MERGE:

MERGE INTO rewards r

USING staff s

ON (r.worker_id = s.worker_id)

WHEN MATCHED THEN

UPDATE SET r.reward_amount = s.pay * 0.05;

WORKER_ID	REWARD_AMOUNT
101	400
102	600
103	475

3 rows selected. 0.01 seconds

DECLARE

   v_emp_lname emp.last_name%TYPE;

BEGIN

   SELECT last_name

   INTO v_emp_lname

   FROM emp

   WHERE emp_id = 100;

   DBMS_OUTPUT.PUT_LINE('His last name is ' || v_emp_lname);

END;

His last name is Smith

Statement processed. 0.01 seconds

DECLARE

   v_emp_lname emp.last_name%TYPE;

   v_id emp.job_id%TYPE;

```
BEGIN

    SELECT last_name, job_id

    INTO v_emp_lname, v_id

    FROM emp

    WHERE emp_id = 202;


    DBMS_OUTPUT.PUT_LINE('Employee Last Name: ' || v_emp_lname);

    DBMS_OUTPUT.PUT_LINE('Job ID: '|| v_id);


END;
```

```
Employee Last Name: Smith
Job ID: SA_REP

Statement processed. 0.00 seconds
```

```
DECLARE

    v_sum_sal NUMBER(10,2);

    v_deptno NUMBER NOT NULL := 60;

BEGIN

    SELECT SUM(salary)

    INTO v_sum_sal

    FROM EMPLOYEE

    WHERE DEPT_ID = v_deptno;


    DBMS_OUTPUT.PUT_LINE('Dep #60 Salary Total: ' || v_sum_sal);

END;
```

```
Dep #60 Salary Total: 150000

Statement processed. 0.01 seconds
```

```
BEGIN
```

INSERT INTO copy_emp

(emp_id, emp_name, dept_id, salary)

VALUES (1009, 'Vivian', 80, 60000);

END;

EMP_ID	EMP_NAME	DEPT_ID	SALARY
1006	Bridget	60	95000
1007	Stella	90	70000
1002	Alex	80	70000
1001	Josh	80	60000
1004	Rhys	90	90000
1005	Jules	70	65000
1008	Ava	60	55000
1003	Harper	90	100000
1009	Vivian	80	60000

9 rows selected. 0.01 seconds

DECLARE

v_sal_increase employees.salary%TYPE := 15000;

BEGIN

UPDATE copy_emp

SET salary = salary + v_sal_increase

WHERE dept_id = 70;

END;

1 row(s) updated. 0.01 seconds

DECLARE

v_deptno empp.dept_id%TYPE := 80;

BEGIN

DELETE FROM copy_emp

WHERE dept_id = v_deptno;

END;

EMP_ID	EMP_NAME	DEPT_ID	SALARY
1006	Bridget	60	95000
1007	Stella	90	70000
1004	Rhys	90	90000
1005	Jules	70	80000
1008	Ava	60	55000
1003	Harper	90	100000

```
BEGIN
  MERGE INTO copy_emp c
  USING empp e
  ON (e.emp_id = c.emp_id)
  WHEN MATCHED THEN
    UPDATE SET
      c.emp_name = e.emp_name,
      c.dept_id = e.dept_id,
      c.salary = e.salary
  WHEN NOT MATCHED THEN
    INSERT (c.emp_id, c.emp_name, c.dept_id, c.salary)
    VALUES (e.emp_id, e.emp_name, e.dept_id, e.salary);
END;
```

1 row(s) updated. 0.01 seconds

```
CREATE TABLE emp_temp AS

 SELECT employee_id, first_name, last_name, email

 FROM employee1;

BEGIN

  UPDATE emp_temp

  SET    email = 'not available'

  WHERE  first_name LIKE 'B%';


  dbms_output.Put_line('Number of records updated: ' || TO_CHAR(SQL%ROWCOUNT));

END;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL
1	John	Doe	john.doe@example.com
2	Jane	Smith	jane.smith@example.com
3	Bill	Johnson	not available
4	Brian	Brown	not available
5	Emily	Davis	emily.davis@example.com

```
DECLARE

      CURSOR c_emp_cursor IS

        SELECT employee_id, last_name FROM employee1

        WHERE department_id = 30;

      v_empno  employee1.employee_id%TYPE;

      v_lname  employee1.last_name%TYPE;

