Database Management System

EXPERIMENT 1 CREATING AND MANAGING TABLES

NAME: Madhan RV

ROLL: 241801142

1. Create EMPLOYEES table

```
CREATE TABLE employees (
    employee_id NUMBER(6) NOT NULL,
    first_name VARCHAR2(20),
    last_name VARCHAR2(25) NOT NULL,
    email VARCHAR2(25) NOT NULL,
    phone_number VARCHAR2(20),
    hire_date DATE NOT NULL,
    job_id VARCHAR2(10) NOT NULL,
    salary NUMBER(8,2),
    commission_pct NUMBER(2,2),
    manager_id NUMBER(6),
    department_id NUMBER(4)
);
```

Expected Output:

```
Table created.
```

Verify table structure:

```
DESC employees;
```

Expected Output:

```
Name           Null?    Type
-------------- -------- ------------
EMPLOYEE_ID    NOT NULL NUMBER(6)
FIRST_NAME              VARCHAR2(20)
LAST_NAME      NOT NULL VARCHAR2(25)
EMAIL          NOT NULL VARCHAR2(25)
PHONE_NUMBER            VARCHAR2(20)
HIRE_DATE      NOT NULL DATE
JOB_ID         NOT NULL VARCHAR2(10)
SALARY                  NUMBER(8,2)
COMMISSION_PCT          NUMBER(2,2)
MANAGER_ID              NUMBER(6)
DEPARTMENT_ID           NUMBER(4)
```

2. Create DEPARTMENT table

```
CREATE TABLE department (
    dept_id NUMBER(6) NOT NULL,
    dept_name VARCHAR2(20) NOT NULL,
    manager_id NUMBER(6),
    location_id NUMBER(4)
);
```

Expected Output:

```
Table created.
```

Verify table structure:

```
DESC department;
```

Expected Output:

```
Name         Null?    Type
------------ -------- -----------
DEPT_ID      NOT NULL NUMBER(6)
DEPT_NAME    NOT NULL VARCHAR2(20)
MANAGER_ID            NUMBER(6)
LOCATION_ID           NUMBER(4)
```

3. Create JOB_GRADE table

```
CREATE TABLE job_grade (
    grade_level VARCHAR2(2),
    lowest_sal NUMBER,
    highest_sal NUMBER
);
```

Expected Output:

```
Table created.
```

4. Create LOCATION table

```
CREATE TABLE location (
    location_id NUMBER(4) NOT NULL,
    st_addr VARCHAR2(40),
    postal_code VARCHAR2(12),
    city VARCHAR2(30) NOT NULL,
    state_province VARCHAR2(25),
    country_id CHAR(2)
```

```
);
```

Expected Output:

```
Table created.
```

Insert Sample Data:

Insert into EMPLOYEES:

```
INSERT INTO employees VALUES (100, 'Steven', 'King',
'SKING', '515.123.4567',
                            TO_DATE('17-JUN-1987', 'DD-MON-
YYYY'), 'AD_PRES', 24000, NULL, NULL, 90);
INSERT INTO employees VALUES (101, 'Neena', 'Kochhar',
'NKOCHHAR', '515.123.4568',
                            TO_DATE('21-SEP-1989', 'DD-MON-
YYYY'), 'AD_VP', 17000, NULL, 100, 90);
INSERT INTO employees VALUES (102, 'Lex', 'De Haan',
'LDEHAAN', '515.123.4569',
                            TO_DATE('13-JAN-1993', 'DD-MON-
YYYY'), 'AD_VP', 17000, NULL, 100, 90);
INSERT INTO employees VALUES (103, 'Alexander', 'Hunold',
'AHUNOLD', '590.423.4567',
                            TO_DATE('03-JAN-1990', 'DD-MON-
YYYY'), 'IT_PROG', 9000, NULL, 102, 60);
INSERT INTO employees VALUES (104, 'Bruce', 'Ernst',
'BERNST', '590.423.4568',
                            TO_DATE('21-MAY-1991', 'DD-MON-
YYYY'), 'IT_PROG', 6000, NULL, 103, 60);
INSERT INTO employees VALUES (107, 'Diana', 'Lorentz',
'DLORENTZ', '590.423.4568',
                            TO_DATE('07-FEB-1999', 'DD-MON-
```

```
YYYY'), 'IT_PROG', 4200, NULL, 103, 60);
COMMIT;
```

Expected Output:

```
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
Commit complete.
```

