1) Write a query for the HR department to produce the addresses of all the departments.Use the LOCATIONS and COUNTRIES tables. Show the location ID, street address, city, state or province, and country in the output. Run the query

=>SELECT location\_id,street\_address,city,state\_province,countries.country\_name FROM locations LEFT JOIN countries ON locations.country\_id=countries.country\_id;

2) The HR department needs a report of employees in Toronto. Display the last name, job, department number, and department name for all employees who work in Toronto.

=> SELECT last\_name,job\_id,employees.department\_id,departments.department\_name,city FROM employees LEFT JOIN departments ON employees.department\_id=departments.department\_id LEFT JOIN locations ON departments.location\_id=locations.location\_id WHERE city='Toronto';

3) Create a report for HR that displays the last name and salary of every employee who reports to King.

=> SELECT last\_name,salary FROM employees WHERE manager\_id=(SELECT employee\_id FROM employees WHERE last\_name='King');

4) Create Table MY\_EMPLOYEE. Description is shown below:

=> CREATE TABLE my\_employee(ID int NOT NULL,LAST\_NAME varchar(25),FIRST\_NAME varchar(25),USERID varchar(8),SALARY int);

5) Add the following data to the MY\_EMPLOYEE table

=>INSERT INTO my\_employee VALUES(1,'patel','Ralph','rpatel',895);

INSERT INTO my\_employee VALUES(2,'Dancs','Betty','bdancs',860);

INSERT INTO my\_employee VALUES(3,'Biri','Ben','bbiri',1100);

INSERT INTO my\_employee VALUES(4,'Newman','Chad','cnewman',750);

INSERT INTO my\_employee VALUES(5,'Ropeburn','Audrey','aropebur',1550);

6) Change the last name of employee 3 to Drexler.

=> UPDATE my\_employee SET LAST\_NAME='Drexler' WHERE ID=3;

7) 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;

8) Delete Betty Dancs from the MY\_EMPLOYEE table.

=> DELETE FROM my\_employee WHERE LAST\_NAME='Dancs';

9) Mark an intermediate point in the processing of the transaction.

=> START TRANSACTION;

SAVEPOINT my\_savepoint;

10) Delete all the rows from the MY\_EMPLOYEE table. Confirm that the table is empty.

=> DELETE FROM my\_employee;

11) Discard the most recent DELETE operation.

=> ROLLBACK TO my\_savepoint;

12) Make the data addition permanent.

=>COMMIT;

13) The staff in the HR department wants to hide some of the data in the EMPLOYEES table. Create a view called EMPLOYEES\_VU based on the employee numbers, employee last names, and department numbers from the EMPLOYEES table. The heading for the employee name should be EMPLOYEE.

Confirm that the view works. Display the contents of the EMPLOYEES\_VU view

=> CREATE VIEW employee\_vu AS SELECT employee\_id,last\_name,department\_id FROM employees;

SELECT \*FROM employee\_vu;