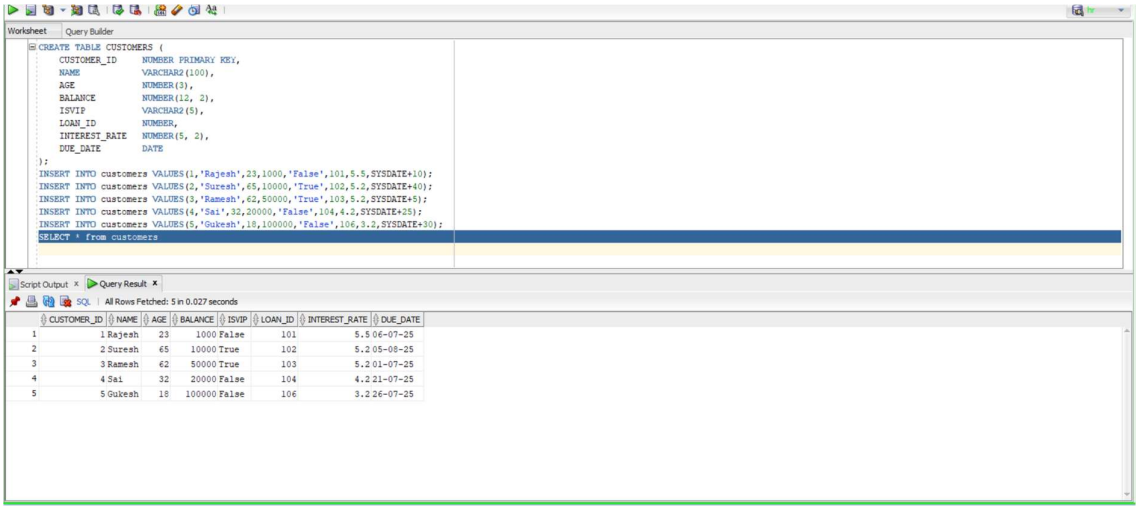
Exercise 1: Control Structures

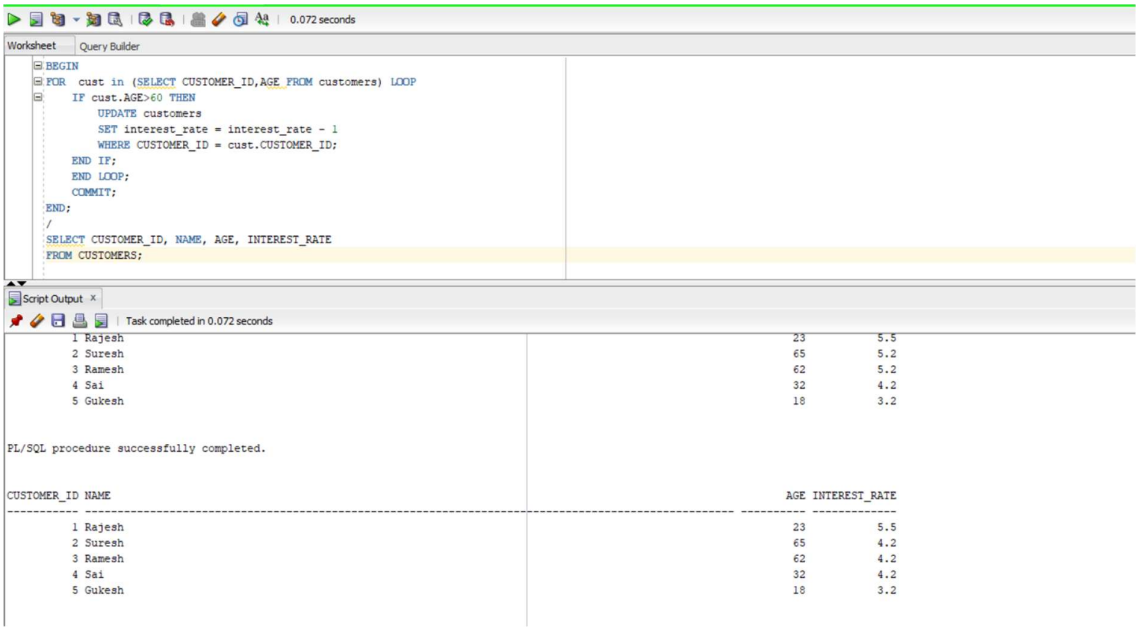
Table



```
CREATE TABLE CUSTOMERS (
    CUSTOMER_ID     NUMBER PRIMARY KEY,
    NAME            VARCHAR2(100),
    AGE             NUMBER(3),
    BALANCE         NUMBER(12, 2),
    ISVIP           VARCHAR2(5),
    LOAN_ID         NUMBER,
    INTEREST_RATE   NUMBER(5, 2),
    DUE_DATE        DATE
);
INSERT INTO customers VALUES(1,'Rajesh',23,1000,'False',101,5.5,SYSDATE+10);
INSERT INTO customers VALUES(2,'Suresh',65,10000,'True',102,5.2,SYSDATE+40);
INSERT INTO customers VALUES(3,'Ramesh',62,50000,'True',103,5.2,SYSDATE+5);
INSERT INTO customers VALUES(4,'Sai',32,20000,'False',104,4.2,SYSDATE+25);
INSERT INTO customers VALUES(5,'Gukesh',18,100000,'False',106,3.2,SYSDATE+30);
SELECT * from customers
```

