('Customer: ' || txn.Name ||

' | Account: ' || txn.AccountID ||

' | Type: ' || txn.TransactionType ||

' | Amount: ' || txn.Amount ||

' | Date: ' || txn.TransactionDate);

END LOOP;

END;

/

DECLARE

CURSOR account\_cursor IS

SELECT AccountID, Balance FROM Accounts;

v\_fee NUMBER := 100; -- Annual maintenance fee

BEGIN

FOR acc IN account\_cursor LOOP

UPDATE Accounts

SET Balance = Balance - v\_fee

WHERE AccountID = acc.AccountID;

DBMS\_OUTPUT.PUT\_LINE('Annual fee of ' || v\_fee ||

' applied to Account ID: ' || acc.AccountID);

END LOOP;

END;

/

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, InterestRate FROM Loans;

v\_new\_rate NUMBER;

BEGIN

FOR loan IN loan\_cursor LOOP

IF loan.InterestRate < 6 THEN

v\_new\_rate := loan.InterestRate + 0.5;

ELSE

v\_new\_rate := loan.InterestRate;

END IF;

UPDATE Loans

SET InterestRate = v\_new\_rate

WHERE LoanID = loan.LoanID;

DBMS\_OUTPUT.PUT\_LINE('Loan ID ' || loan.LoanID ||

' updated interest rate to ' || v\_new\_rate);

END LOOP;

END;

/