

1. Tampilkanlah ename dan salary dari tabel EMP dengan menggunakan cursor yang memiliki sebuah parameter. Kemudian Tampilkan data karyawan tersebut dimana karyawan tersebut memiliki salary yang nilainya lebih kecil dari salary dari parameter cursor.

The screenshot displays a SQL development environment with a 'Query Builder' window and a 'Script Output' window.

Query Builder Window:

```
-- Elmo Allistair - 12118220
-- M7
-- 1
SET SERVEROUTPUT ON
SET VERIFY OFF

DECLARE
    v_num number(4) := sparameter;
    cursor cur_peg is SELECT ename, sal FROM emp WHERE sal < v_num;
    v_ename emp.ename%type;
    v_sal emp.sal%type;
BEGIN
    OPEN cur_peg;
    LOOP
        FETCH cur_peg into v_ename, v_sal;
        EXIT WHEN cur_peg%notfound;
        dbms_output.put_line
            ('Pegawai dengan name '||v_ename||' memiliki gaji '|| v_sal);
    END LOOP;
    CLOSE cur_peg;
END;
```

Script Output Window:

The output shows the results of the PL/SQL procedure execution:

```
Pegawai dengan name SMITH memiliki gaji 800
Pegawai dengan name ALLEN memiliki gaji 1600
Pegawai dengan name WARD memiliki gaji 1250
Pegawai dengan name MARTIN memiliki gaji 1250
Pegawai dengan name TURNER memiliki gaji 1500
Pegawai dengan name ADAMS memiliki gaji 1100
Pegawai dengan name JAMES memiliki gaji 950
Pegawai dengan name MILLER memiliki gaji 1300

PL/SQL procedure successfully completed.
```

An 'Enter Substitution Variable' dialog box is open, prompting for a value for the parameter 'sparameter'. The value '2000' is entered in the text field.

2. Buatlah sebuah table yang sama dengan table EMP dengan nama EMP2. Kemudian buatlah sebuah cursor yang berguna untuk menghapus data karyawan yang memiliki empno sesuai dengan input user (gunakan parameter!!). Dan pastikan data yang telah dihapus tidak dapat dikembalikan dan langsung tampilkan table EMP2 (masukkan query kedalam PL/SQL!!).

The screenshot shows the Oracle SQL Developer interface. The 'Query Builder' window contains the following PL/SQL script:

```
-- 2
CREATE TABLE emp2 AS (SELECT * FROM emp);

SET VERIFY OFF
ACCEPT p_empno prompt 'input no. pegawai :'
DECLARE
v_empno number(4) := &p_empno;
cursor cur_peg is SELECT * FROM emp2;
BEGIN
FOR peg in cur_peg
LOOP
IF peg.empno = v_empno then
DELETE FROM emp2 WHERE empno = v_empno;
dbms_output.put_line('Data berhasil dihapus');
END IF;
END LOOP;
commit;
END;
```

An 'Enter Value' dialog box is open, prompting for 'input no. pegawai :'. The value '7839' has been entered.

The 'Script Output' window shows the following messages:

```
Table EMP2 created.

Data berhasil dihapus

PL/SQL procedure successfully completed.
```

| | EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
|----|-------|--------|----------|------|-----------|------|--------|--------|
| 1 | 7369 | SMITH | CLERK | 7902 | 17-DEC-80 | 800 | (null) | 20 |
| 2 | 7499 | ALLEN | SALESMAN | 7698 | 20-FEB-81 | 1600 | 300 | 30 |
| 3 | 7521 | WARD | SALESMAN | 7698 | 22-FEB-81 | 1250 | 500 | 30 |
| 4 | 7566 | JONES | MANAGER | 7839 | 02-APR-81 | 2975 | (null) | 20 |
| 5 | 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-81 | 1250 | 1400 | 30 |
| 6 | 7698 | BLAKE | MANAGER | 7839 | 01-MAY-81 | 2850 | (null) | 30 |
| 7 | 7782 | CLARK | MANAGER | 7839 | 09-JUN-81 | 2450 | (null) | 10 |
| 8 | 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 | 3000 | (null) | 20 |
| 9 | 7844 | TURNER | SALESMAN | 7698 | 08-SEP-81 | 1500 | 0 | 30 |
| 10 | 7876 | ADAMS | CLERK | 7788 | 23-MAY-87 | 1100 | (null) | 20 |
| 11 | 7900 | JAMES | CLERK | 7698 | 03-DEC-81 | 950 | (null) | 30 |
| 12 | 7902 | FORD | ANALYST | 7566 | 03-DEC-81 | 3000 | (null) | 20 |
| 13 | 7934 | MILLER | CLERK | 7782 | 23-JAN-82 | 1300 | (null) | 10 |

3. Buatlah sebuah cursor yang digunakan untuk melakukan update salary karyawan menjadi 2 kali gaji awal dimana nomor pegawai tersebut sesuai dengan input user (gunakan parameter!!). Kemudian tampilkan pesan 'Data berhasil di update!' ketika data telah berhasil di update.

The screenshot shows the Oracle SQL Developer interface. The 'Query Builder' tab is active, displaying a PL/SQL script. The script includes a cursor declaration and an update operation. A dialog box titled 'Enter Value' is open, prompting for 'input no. pegawai :'. The value '7566' is entered in the text field. Below the dialog, the 'Script Output' window shows the message 'Success UPDATE pegawai emp = 7566' and 'PL/SQL procedure successfully completed.'

```
-- 3
SET SERVEROUTPUT ON
SET VERIFY OFF
ACCEPT p_empno prompt 'input no. pegawai :'

DECLARE
    cursor cur_peg is SELECT * FROM emp2;
    peg cur_peg%rowtype;
BEGIN
    OPEN cur_peg;
    FETCH cur_peg into peg;
    IF cur_peg%found then
        UPDATE emp2 set sal = sal*2 WHERE empno = '&p_empno';
        dbms_output.put_line('Success UPDATE pegawai emp = '||&p_empno);
    END IF;
    CLOSE cur_peg;
END;
```

Script Output x

ScriptRunner Task

Success UPDATE pegawai emp = 7566

PL/SQL procedure successfully completed.

| | EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
|----|-------|--------|----------|------|-----------|------|--------|--------|
| 1 | 7369 | SMITH | CLERK | 7902 | 17-DEC-80 | 800 | (null) | 20 |
| 2 | 7499 | ALLEN | SALESMAN | 7698 | 20-FEB-81 | 1600 | 300 | 30 |
| 3 | 7521 | WARD | SALESMAN | 7698 | 22-FEB-81 | 1250 | 500 | 30 |
| 4 | 7566 | JONES | MANAGER | 7839 | 02-APR-81 | 5950 | (null) | 20 |
| 5 | 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-81 | 1250 | 1400 | 30 |
| 6 | 7698 | BLAKE | MANAGER | 7839 | 01-MAY-81 | 2850 | (null) | 30 |
| 7 | 7782 | CLARK | MANAGER | 7839 | 09-JUN-81 | 2450 | (null) | 10 |
| 8 | 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 | 3000 | (null) | 20 |
| 9 | 7844 | TURNER | SALESMAN | 7698 | 08-SEP-81 | 1500 | 0 | 30 |
| 10 | 7876 | ADAMS | CLERK | 7788 | 23-MAY-87 | 1100 | (null) | 20 |
| 11 | 7900 | JAMES | CLERK | 7698 | 03-DEC-81 | 950 | (null) | 30 |
| 12 | 7902 | FORD | ANALYST | 7566 | 03-DEC-81 | 3000 | (null) | 20 |
| 13 | 7934 | MILLER | CLERK | 7782 | 23-JAN-82 | 1300 | (null) | 10 |