

CIS 442

WEEK THREE

LAB ON BASIC STORED PROCEDURE

Database Programming

Author:

Jason MANSFIELD

Instructor:

Hamid KAYHAN

November 11, 2012

Contents

0.1	Create the Table	1
0.2	Insert rows	2
0.3	PL/SQL stored procedure	3
0.4	SPOOL Output	4

0.1 Create the Table

```
1  /*CIS442-WK3-Jason N Mansfield*/
2  /*This SQL simply creates the needed tables*/
3  DROP TABLE EMP;
4  DROP TABLE DEPT;
5  /*Create DEPT table*/
6
7  CREATE TABLE DEPT
8  (
9      Dept_id int NOT NULL,
10     Name varchar(50) NOT NULL,
11     Location varchar(100) NOT NULL,
12     CONSTRAINT pk_deptID PRIMARY KEY(Dept_id)
13 );
14 /*Create EMP Table*/
15 CREATE TABLE EMP
16 (
17     EMP_id int NOT NULL,
18     Name varchar(50) NOT NULL,
19     Job varchar(50) NOT NULL,
20     Manager varchar(50) NOT NULL,
21     Hire_dt date,
22     Salary numeric(8,2) NOT NULL,
23     Commission numeric(6,2) NOT NULL,
24     Dept_id int NOT NULL,
25     CONSTRAINT pk_empID PRIMARY KEY(EMP_id)
26 );
```

bspcreate.sql

0.2 Insert rows

```
/*CIS442-WK3-Jason N Mansfield*/
2 /*This SQL simply fills the needed tables with the class assigned data*/
/*DELETE previous data and re-fill*/
4 DELETE FROM DEPT;
DELETE FROM EMP;
6 /*fill MIT related values for DEPT*/
INSERT INTO DEPT
8 (Dept_id,Name, Location)
VALUES(1, 'School of Architecture and Planning','Rogers Building');
10 INSERT INTO DEPT
(Dept_id, Name, Location)
12 VALUES(2, 'School of Engineering', 'Eastman Laboratories');
INSERT INTO DEPT
14 (Dept_id, Name, Location)
VALUES(3, 'School of Humanities, Arts and Social Sciences', 'Maclaurin Buildings');
16 INSERT INTO DEPT
(Dept_id, Name, Location)
18 VALUES(4, 'Sloan School of Management','MIT Sloan');
INSERT INTO DEPT
20 (Dept_id, Name, Location)
VALUES(5, 'School of Science','Herman Building');
22 /*fill professors info*/
INSERT INTO EMP
24 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(1, 'Dr. Black', 'Assistant Professor of Architecture',
26 'Dr. Red','01-JAN-10',800.00,0000.00,10);
INSERT INTO EMP
28 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(2, 'Dr. Red', 'Professor of Architecture', 'Dr. Green',
30 '01-JAN-09',900.00,0000.00,10);
INSERT INTO EMP
32 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(3, 'Dr. Green', 'Associate Dean', 'Dr. Orange',
34 '20-JUN-08',1000.00,0000.00,11);
INSERT INTO EMP
36 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(4, 'Dr. Orange', 'Dean', 'Dr. Pink',
38 '01-MAY-07',700.00,0000.00,11);
INSERT INTO EMP
40 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(5, 'Dr. Pink', 'Assist Vice-Chancellor', 'Dr. Silver',
42 '04-AUG-09',3000.00,0000.00,12);
INSERT INTO EMP
44 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(6, 'Dr. Silver', 'Associate Provost', 'Dr. Yellow',
46 '23-JAN-89',3000.00,0000.00,12);
INSERT INTO EMP
48 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(7, 'Dr. Yellow', 'Provost', 'Dr. White',
50 '20-FEB-85',3000.00,0000.00,12);
INSERT INTO EMP
52 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(8, 'Dr. White', 'Chancellor', 'Edward Ming',
54 '01-JAN-80',1000.00,0000.00,13);
INSERT INTO EMP
56 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(9, 'Dr. Brown', 'Chancellor', 'Edward Ming',
58 '30-APR-79',1000.00,0000.00,13);
INSERT INTO EMP
60 (EMP_id,Name,Job,Manager,Hire_dt,Salary,Commission,Dept_id)
VALUES(10, 'Dr. Entropy', 'Professor of Science', 'Dr. Yellow',
62 '02-NOV-10',700.00,0000.00,10);
```