Insert into DEPARTMENT:

```
INSERT INTO department VALUES (10, 'Administration', 200,
1700);
INSERT INTO department VALUES (20, 'Marketing', 201, 1800);
INSERT INTO department VALUES (50, 'Shipping', 124, 1500);
INSERT INTO department VALUES (60, 'IT', 103, 1400);
INSERT INTO department VALUES (80, 'Sales', 149, 2500);
INSERT INTO department VALUES (90, 'Executive', 100, 1700);
INSERT INTO department VALUES (110, 'Accounting', 205,
1700);
COMMIT;
```

Expected Output:

```
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
```

```
1 row created.
1 row created.
Commit complete.
```

Insert into JOB_GRADE:

```
INSERT INTO job_grade VALUES ('A', 1000, 2999);
INSERT INTO job_grade VALUES ('B', 3000, 5999);
INSERT INTO job_grade VALUES ('C', 6000, 9999);
INSERT INTO job_grade VALUES ('D', 10000, 14999);
INSERT INTO job_grade VALUES ('E', 15000, 24999);
INSERT INTO job_grade VALUES ('F', 25000, 40000);
COMMIT;
```

Expected Output:

```
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
Commit complete.
```

Insert into LOCATION:

```
INSERT INTO location VALUES (1400, '2014 Jabberwocky Rd',
'26192', 'Southlake', 'Texas', 'US');
INSERT INTO location VALUES (1500, '2011 Interior Blvd',
'99236', 'South San Francisco', 'California', 'US');
INSERT INTO location VALUES (1700, '2004 Charade Rd',
'98199', 'Seattle', 'Washington', 'US');
INSERT INTO location VALUES (1800, '460 Bloor St', 'M5S
```

```
1XB', 'Toronto', 'Ontario', 'CA');
INSERT INTO location VALUES (2500, 'Magdalen Centre, The
Oxford Science Park', 'OX9 9ZB', 'Oxford', 'Oxford', 'UK');
COMMIT;
```

Expected Output:

```
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
Commit complete.
```

Verify Data:

Check EMPLOYEES data:

```
SELECT * FROM employees;
```

Expected Output:

```
EMPLOYEE_ID FIRST_NAME           LAST_NAME
EMAIL                     PHONE_NUMBER         HIRE_DATE
JOB_ID     SALARY COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
----------- -------------------- ------------------------- -
---------------------- ------------------- --------- ----
------ ---------- ------------- ---------- -------------
        100 Steven               King
SKING                     515.123.4567         17-JUN-87
AD_PRES      24000                          90
        101 Neena                Kochhar
NKOCHHAR                  515.123.4568         21-SEP-89
```

```
AD_VP        17000                100         90
        102 Lex                   De Haan
LDEHAAN                  515.123.4569         13-JAN-93
AD_VP        17000                100         90
        103 Alexander             Hunold
AHUNOLD                  590.423.4567         03-JAN-90
IT_PROG       9000                102         60
        104 Bruce                 Ernst
BERNST                   590.423.4568         21-MAY-91
IT_PROG       6000                103         60
        107 Diana                 Lorentz
DLORENTZ                 590.423.4568         07-FEB-99
IT_PROG       4200                103         60
```

Check DEPARTMENT data:

```
SELECT * FROM department;
```

Expected Output:

```
  DEPT_ID DEPT_NAME               MANAGER_ID LOCATION_ID
---------- -------------------- ---------- -----------
        10 Administration                200        1700
        20 Marketing                     201        1800
        50 Shipping                      124        1500
        60 IT                            103        1400
        80 Sales                         149        2500
        90 Executive                     100        1700
       110 Accounting                    205        1700
```

Check JOB_GRADE data:

```
SELECT * FROM job_grade;
```

Expected Output:

```
GR LOWEST_SAL HIGHEST_SAL
-- ---------- -----------
A        1000        2999
B        3000        5999
C        6000        9999
D       10000       14999
E       15000       24999
F       25000       40000
```

Check LOCATION data:

```
SELECT * FROM location;
```

Expected Output:

```
LOCATION_ID ST_ADDR
POSTAL_CODE  CITY                              STATE_PROVINCE
CO
----------- ---------------------------------------- -------
----- ------------------------------ ----------------------
-- --
       1400 2014 Jabberwocky Rd                        26192
Southlake                      Texas                      US
       1500 2011 Interior Blvd                         99236
South San Francisco            California                 US
       1700 2004 Charade Rd                            98199
Seattle                        Washington                 US
       1800 460 Bloor St                               M5S 1XB
Toronto                        Ontario                    CA
```

```
       2500 Magdalen Centre, The Oxford Science Park OX9 9ZB
Oxford                          Oxford                       UK
```

