**Exercise – 3 Stored Procedures**

**Scenario 1:** The bank needs to process monthly interest for all savings accounts.

**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 TABLE SavingsAccount (

  account\_id NUMBER PRIMARY KEY,

  customer\_name VARCHAR2(100),

  balance NUMBER(10,2)

);

INSERT INTO SavingsAccount VALUES (101, 'Alice', 1000.00);

INSERT INTO SavingsAccount VALUES (102, 'Bob', 2000.00);

INSERT INTO SavingsAccount VALUES (103, 'Charlie', 1650.00);

INSERT INTO SavingsAccount VALUES (104, 'David', 2100.00);

INSERT INTO SavingsAccount VALUES (105, 'Eva', 1700.00);

SELECT \* FROM SAVINGSACCOUNT;

CREATE PROCEDURE ProcessMonthlyInterest IS

BEGIN

  UPDATE SavingsAccount SET balance = balance \* 1.01;

END;

/

BEGIN

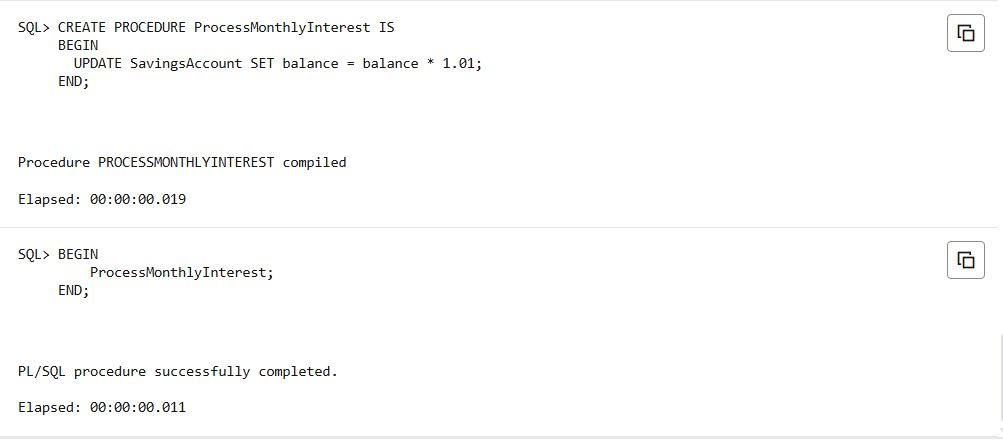
    ProcessMonthlyInterest;

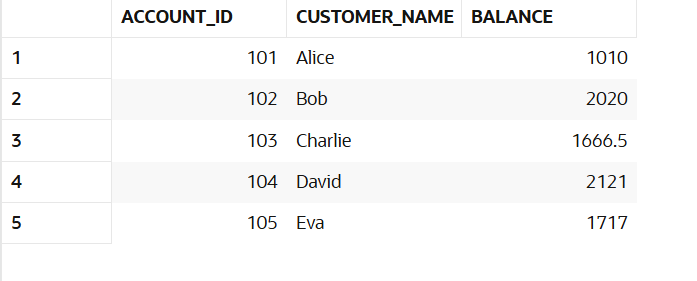
END;

/

SELECT \* FROM SavingsAccount;

**Output**





**Scenario 2:** The bank wants to implement a bonus scheme for employees based on their performance.

**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 TABLE Employee (

  emp\_id NUMBER PRIMARY KEY,

  emp\_name VARCHAR2(10),

  salary NUMBER(10,2),

  department\_id NUMBER

);

INSERT INTO Employee VALUES (1, 'Alice', 5000, 10);

INSERT INTO Employee VALUES (2, 'Bob', 6000, 10);

INSERT INTO Employee VALUES (3, 'Charlie', 7000, 20);

INSERT INTO Employee VALUES (4, 'David', 6500, 20);

INSERT INTO Employee VALUES (5, 'Eva', 8000, 30);

INSERT INTO Employee VALUES (6, 'Frank', 5700, 20);

CREATE PROCEDURE UpdateEmployeeBonus (

  id\_x IN NUMBER,

  bonus\_x IN NUMBER

) IS

BEGIN

  UPDATE Employee

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

  WHERE department\_id = id\_x;

END;

/

BEGIN

    UpdateEmployeeBonus(10, 10);

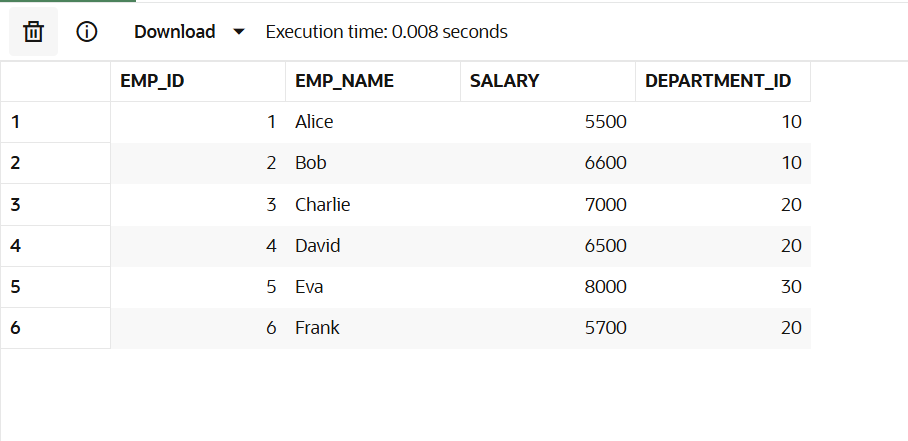
END;

/

SELECT \* FROM Employee;

**Output**





**Scenario 3:** Customers should be able to transfer funds between their accounts.

**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 TABLE Accounts(

  account\_id NUMBER PRIMARY KEY,

  customer\_name VARCHAR2(100),

  balance NUMBER(10,2)

);

INSERT INTO Accounts VALUES (1001, 'Alice', 5000);

INSERT INTO Accounts VALUES (1002, 'Bob', 3000);

INSERT INTO Accounts VALUES (1003, 'Charlie', 2000);

INSERT INTO Accounts VALUES (1004, 'David', 3500);

INSERT INTO Accounts VALUES (1005, 'Eva', 1700);

CREATE or replace PROCEDURE TransferFunds (

  from\_account IN NUMBER,

  to\_account   IN NUMBER,

  amount       IN NUMBER

) IS

BEGIN

  UPDATE Accounts

  SET balance = balance - amount

  WHERE account\_id = from\_account;

  UPDATE Accounts

  SET balance = balance + amount

  WHERE account\_id = to\_account;

END;

/

BEGIN

  TransferFunds(1001, 1002, 1000);

END;

/

SELECT \* FROM Accounts;

**Output**