bspfill.sql

0.3 PL/SQL stored procedure

```
DECLARE
2   v_name EMP.name%TYPE;
   v_salary EMP.Salary%Type;
4   com EMP.Commission%Type;
   prompt INT;
6 BEGIN
   prompt := 1;
   com := 0.0;
8   SELECT name
      INTO v_name
     FROM EMP
    WHERE EMP_ID = prompt;
12  DBMS_OUTPUT.PUT_LINE('Employee: ' || v_name);
14  SELECT Salary
      INTO v_salary
     FROM EMP
    WHERE EMP_ID = prompt;
18  DBMS_OUTPUT.PUT_LINE('Salary: ' || v_salary);
/*Begin IF Statements*/
20  IF v_salary < 1000 THEN
    com := v_salary * 0.10;
22  DBMS_OUTPUT.PUT_LINE('Commission: ' || com);
    DBMS_OUTPUT.PUT_LINE('10% Commission');
24  ELSIF
    v_salary >= 1000 AND v_salary < 1500 THEN
26  com := v_salary * 0.15;
    DBMS_OUTPUT.PUT_LINE('Commission: ' || com);
28  DBMS_OUTPUT.PUT_LINE('15% Commission');
    ELSIF
30  v_salary > 1500 THEN
    com := v_salary * 0.20;
32  DBMS_OUTPUT.PUT_LINE('Commission: ' || com);
    DBMS_OUTPUT.PUT_LINE('20% Commission');
34  ELSIF
    v_salary = 0 THEN
36  DBMS_OUTPUT.PUT_LINE('0% Commission');
    END IF;
38 /*End IF statements*/
    UPDATE EMP
40 SET Commission=com
   WHERE EMP_ID = prompt;
42 /*exception if ID is not found*/
   EXCEPTION
44 WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('Not a valid ID');
46 END;
/
```

plscript.sql

0.4 SPOOL Output

The following series of employee ids were placed as the variable for prompt followed by running the plscript.sql file:

- 0 which is not a valid id so it simply uses the exception to finish without an error.
- 1 which is Dr. Black who makes 800.00
- 3 which is Dr. Green who makes 1000.00
- 6 which is Dr. Silver who makes 3000.00

View the spool output and you can see the calculated commissions being updated for all three individuals each time prompt is change and plscript is called. Notice that the first call to plscript is when an invalid employee id is offered. Therefore, no commissions are updated on the first run.

```
2 @\\Client\H$\Development\CIS442\CIS442-WK3-Lab\bspcreate.sql
3 Table dropped.
4 DROP TABLE DEPT
5 *
6 ERROR at line 1:
7 ORA-00942: table or view does not exist
8
9
10 Table created.
11
12 CREATE TABLE DEPT
13 *
14 ERROR at line 1:
15 ORA-00955: name is already used by an existing object
16
17
18 Table created.
19
20 SQL> @\\Client\H$\Development\CIS442\CIS442-WK3-Lab\bspfill.sql
21 CREATE TABLE EMP
22 *
23 ERROR at line 1:
24 ORA-00955: name is already used by an existing object
25
26
27 CREATE TABLE EMP
28 *
29 ERROR at line 1:
30 ORA-00955: name is already used by an existing object
31
32
33 CREATE TABLE EMP
34 *
35 ERROR at line 1:
36 ORA-00955: name is already used by an existing object
37
38
39
40 0 rows deleted.
41
42
43
44 0 rows deleted.
45
46
47 0 rows deleted.
48
```

```

50 1 row created.
52
54 1 row created.
56
58 1 row created.
60
62 1 row created.
64 INSERT INTO DEPT
66 *
68 ERROR at line 1:
ORA-00001: unique constraint (CIS442FX70_08.PK_DEPTID) violated
70
72 1 row created.
74
76 1 row created.
78
80 1 row created.
82
84 1 row created.
86
88 1 row created.
90
92 1 row created.
94
96 1 row created.
98
100 SQL>
102 SQL> set PAGESIZE 180;
104 SQL> set LINESIZE 264;
SQL> SELECT EMP_id, Name, Salary, Commission FROM EMP;
106
EMP_ID NAME SALARY COMMISSION
-----
108 1 Dr. Black 800 0
110 2 Dr. Red 900 0
112 3 Dr. Green 1000 0
114 4 Dr. Orange 700 0
116 5 Dr. Pink 3000 0
118 6 Dr. Silver 3000 0
120 7 Dr. Yellow 3000 0
122 8 Dr. White 1000 0
124 9 Dr. Brown 1000 0
126 10 Dr. Entropy 700 0
128
10 rows selected.
SQL> @\\Client\\H$\\Development\\CIS442\\CIS442-WK3-Lab\\plscript.sql
122
PL/SQL procedure successfully completed.
124
SQL> SELECT EMP_id, Name, Salary, Commission FROM EMP;
126
EMP_ID NAME SALARY COMMISSION
-----
128 1 Dr. Black 800 0
130 2 Dr. Red 900 0
3 Dr. Green 1000 0

