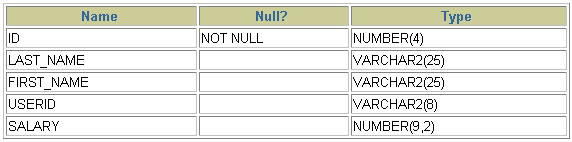
The HR department wants you to create SQL statements to insert, update, and delete employee data. As a prototype, you use the MY\_EMPLOYEE table, prior to giving the statements to the HR department.

**Insert data into the MY\_EMPLOYEE table.**

1. Describe the structure of the MY\_EMPLOYEE table to identify the column names.



--🡪 DESCRIBE my\_employee

2. 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. *Do not enter all rows yet.*



-**-🡪 INSERT INTO my\_employee**

**VALUES (1, ’Patel’, ’Ralph’, ’rpatel’, 895);**

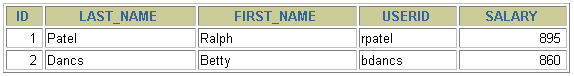
3. Populate the MY\_EMPLOYEE table with the second row of sample data from the preceding list. This time, list the columns explicitly in the INSERT clause.

-🡪 **INSERT INTO my\_employee (id, last\_name, first\_name,**

**userid, salary)**

**VALUES (2, ’Dancs’, ’Betty’, ’bdancs’, 860);**

4. Confirm your addition to the table.



--🡪 **SELECT \***

**FROM my\_employee;**

**ID LAST\_NAME FIRST\_NAME USERID SALARY**

**1 Patel Ralph rpatel 895**

**2 Dancs Betty bdancs 860**

**3 Biri Ben bbiri 1100**

**4 Newman Chad cnewman 750**

**5 Ropeburn Audrey aropebur 1550**

5. Populate the table with the next two rows of sample data by running the insert

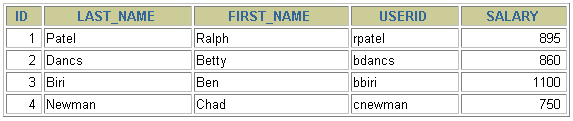
--🡪 **INSERT INTO my\_employee**

**VALUES (&p\_id, ’&p\_last\_name’, ’&p\_first\_name’,**

**lower(substr(’&p\_first\_name’, 1, 1) ||**

**substr(’&p\_last\_name’, 1, 7)), &p\_salary);**

6. Confirm your additions to the table.



-🡪 **SELECT \***

**FROM my\_employee;**

7. Make the data additions permanent.

--🡪 **COMMIT;**

Update and delete data in the MY\_EMPLOYEE table.

8. Change the last name of employee 3 to Drexler.

**--🡪 SET last\_name = ’Drexler’**

**WHERE id = 3;**

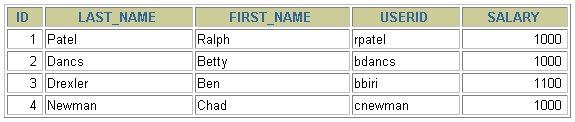
9. Change the salary to $1,000 for all employees who have a salary less than $900.

-🡪 **UPDATE my\_employee**

**SET salary = 1000**

**WHERE salary < 900;**

10. Verify your changes to the table.



**-🡪 SELECT last\_name, salary**

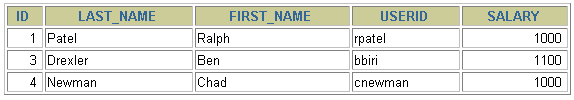
**FROM my\_employee;**

11. Delete Betty Dancs from the MY\_EMPLOYEE table.

-🡪 **SELECT last\_name, salary**

**FROM my\_employee;**

12. Confirm your changes to the table.



**--🡪 SELECT \***

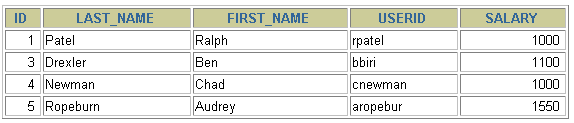
**FROM my\_employee;**

13. Commit all pending changes.

--🡪 **COMMIT;**

Control data transaction to the MY\_EMPLOYEE table.

14. Confirm your addition to the table.



**---🡪 SELECT \***

**FROM my\_employee;**

15. Mark an intermediate point in the processing of the transaction.

**--🡪 SAVEPOINT step\_17;**

16. Empty the entire table.

**-🡪 DELETE**

**FROM my\_employee**;

17. Confirm that the table is empty.

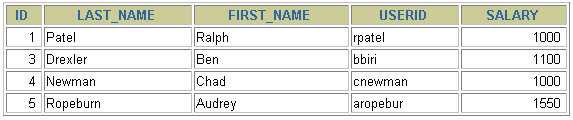
**--🡪 SELECT \***

**FROM my\_employee;**

18. Discard the most recent DELETE operation without discarding the earlier INSERT operation.

--🡪 **ROLLBACK TO step\_17;**

19. Confirm that the new row is still intact.



**--🡪 SELECT \***

**FROM my\_employee;**

20. Make the data addition permanent.

**---🡪 COMMIT;**