CUSTOMER_ID	NAME	AGE	BALANCE	ISVIP	LOAN_ID	INTEREST_RATE	DUE_DATE
1	1 Rajesh	23	1000	False	101	5.5	06-07-25
2	2 Suresh	65	10000	True	102	5.2	05-08-25
3	3 Ramesh	62	50000	True	103	5.2	01-07-25
4	4 Sai	32	20000	False	104	4.2	21-07-25
5	5 Gukesh	18	100000	False	106	3.2	26-07-25

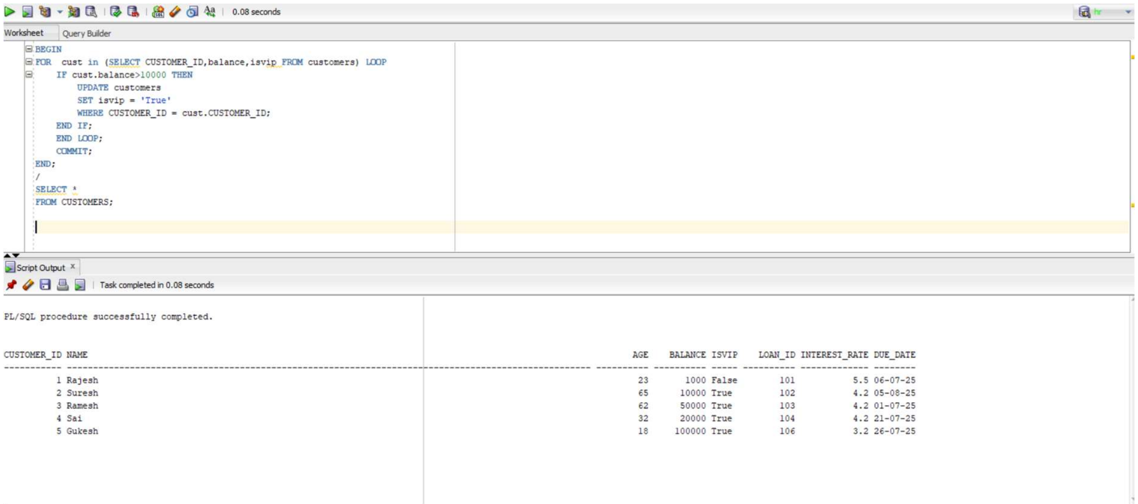
Scenario 1 Output



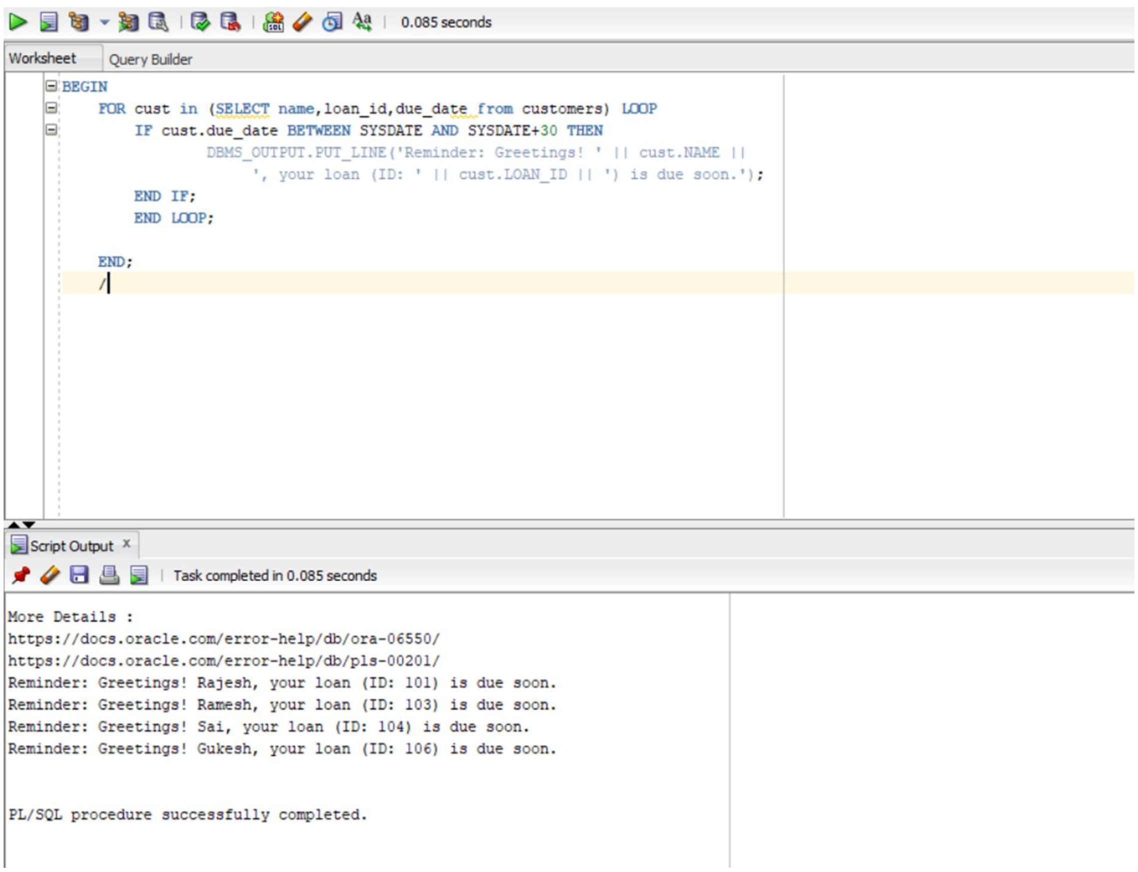
```
BEGIN
FOR  cust in (SELECT CUSTOMER_ID,AGE FROM customers) LOOP
    IF cust.AGE>60 THEN
        UPDATE customers
        SET interest_rate = interest_rate - 1
        WHERE CUSTOMER_ID = cust.CUSTOMER_ID;
    END IF;
    END LOOP;
    COMMIT;
END;
/
SELECT CUSTOMER_ID, NAME, AGE, INTEREST_RATE
FROM CUSTOMERS;
```

```
Task completed in 0.072 seconds

         1 Rajesh                                                                                                23          5.5
         2 Suresh                                                                                                65          5.2
         3 Ramesh                                                                                                62          5.2
         4 Sai                                                                                                   32          4.2
         5 Gukesh                                                                                                18          3.2


PL/SQL procedure successfully completed.


CUSTOMER_ID NAME                                                                                                AGE INTEREST_RATE
----------- ----------------------------------------------------------------------------------------------------
