Oracle SQL*Plus

File Edit Search Options Help

```
Copyright (c) 1982, 2005, Oracle.  All rights reserved.


Connected to:
Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options

SQL> CREATE TABLE Customers (
  2      CustomerID NUMBER PRIMARY KEY,
  3      Name VARCHAR2(100),
  4      DOB DATE,
  5      Balance NUMBER,
  6      LastModified DATE
  7  );

Table created.

SQL> CREATE TABLE Accounts (
  2      AccountID NUMBER PRIMARY KEY,
  3      CustomerID NUMBER,
  4      AccountType VARCHAR2(20),
  5      Balance NUMBER,
  6      LastModified DATE,
  7      FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
  8  );

Table created.

SQL> CREATE TABLE Transactions (
  2      TransactionID NUMBER PRIMARY KEY,
  3      AccountID NUMBER,
  4      TransactionDate DATE,
  5      Amount NUMBER,
  6      TransactionType VARCHAR2(10),
  7      FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID)
  8  );

Table created.

SQL> CREATE TABLE Loans (
  2      LoanID NUMBER PRIMARY KEY,
  3      CustomerID NUMBER,
  4      LoanAmount NUMBER,
  5      InterestRate NUMBER,
  6      StartDate DATE,
  7      EndDate DATE,
  8      FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
  9  );

Table created.
```

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> CREATE TABLE Employees (
  2      EmployeeID NUMBER PRIMARY KEY,
  3      Name VARCHAR2(100),
  4      Position VARCHAR2(50),
  5      Salary NUMBER,
  6      Department VARCHAR2(50),
  7      HireDate DATE
  8  );

Table created.

SQL> INSERT INTO Customers (CustomerID, Name, DOB, Balance, LastModified)
  2  VALUES (1, 'John Doe', TO_DATE('1985-05-15', 'YYYY-MM-DD'), 1000, SYSDATE);

1 row created.

SQL>
SQL> INSERT INTO Customers (CustomerID, Name, DOB, Balance, LastModified)
  2  VALUES (2, 'Jane Smith', TO_DATE('1990-07-20', 'YYYY-MM-DD'), 1500, SYSDATE);

1 row created.

SQL> INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified)
  2  VALUES (1, 1, 'Savings', 1000, SYSDATE);

1 row created.

SQL>
SQL> INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified)
  2  VALUES (2, 2, 'Checking', 1500, SYSDATE);

1 row created.

SQL> INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType)
  2  VALUES (1, 1, SYSDATE, 200, 'Deposit');

1 row created.

SQL>
SQL> INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType)
  2  VALUES (2, 2, SYSDATE, 300, 'Withdrawal');

1 row created.

SQL> INSERT INTO Loans (LoanID, CustomerID, LoanAmount, InterestRate, StartDate, EndDate)
  2  VALUES (1, 1, 5000, 5, SYSDATE, ADD_MONTHS(SYSDATE, 60));

1 row created.
```

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> INSERT INTO Employees (EmployeeID, Name, Position, Salary, Department, HireDate)
  2  VALUES (1, 'Alice Johnson', 'Manager', 70000, 'HR', TO_DATE('2015-06-15', 'YYYY-MM-DD'));

1 row created.

SQL>
SQL> INSERT INTO Employees (EmployeeID, Name, Position, Salary, Department, HireDate)
  2  VALUES (2, 'Bob Brown', 'Developer', 60000, 'IT', TO_DATE('2017-03-20', 'YYYY-MM-DD'));

1 row created.

SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
  2      v_age NUMBER;
  3      v_dob DATE;
  4      v_interest_rate NUMBER;
  5      CURSOR c_customers IS
  6          SELECT CustomerID, DOB FROM Customers;
  7  BEGIN
  8      FOR customer_rec IN c_customers LOOP
  9          v_dob := customer_rec.DOB;
 10
 11          -- Calculate age based on DOB
 12          v_age := EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM v_dob);
 13
 14          IF v_age > 60 THEN
 15              -- Update loan interest rate for customers above 60
 16              FOR loan_rec IN (SELECT LoanID, InterestRate FROM Loans WHERE CustomerID = customer
_rec.CustomerID) LOOP
 17                  v_interest_rate := loan_rec.InterestRate;
 18
 19                  -- Apply 1% discount
 20                  v_interest_rate := v_interest_rate - 1;
 21
 22                  -- Update the loan record with the new interest rate
 23                  UPDATE Loans
 24                  SET InterestRate = v_interest_rate
 25                  WHERE LoanID = loan_rec.LoanID;
 26              END LOOP;
 27          END IF;
 28      END LOOP;
 29
 30      COMMIT;
 31  END;
 32  /

PL/SQL procedure successfully completed.

SQL> ALTER TABLE Customers ADD (IsVIP VARCHAR2(5) DEFAULT 'FALSE');
```

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> BEGIN
  2      FOR customer_rec IN (SELECT CustomerID, Balance FROM Customers) LOOP
  3          IF customer_rec.Balance > 10000 THEN
  4              -- Update IsVIP status for customers with balance over $10,000
  5              UPDATE Customers
  6              SET IsVIP = 'TRUE'
  7              WHERE CustomerID = customer_rec.CustomerID;
  8          ELSE
  9              -- Ensure others are marked as not VIP
 10              UPDATE Customers
 11              SET IsVIP = 'FALSE'
 12              WHERE CustomerID = customer_rec.CustomerID;
 13          END IF;
 14      END LOOP;
 15
 16      COMMIT;
 17  END;
 18  /

PL/SQL procedure successfully completed.

SQL> DECLARE
  2      v_due_date DATE;
  3      v_customer_name VARCHAR2(100);
  4      v_loan_id NUMBER;
  5      CURSOR c_loans IS
  6          SELECT LoanID, CustomerID, EndDate
  7          FROM Loans
  8          WHERE EndDate BETWEEN SYSDATE AND SYSDATE + 30;
  9  BEGIN
 10      FOR loan_rec IN c_loans LOOP
 11          v_due_date := loan_rec.EndDate;
 12          v_loan_id := loan_rec.LoanID;
 13
 14          -- Fetch customer name
 15          SELECT Name INTO v_customer_name
 16          FROM Customers
 17          WHERE CustomerID = loan_rec.CustomerID;
 18
 19          -- Print reminder message
 20          DBMS_OUTPUT.PUT_LINE('Reminder: Customer ' || v_customer_name ||
 21                               ', your loan with Loan ID ' || v_loan_id ||
 22                               ' is due on ' || TO_CHAR(v_due_date, 'YYYY-MM-DD'));
 23      END LOOP;
 24  END;
 25  /

PL/SQL procedure successfully completed.

SQL>
```