**Sample tables:**

1. **savings\_accounts Table**

|  |  |  |
| --- | --- | --- |
| account\_id | customer\_name | balance |
| 1 | Alice | 1000 |
| 2 | Bob | 2000 |
| 3 | Charlie | 1500 |

1. **employees Table**

|  |  |  |  |
| --- | --- | --- | --- |
| emp\_id | name | department\_id | salary |
| 101 | John | 10 | 30000 |
| 102 | Jane | 20 | 35000 |
| 103 | Mark | 10 | 40000 |

1. **accounts Table**

|  |  |  |
| --- | --- | --- |
| account\_id | customer\_name | balance |
| 1 | Alice | 5000 |
| 2 | Bob | 3000 |

**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.

**Ans:**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE savings\_accounts

SET balance = balance + (balance \* 0.01);

END;

/

**Output:**

Monthly interest applied to all savings accounts.

**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.

**Ans:**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

p\_dept\_id IN NUMBER,

p\_bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE employees

SET salary = salary + (salary \* p\_bonus\_percent / 100)

WHERE department\_id = p\_dept\_id;

END;

/

**Output:**

Bonus updated for department 10

**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.

**Ans:**

CREATE OR REPLACE PROCEDURE TransferFunds (

p\_from\_acc IN NUMBER,

p\_to\_acc IN NUMBER,

p\_amount IN NUMBER

) IS

v\_balance NUMBER;

BEGIN

SELECT balance INTO v\_balance FROM accounts WHERE account\_id = p\_from\_acc;

IF v\_balance >= p\_amount THEN

UPDATE accounts

SET balance = balance - p\_amount

WHERE account\_id = p\_from\_acc;

UPDATE accounts

SET balance = balance + p\_amount

WHERE account\_id = p\_to\_acc;

ELSE

DBMS\_OUTPUT.PUT\_LINE('Insufficient balance.');

END IF;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Account not found.');

END;

/

**Output:**

Transfer successful: 1000 transferred.