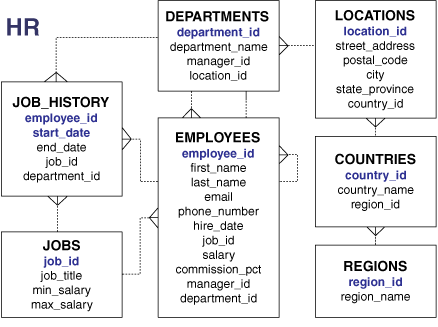
## **Objectives:**

The aim of this lab is to practice DML commands with subqueries. We do this based on the HR database described below.



## **Lab Work:**

1. We can insert into a table values through a subquery as in

create table sales\_reps(id number primary key, name varchar2(25), salary number(8,2), commission\_pct number(2,2));

INSERT INTO sales\_reps(id, name, salary, commission\_pct)

SELECT employee\_id, last\_name, salary, commission\_pct

FROM employees

WHERE job\_id LIKE '%REP%';

1. Update employee 114’s job and salary to match that of employee 205.

UPDATE employees

SET job\_id = (SELECT job\_id

FROM employees

WHERE employee\_id = 205),

salary = (SELECT salary

FROM employees

WHERE employee\_id = 205)

WHERE employee\_id = 114;

1. To create a copy of the employees tablecreate table copy\_emp as

select \* from employees;

1. Updates the COPY\_EMP table by changing the department number of all employees with employee 200’s job ID to employee 100’s current department number.

UPDATE copy\_emp

SET department\_id = (SELECT department\_id

FROM employees

WHERE employee\_id = 100)

WHERE job\_id = (SELECT job\_id

FROM employees

WHERE employee\_id = 200);

1. Delete the department of Payroll

DELETE FROM departments

WHERE department\_name = 'Payroll';

1. To delete the whole table content

DELETE FROM copy\_emp;

1. To delete based on a sub query. This command deletes all employees whose department name includes ‘Rec’.

DELETE FROM employees

WHERE department\_id =

(SELECT department\_id

FROM departments

WHERE department\_name

LIKE '%Rec%');

1. To empty a table

TRUNCATE TABLE copy\_emp;

1. We can update a specific record through insert as follows

INSERT INTO

(SELECT employee\_id, last\_name,

email, hire\_date, job\_id, salary,

department\_id

FROM employees

WHERE department\_id = 50)

VALUES (99999, 'Taylor', 'DTAYLOR',

TO\_DATE ('07-JUN-99', 'DD-MON-RR'),

'ST\_CLERK', 5000, 50);

1. After a DML command issued we can undo or confirm permanently the command by using rollback and commit

DELETE FROM employees

WHERE employee\_id = 99999;

Rollback;

or

Commit;

1. Another example for commit

DELETE FROM departments

WHERE department\_id IN (290, 300);

UPDATE employees

SET department\_id = 80

WHERE employee\_id = 206;

COMMIT;

1. We can insert a save point to be used for commit or rollback

Savepoint step\_10;

## **Class Exercise:**

1. Create the table my\_employee



1. Create an INSERT statement to add the first row of data to the MY\_EMPLOYEE table from the following sample data. Do not list the columns in the INSERT clause.

ID LAST\_NAME FIRST\_NAME USERID SALARY

1- Patel Ralph rpatel 895

1. Dancs Betty bdancs 860

3- Biri Ben bbiri 1100

4- Newman Chad cnewman 750

5- Ropeburn Audrey aropebur 1550

1. Populate the MY\_EMPLOYEE table with the second row of sample data from the preceding list. This time, list the columns explicitly in the INSERT clause.
2. Confirm your addition to the table.
3. Make the data additions permanent.
4. Change the last name of employee 3 to Drexler.
5. Change the salary to $1,000 for all employees with a salary less than $900.
6. Verify your changes to the table.
7. Create a save point.
8. Delete Betty Dancs from the MY\_EMPLOYEE table.
9. Undo the command till the last save point.