    BEGIN

     OPEN c_emp_cursor;

     LOOP

      FETCH c_emp_cursor INTO v_empno, v_lname;
```

```
    EXIT WHEN c_emp_cursor%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE (v_empno || ' ' || v_lname);

    END LOOP;

    CLOSE c_emp_cursor;

  END;
```

```
3 Johnson

Statement processed. 0.02 seconds
```

```
DECLARE

    CURSOR c_emp_cursor IS

     SELECT employee_id, last_name FROM employees

     WHERE department_id = 30;

    v_emp_record  c_emp_cursor%ROWTYPE;

   BEGIN

    OPEN c_emp_cursor;

    LOOP

     FETCH c_emp_cursor INTO v_emp_record;

    EXIT WHEN c_emp_cursor%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE (v_emp_record.employee_id || ' ' ||

     v_emp_record.last_name);

    END LOOP;

    CLOSE c_emp_cursor;

   END;
```

```
3 Johnson

Statement processed. 0.02 seconds
```

```
CREATE TABLE persons (
```

```sql
    id NUMBER,

    name VARCHAR2(50),

    birthdate DATE

);

INSERT INTO persons (id, name, birthdate) VALUES (1, 'Alice', DATE '1990-05-15');

INSERT INTO persons (id, name, birthdate) VALUES (2, 'Bob', DATE '1985-11-23');

INSERT INTO persons (id, name, birthdate) VALUES (3, 'Charlie', DATE '2000-01-30');

INSERT INTO persons (id, name, birthdate) VALUES (4, 'Diana', DATE '1992-07-12');

INSERT INTO persons (id, name, birthdate) VALUES (5, 'Eve', DATE '1988-09-05');

INSERT INTO persons (id, name, birthdate) VALUES (6, 'Frank', DATE '1994-12-25');

INSERT INTO persons (id, name, birthdate) VALUES (7, 'Grace', DATE '1991-03-17');

INSERT INTO persons (id, name, birthdate) VALUES (8, 'Hank', DATE '1993-06-22');

INSERT INTO persons (id, name, birthdate) VALUES (9, 'Ivy', DATE '1987-10-11');

INSERT INTO persons (id, name, birthdate) VALUES (10, 'Jack', DATE '1995-04-18');

DECLARE

CURSOR persons_cursor IS

SELECT id, name, birthdate

FROM persons

ORDER BY birthdate ASC;

v_id persons.id%TYPE;

v_name persons.name%TYPE;

v_birthdate persons.birthdate%TYPE;

BEGIN

OPEN persons_cursor;

LOOP

FETCH persons_cursor INTO v_id, v_name, v_birthdate;

EXIT WHEN persons_cursor%NOTFOUND;

DBMS_OUTPUT.PUT_LINE('ID: ' || v_id || ', Name: ' || v_name || ', Birthdate: ' ||
TO_CHAR(v_birthdate, 'YYYY-MM-DD'));

END LOOP;

CLOSE persons_cursor;
```

```
1 row(s) inserted. 0.00 seconds

DECLARE CURSOR persons_cursor IS SELECT id, name, birthdate FROM persons ORDER BY birthdate ASC; v_id
persons_cursor; LOOP FETCH persons_cursor INTO v_id, v_name, v_birthdate; EXIT WHEN persons_cursor%NOT
TO_CHAR(v_birthdate, 'YYYY-MM-DD')); END LOOP; CLOSE persons_cursor; END;

ID: 2, Name: Bob, Birthdate: 1985-11-23
ID: 9, Name: Ivy, Birthdate: 1987-10-11
ID: 5, Name: Eve, Birthdate: 1988-09-05
ID: 1, Name: Alice, Birthdate: 1990-05-15
ID: 7, Name: Grace, Birthdate: 1991-03-17
ID: 4, Name: Diana, Birthdate: 1992-07-12
ID: 8, Name: Hank, Birthdate: 1993-06-22
ID: 6, Name: Frank, Birthdate: 1994-12-25
ID: 10, Name: Jack, Birthdate: 1995-04-18
ID: 3, Name: Charlie, Birthdate: 2000-01-30

Statement processed. 0.02 seconds
```

```
CREATE TABLE num_table (

  n NUMBER

);
DECLARE

  TYPE num_list_type IS TABLE OF NUMBER INDEX BY BINARY_INTEGER;

  v_nums num_list_type;

