**Exercise 2: Error Handling**

**ANS1.**

CREATE OR REPLACE PROCEDURE SafeTransferFunds (

FRMAID IN Accounts.AccountID%TYPE,

TOAID IN Accounts.AccountID%TYPE,

PAMT IN NUMBER

) AS

VFBAL Accounts.Balance%TYPE;

VTOBAL Accounts.Balance%TYPE;

insufficient\_funds EXCEPTION;

BEGIN

SELECT Balance INTO VFBAL

FROM Accounts

WHERE AccountID = FRMAID

FOR UPDATE;

SELECT Balance INTO VTOBAL

FROM Accounts

WHERE AccountID = TOAID

FOR UPDATE;

IF PAMT > VFBAL THEN

RAISE insufficient\_funds;

ELSE

UPDATE Accounts

SET Balance = VFBAL - PAMT

WHERE AccountID = FRMAID;

UPDATE Accounts

SET Balance = VTOBAL + PAMT

WHERE AccountID = TOAID;

COMMIT;

END IF;

EXCEPTION

WHEN insufficient\_funds THEN

DBMS\_OUTPUT.PUT\_LINE('INSUFFICIENT FUNDS!!');

ROLLBACK;

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('AN UNEXPECTED ERROR OCCURRED: ' || SQLERRM);

ROLLBACK;

END SafeTransferFunds;

/

DECLARE

A NUMBER:=&A;

B NUMBER:=&B;

C NUMBER:=&C;

BEGIN

SafeTransferFunds(A,B,C);

END;

/

**========================================================================================**

**ANS2.**

CREATE OR REPLACE PROCEDURE UpdateSalary (

EID IN EMPLOYEES.EMPLOYEEID%TYPE,

PERCENTAGE IN NUMBER

) AS

ESAL EMPLOYEES.SALARY%TYPE;

BEGIN

SELECT SALARY INTO ESAL

FROM EMPLOYEES WHERE EMPLOYEEID=EID;

UPDATE EMPLOYEES

SET SALARY=ESAL+(ESAL\*PERCENTAGE)/100

WHERE EMPLOYEEID=EID;

COMMIT;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('EMPLOYEE ID DOES NOT EXIST ' || SQLERRM);

ROLLBACK;

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('AN UNEXPECTED ERROR OCCURRED: ' || SQLERRM);

ROLLBACK;

END;

/

DECLARE

A NUMBER:=&A;

B NUMBER:=&B;

BEGIN

UpdateSalary(A,B);

END;

/

**======================================================================================**

**ANS3.**

**CREATE OR REPLACE PROCEDURE AddNewCustomer (**

**CID IN CUSTOMERS.CUSTOMERID%TYPE,**

**CNAME IN CUSTOMERS.NAME%TYPE,**

**CDOB IN CUSTOMERS.DOB%TYPE,**

**CBAL IN CUSTOMERS.BALANCE%TYPE,**

**CLAST IN CUSTOMERS.LASTMODIFIED%TYPE,**

**CVIP IN CUSTOMERS.ISVIP%TYPE**

**) AS**

**BEGIN**

**INSERT INTO Customers VALUES (**

**CID, CNAME,CDOB,CBAL,CLAST,CVIP);**

**COMMIT;**

**EXCEPTION**

**WHEN DUP\_VAL\_ON\_INDEX THEN**

**DBMS\_OUTPUT.PUT\_LINE('SAME CUSTOMER ID ALREADY PRESENT: ' || SQLERRM);**

**ROLLBACK;**

**WHEN OTHERS THEN**

**DBMS\_OUTPUT.PUT\_LINE('An unexpected error occurred: ' || SQLERRM);**

**ROLLBACK;**

**END;**

**/**

**DECLARE**

**A NUMBER:=&A;**

**B VARCHAR2(100):=&B;**

**C DATE:=&C;**

**D NUMBER:=&D;**

**E DATE:=&E;**

**F CHAR(1):=&F;**

**BEGIN**

**AddNewCustomer(A,B,C,D,E,F);**

**END;**

**/**