         1 Rajesh                                                                                                23          5.5
         2 Suresh                                                                                                65          4.2
         3 Ramesh                                                                                                62          4.2
         4 Sai                                                                                                   32          4.2
         5 Gukesh                                                                                                18          3.2
```

Scenario 2 Output



<u>Scenario 3 Output</u>



Exercise 3: Stored Procedures

Table

	ID	NAME	BALANCE	ACCOUNT_TYPE	DEPARTMENT_ID	SALARY
1	101	Rajesh	5000	SAVINGS	(null)	(null)
2	102	Suresh	7000	SAVINGS	(null)	(null)
3	103	Ramesh	(null)	EMPLOYEE	10	3150
4	104	Divya	(null)	EMPLOYEE	10	3360
5	105	Gukesh	10000	CHECKING	(null)	(null)
6	106	Mounica	8000	CHECKING	(null)	(null)

Scenario 1 Output

0.116 seconds

Worksheet | Query Builder

```sql
CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS
    BEGIN
        FOR acc in (SELECT id,balance,account_type FROM bank_data) LOOP
            IF acc.account_type ='SAVINGS' THEN
                UPDATE bank_data
                SET balance = acc.balance + (acc.balance*0.01)
                WHERE id = acc.id;
            END IF;
            END LOOP;
        COMMIT;
    END;
    /
EXEC ProcessMonthlyInterest;
SELECT id,name,account_type,balance FROM bank_data;
```