```

```

132      4 Dr. Orange                700      0
133      5 Dr. Pink                  3000     0
134      6 Dr. Silver                3000     0
135      7 Dr. Yellow                 3000     0
136      8 Dr. White                 1000     0
137      9 Dr. Brown                 1000     0
138     10 Dr. Entropy               700      0
139
140 10 rows selected.
141
142 SQL> @\\Client\H$\Development\CIS442\CIS442-WK3-Lab\plscript.sql
143
144 PL/SQL procedure successfully completed.
145
146 SQL> SELECT EMP_id, Name, Salary, Commission FROM EMP;
147
148      EMP_ID NAME                SALARY COMMISSION
149 -----
150      1 Dr. Black                800      80
151      2 Dr. Red                  900      0
152      3 Dr. Green                1000     0
153      4 Dr. Orange               700      0
154      5 Dr. Pink                 3000     0
155      6 Dr. Silver               3000     0
156      7 Dr. Yellow              3000     0
157      8 Dr. White                1000     0
158      9 Dr. Brown               1000     0
159     10 Dr. Entropy              700      0
160
161 10 rows selected.
162
163 SQL> @\\Client\H$\Development\CIS442\CIS442-WK3-Lab\plscript.sql
164
165 PL/SQL procedure successfully completed.
166
167 SQL> SELECT EMP_id, Name, Salary, Commission FROM EMP;
168
169      EMP_ID NAME                SALARY COMMISSION
170 -----
171      1 Dr. Black                800      80
172      2 Dr. Red                  900      0
173      3 Dr. Green                1000    150
174      4 Dr. Orange               700      0
175      5 Dr. Pink                 3000     0
176      6 Dr. Silver               3000     0
177      7 Dr. Yellow              3000     0
178      8 Dr. White                1000     0
179      9 Dr. Brown               1000     0
180     10 Dr. Entropy              700      0
181
182 10 rows selected.
183
184 SQL> @\\Client\H$\Development\CIS442\CIS442-WK3-Lab\plscript.sql
185
186 PL/SQL procedure successfully completed.
187
188 SQL> SELECT EMP_id, Name, Salary, Commission FROM EMP;
189
190      EMP_ID NAME                SALARY COMMISSION
191 -----
192      1 Dr. Black                800      80
193      2 Dr. Red                  900      0
194      3 Dr. Green                1000    150
195      4 Dr. Orange               700      0
196      5 Dr. Pink                 3000     0
197      6 Dr. Silver               3000    600
198      7 Dr. Yellow              3000     0
199      8 Dr. White                1000     0
200      9 Dr. Brown               1000     0
201     10 Dr. Entropy              700      0
202
203 10 rows selected.
204
205 SQL>

```

Mansfield-CIS442-WK3-SPOOL.txt