Exercise Solutions:

1. Create DEPT table based on the table instance chart

```
CREATE TABLE dept (
    id NUMBER(7),
    name VARCHAR2(25)
);
```

Expected Output:

```
Table created.
```

Verify:

```
DESC dept;
```

Expected Output:

```
Name Null?    Type
---- -------- -----------
ID            NUMBER(7)
NAME          VARCHAR2(25)
```

2. Create EMP table based on the instance chart

```
CREATE TABLE emp (
    id NUMBER(7),
    last_name VARCHAR2(25),
    first_name VARCHAR2(25),
```

```
    dept_id NUMBER(7)
);
```

Expected Output:

```
Table created.
```

Verify:

```
DESC emp;
```

Expected Output:

```
Name       Null? Type
---------- -------- -----------
ID                  NUMBER(7)
LAST_NAME           VARCHAR2(25)
FIRST_NAME          VARCHAR2(25)
DEPT_ID             NUMBER(7)
```

3. Modify EMP table to allow longer employee last names

```
ALTER TABLE emp MODIFY last_name VARCHAR2(50);
```

Expected Output:

```
Table altered.
```

Verify:

```
DESC emp;
```

Expected Output:

```
Name       Null? Type
---------- -------- -----------
ID                  NUMBER(7)
LAST_NAME           VARCHAR2(50)
FIRST_NAME          VARCHAR2(25)
DEPT_ID             NUMBER(7)
```

4. Create EMPLOYEES2 table from EMPLOYEES structure

```
CREATE TABLE employees2 AS
SELECT
    employee_id AS id,
    first_name,
    last_name,
    salary,
    department_id AS dept_id
FROM employees
WHERE 1=0;
```

Expected Output:

```
Table created.
```

Verify:

```
DESC employees2;
```

Expected Output:

```
Name      Null? Type
--------- -------- ------------
```

```
ID                NUMBER(6)
FIRST_NAME        VARCHAR2(20)
LAST_NAME         VARCHAR2(25)
SALARY            NUMBER(8,2)
DEPT_ID           NUMBER(4)
```

5. Drop the EMP table

```
DROP TABLE emp;
```

Expected Output:

```
Table dropped.
```

6. Rename EMPLOYEES2 table as EMP

```
RENAME employees2 TO emp;
```

Expected Output:

```
Table renamed.
```

Verify:

```
SELECT table_name FROM user_tables WHERE table_name = 'EMP';
```

Expected Output:

```
TABLE_NAME
----------
EMP
```

7. Add comments on DEPT and EMP tables

```
COMMENT ON TABLE dept IS 'This table stores department
information';
COMMENT ON TABLE emp IS 'This table stores employee
details';
```

Expected Output:

```
Comment created.
Comment created.
```

Verify comments:

```
SELECT table_name, comments
FROM user_tab_comments
WHERE table_name IN ('DEPT', 'EMP');
```

Expected Output:

```
TABLE_NAME COMMENTS
---------- --------------------
DEPT       This table stores department information
EMP        This table stores employee details
```

8. Drop the First_name column from EMP table

```
ALTER TABLE emp DROP COLUMN first_name;
```

Expected Output:

```
Table altered.
```

Verify:

```
DESC emp;
```

Expected Output:

```
Name      Null? Type
--------- -------- ------------
ID                 NUMBER(6)
LAST_NAME          VARCHAR2(25)
SALARY             NUMBER(8,2)
DEPT_ID            NUMBER(4)
```

9. Mark DEPT_ID column as UNUSED in EMP table

```
ALTER TABLE emp SET UNUSED (dept_id);
```

Expected Output:

```
Table altered.
```

Verify:

```
DESC emp;
```

Expected Output:

```
Name      Null? Type
--------- -------- ------------
ID                 NUMBER(6)
LAST_NAME          VARCHAR2(25)
SALARY             NUMBER(8,2)
```

10. Drop all UNUSED columns from EMP table

```
ALTER TABLE emp DROP UNUSED COLUMNS;
```

Expected Output:

```
Table altered.
```

Final Table Status:

List all tables created:

```
SELECT table_name FROM user_tables;
```

Expected Output:

```
TABLE_NAME
----------
EMPLOYEES
DEPARTMENT
JOB_GRADE
LOCATION
DEPT
EMP
```