Q1:

-- Table: Employee

CREATE TABLE Employee (

ID INT PRIMARY KEY, -- Unique Identifier for Employee

person\_name VARCHAR(100) NOT NULL, -- Employee's Name

street VARCHAR(100), -- Street Address

city VARCHAR(100) -- City

);

-- Table: Company

CREATE TABLE Company (

ID INT, -- Unique Identifier for Company

company\_name VARCHAR(100), -- Company Name

city VARCHAR(100), -- City

PRIMARY KEY (ID, company\_name) -- Composite Primary Key

);

-- Table: Works

CREATE TABLE Works (

ID INT PRIMARY KEY, -- Unique Identifier for Employee (References Employee)

company\_name VARCHAR(100) NOT NULL, -- Company Name (References Company)

salary DECIMAL(10, 2) NOT NULL, -- Salary

CONSTRAINT fk\_works\_employee FOREIGN KEY (ID) REFERENCES Employee(ID)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT fk\_works\_company FOREIGN KEY (ID, company\_name) REFERENCES Company(ID, company\_name)

ON DELETE CASCADE ON UPDATE CASCADE

);

-- Table: Manages

CREATE TABLE Manages (

ID INT PRIMARY KEY, -- Unique Identifier for Employee (References Employee)

manager\_id INT NOT NULL, -- ID of the Manager (References Employee)

CONSTRAINT fk\_manages\_employee FOREIGN KEY (ID) REFERENCES Employee(ID)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT fk\_manages\_manager FOREIGN KEY (manager\_id) REFERENCES Employee(ID)

ON DELETE CASCADE ON UPDATE CASCADE

);

Q2:

A:

SELECT DISTINCT D.ID

FROM depositor D

WHERE D.ID NOT IN (

SELECT B.ID

FROM borrower B

);

B:

SELECT C2.ID

FROM customer C1

JOIN customer C2 ON C1.customer\_street = C2.customer\_street

AND C1.customer\_city = C2.customer\_city

WHERE C1.ID = '12345' AND C2.ID <> '12345';

C:

SELECT DISTINCT A.branch\_name

FROM account A

JOIN depositor D ON A.account\_number = D.account\_number

JOIN customer C ON D.ID = C.ID

WHERE C.customer\_city = 'Harrison';

Q3:

A:

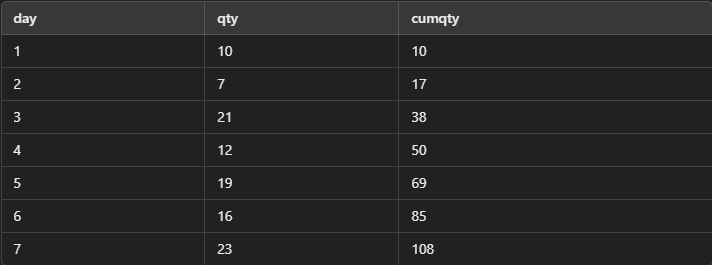
SELECT

day,

qty,

SUM(qty) OVER (ORDER BY day) AS cumqty

FROM demand;



B:

WITH RankedDemand AS (

SELECT

product,

day,

qty,

RANK() OVER (PARTITION BY product ORDER BY qty ASC) AS RN

FROM demand

)

SELECT

product,

day,

qty,

RN

FROM RankedDemand

WHERE RN <= 2;