Script Output | Query Result

Task completed in 0.116 seconds

```
PL/SQL procedure successfully completed.


        ID NAME                                                    ACCOUNT_TYPE            BALANCE
---------- ------------------------------------------------------- -------------------- ----------
       101 Rajesh                                                  SAVINGS                 5151.51
       102 Suresh                                                  SAVINGS                 7212.11
       103 Ramesh                                                  EMPLOYEE
       104 Divya                                                   EMPLOYEE
       105 Gukesh                                                  CHECKING                  10000
       106 Mounica                                                 CHECKING                   8000
```

Scenario 2 Output

0.099 seconds

Worksheet | Query Builder

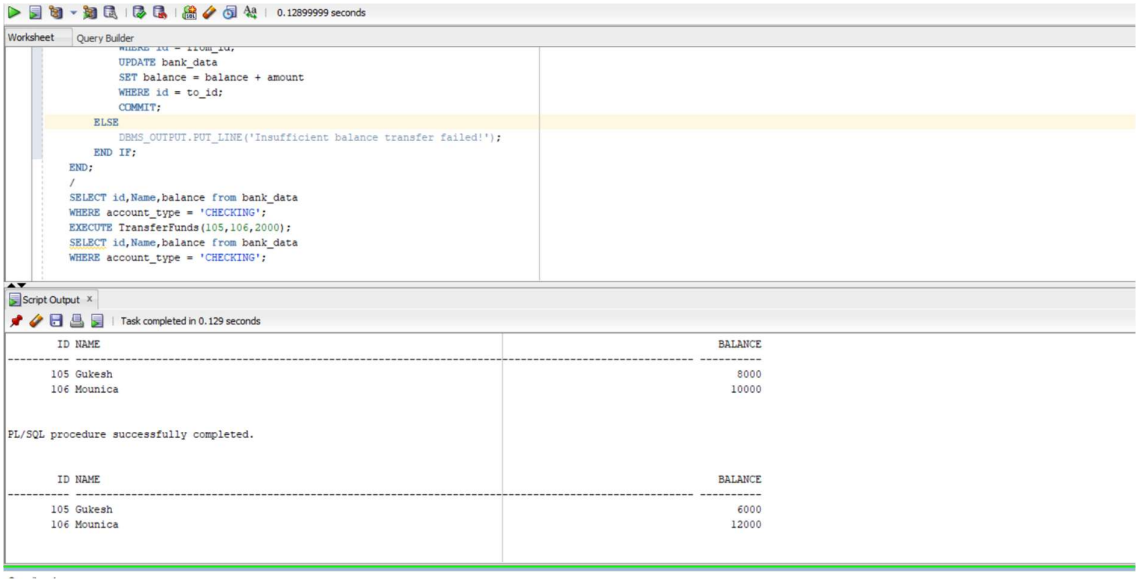
```sql
CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(
    dep_id in NUMBER,
    bonus_percentage in NUMBER
) IS
BEGIN
    UPDATE bank_data
    SET salary = salary + (salary *(bonus_percentage/100))
    WHERE department_id = dep_id;
    COMMIT;
END;
/
SELECT name,salary FROM bank_data WHERE account_type = 'EMPLOYEE';
EXECUTE updateemployeebonus(10,5);
SELECT name,salary FROM bank_data WHERE account_type ='EMPLOYEE'
```

Script Output | Query Result

Task completed in 0.099 seconds

```
NAME                                                        SALARY
----------------------------------------------------- ----------
Ramesh                                                        3150
Divya                                                         3360


PL/SQL procedure successfully completed.


NAME                                                        SALARY
----------------------------------------------------- ----------
Ramesh                                                      3307.5
Divya                                                         3528
```

Scenario 3 Output


0.12899999 seconds
Worksheet
Query Builder
WHERE id = from_id;
UPDATE bank_data
SET balance = balance + amount
WHERE id = to_id;
COMMIT;
ELSE
DBMS_OUTPUT.PUT_LINE('Insufficient balance transfer failed!');
END IF;
END;
/
SELECT id,Name,balance from bank_data
WHERE account_type = 'CHECKING';
EXECUTE TransferFunds(105,106,2000);
SELECT id,Name,balance from bank_data
WHERE account_type = 'CHECKING';
Script Output
Task completed in 0.129 seconds
ID NAME BALANCE
105 Gukesh 8000
106 Mounica 10000
PL/SQL procedure successfully completed.
ID NAME BALANCE
105 Gukesh 6000
106 Mounica 12000