Q1)

Sales

Transaction\_ID | amount | Customer\_ID | Date | num\_units | product\_ID | weeks

Product

Product\_ID | Product\_Name | Product\_Category

**List out the products that don’t have any sales in last two consecutive weeks**

Select p.product\_ID,p.Product\_Name from Sales s inner join Product p on s.product\_ID=p.Product\_ID where week not in (week n, week n-1 )

**List out the products that don’t have any sales in any two consecutive weeks**

Select p.product\_ID,p.Product\_Name,lead(weeks) as l\_week over(partition by Product\_ID order by weeks) from Sales s inner join Product p on s.product\_ID=p.Product\_ID where week - l\_week>2;

Q2)

Write an SQL query to find employee\_id of all employees that directly or indirectly report their work to the head of the company.

The indirect relation between managers will not exceed 3 managers as the company is small.

Return result table in any order without duplicates.

The query result format is in the following example:

Employees table:

+-------------+---------------+------------+

| employee\_id | employee\_name | manager\_id |

+-------------+---------------+------------+

| 1 | Boss | 1 |

| 3 | Alice | 3 |

| 2 | Bob | 1 |

| 4 | Daniel | 2 |

| 7 | Luis | 4 |

| 8 | Jhon | 3 |

| 9 | Angela | 8 |

| 77 | Robert | 1 |

+-------------+---------------+------------+

Output should look like:

+-------------+

| employee\_id |

+-------------+

| 2 |

| 77 |

| 4 |

| 7 |

+-------------+

Explanation:

The head of the company is the employee with employee\_id 1.

The employees with employee\_id 2 and 77 report their work directly to the head of the company.

The employee with employee\_id 4 report his work indirectly to the head of the company 4 --> 2 --> 1.

The employee with employee\_id 7 report his work indirectly to the head of the company 7 --> 4 --> 2 --> 1.

The employees with employee\_id 3, 8 and 9 don't report their work to the head of company directly or indirectly.

Data population:

-- create a table

CREATE TABLE employee (

employee\_id INTEGER PRIMARY KEY,

employee\_name TEXT NOT NULL,

manager\_id INTEGER NOT NULL

);

-- insert some values

INSERT INTO employee VALUES (1, 'Boss', '1');

INSERT INTO employee VALUES (3, 'Alice', '3');

INSERT INTO employee VALUES (2, 'Bob', '1');

INSERT INTO employee VALUES (4, 'Daniel', '2');

INSERT INTO employee VALUES (7, 'Luis', '4');

INSERT INTO employee VALUES (8, 'Jhon', '3');

INSERT INTO employee VALUES (9, 'Angela', '8');

INSERT INTO employee VALUES (77, 'Robert', '1');

-- fetch some values

SELECT \* FROM employee;

Ans) SELECT e1.employee\_id FROM employee e1 inner join employee e2 on e1.manager\_id=e2.employee\_id inner join employee e3 on e2.manager\_id=e3.employee\_id where e3.manager\_id=1 AND e1.employee\_id!=1;

Q3) Find the result of different joins of Table A and Table B

TableA TableB

Id Id

1 7

2 NULL

NULL NULL

3 3

2 4

4 4

4 2

6 2

10 8

Ans)

Inner Join - Select \* from TableA Inner Join TableB on TableA.id=TableB.id;

Left Join - Select \* from TableA Left Join TableB on TableA.id=TableB.id;

Right Join - Select \* from TableA Right Join TableB on TableA.id=TableB.id;

Full Join(Outer Join) - Select \* from TableA Full Join TableB on TableA.id=TableB.id

Left Anti Join - Select \* from TableA Left Join TableB on TableA.id=TableB.id Where Table.id IS NULL;

Q4)

CREATE TABLE CUSTOMERS2(

CUSTOMER\_ID INT,

Date1 VARCHAR(6)

);

INSERT INTO CUSTOMERS2 VALUES(1111,'202212');

INSERT INTO CUSTOMERS2 VALUES(1111,'202210');

INSERT INTO CUSTOMERS2 VALUES(1111,'202209');

INSERT INTO CUSTOMERS2 VALUES(1111,'202301');

INSERT INTO CUSTOMERS2 VALUES(2222,'202201');

INSERT INTO CUSTOMERS2 VALUES(2222,'202205');

INSERT INTO CUSTOMERS2 VALUES(2222,'202204');

WITH CTE1 AS(

SELECT CUSTOMER\_ID, DATE1, MIN(DATE1) OVER(PARTITION BY CUSTOMER\_ID) AS DATE2 FROM CUSTOMERS2)

SELECT CUSTOMER\_ID, DATEDIFF(month,LEFT(DATE2, 4) + '-' + RIGHT(DATE2, 2) + '-01', LEFT(DATE1, 4) + '-' + RIGHT(DATE1, 2) + '-01') AS DIFF\_MONTHS FROM CTE1;

Above Query works only in sql server.