BEGIN

  v_nums(1) := 1;

  v_nums(2) := 3;

  v_nums(3) := 5;

  v_nums(4) := 7;

  v_nums(5) := 11;


  FORALL i IN v_nums.FIRST .. v_nums.LAST

    INSERT INTO num_table (n) VALUES (v_nums(i));


  FOR i IN v_nums.FIRST .. v_nums.LAST

  LOOP

    DBMS_OUTPUT.PUT_LINE('Inserted ' || SQL%BULK_ROWCOUNT(i) || ' row(s) on iteration ' || i);

  END LOOP;
```

END;

N
1
3
5
7
11

5 rows selected. 0.00 seconds

DECLARE

  a number(2) := 30;

BEGIN

  <<loopstart>>

  WHILE a < 50 LOOP

    dbms_output.put_line ('value of a: ' || a);

    a := a + 1;

    IF a = 35 THEN

      a := a + 1;

      GOTO loopstart;

    END IF;

  END LOOP;

END;

```
value of a: 30
value of a: 31
value of a: 32
value of a: 33
value of a: 34
value of a: 36
value of a: 37
value of a: 38
value of a: 39
value of a: 40
value of a: 41
value of a: 42
value of a: 43
value of a: 44
value of a: 45
value of a: 46
value of a: 47
value of a: 48
value of a: 49

Statement processed. 0.00 seconds
```

```sql
DECLARE

dividend   NUMBER := 10;

divisor    NUMBER := 0;

result     NUMBER;

BEGIN

  BEGIN

result := dividend / divisor;

    DBMS_OUTPUT.PUT_LINE('Result: ' || result);

  EXCEPTION

    WHEN ZERO_DIVIDE THEN

      DBMS_OUTPUT.PUT_LINE('Error: Division by zero');

  END;

END;
```

```
Error: Division by zero

Statement processed. 0.00 seconds
```

```sql
DECLARE

  emp_name   VARCHAR2(100);

emp_id     NUMBER := 225;
```

```
BEGIN

  BEGIN

    SELECT first_name INTO emp_name FROM employees WHERE employee_id = emp_id;

    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || emp_name);

  EXCEPTION

    WHEN NO_DATA_FOUND THEN

      DBMS_OUTPUT.PUT_LINE('Error: No matching record found');

  END;

END;
```



```
Error: No matching record found

Statement processed. 0.02 seconds
```

```
DECLARE

  v_dep_id employee1.department_id%TYPE := 90;

  v_emp_id employee1.employee_id%TYPE;

  v_emp_name employee1.first_name%TYPE;

  v_emp_email employee1.email%TYPE;

  v_exception_msg VARCHAR2(200);

BEGIN

  SELECT first_name, department_id, email

  INTO v_emp_name,v_dep_id, v_emp_email

  FROM employee1

  WHERE department_id = v_dep_id;

  EXCEPTION

    WHEN TOO_MANY_ROWS THEN

      v_exception_msg := 'Multiple rows found for the given department ID: ' || v_dep_id;

      DBMS_OUTPUT.PUT_LINE(v_exception_msg);

    WHEN OTHERS THEN

      v_exception_msg := 'An error occurred: ' || SQLERRM;
```

```
        DBMS_OUTPUT.PUT_LINE(v_exception_msg);
END;
```

An error occurred: ORA-01403: no data found

Statement processed. 0.01 seconds

```
DECLARE
 v_dep_id employee1.department_id%TYPE := 90;
 v_emp_id employee1.employee_id%TYPE;
 v_emp_name employee1.first_name%TYPE;
 v_emp_email employee1.email%TYPE;
 v_exception_msg VARCHAR2(200);
BEGIN
 SELECT first_name, department_id, email
 INTO v_emp_name,v_dep_id, v_emp_email
 FROM employee1
 WHERE department_id = v_dep_id;
 EXCEPTION
  WHEN TOO_MANY_ROWS THEN
   v_exception_msg := 'Multiple rows found for the given department ID: ' || v_dep_id;
   DBMS_OUTPUT.PUT_LINE(v_exception_msg);
  WHEN OTHERS THEN
   v_exception_msg := 'An error occurred: ' || SQLERRM;
   DBMS_OUTPUT.PUT_LINE(v_exception_msg);
END;
```

An error occurred: ORA-01403: no data found

Statement processed. 0.01 seconds