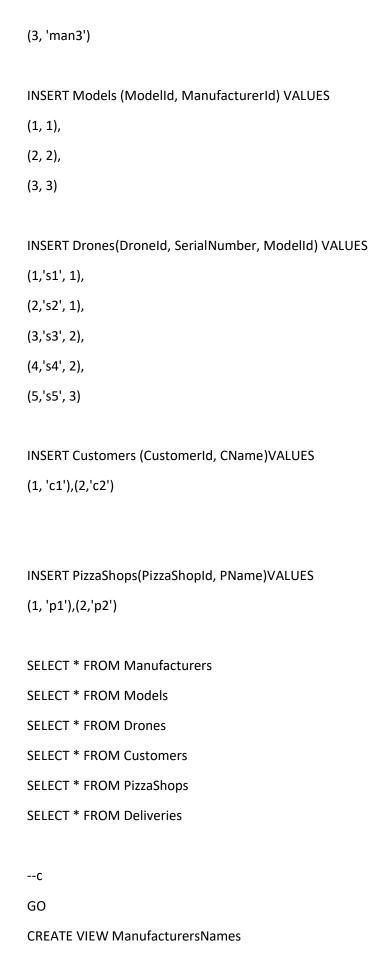
```
CREATE DATABASE Lab13
GO
USE Lab13
GO
DROP TABLE Deliveries;
DROP TABLE PizzaShops
DROP TABLE Customers
DROP TABLE Drones
DROP TABLE Models
DROP TABLE Manufacturers
GO
--a
CREATE TABLE Manufacturers(
ManufacturerId INT PRIMARY KEY,
MName VARCHAR(60)
);
CREATE TABLE Models(
Modelld INT PRIMARY KEY,
MName VARCHAR(60),
MBaterry INT,
MSpeed INT,
ManufacturerId INT REFERENCES Manufacturers(ManufacturerId)
);
CREATE TABLE Drones(
Droneld INT PRIMARY KEY,
SerialNumber VARCHAR(60) UNIQUE,
ModelId INT REFERENCES Models(ModelId)
);
```

```
CREATE TABLE Customers(
CustomerId INT PRIMARY KEY,
CName VARCHAR(60) UNIQUE,
CScore INT
);
CREATE TABLE PizzaShops(
PizzaShopId INT PRIMARY KEY,
PName VARCHAR(60),
PAddress INT
);
CREATE TABLE Deliveries(
DeliveryId INT PRIMARY KEY IDENTITY(1,1),
PizzaShipId INT REFERENCES PizzaShops(PizzaShopId),
CustomerId INT REFERENCES Customers(CustomerId),
Droneld INT REFERENCES Drones(Droneld),
DDateTime datetime
);
GO
--b
GO
CREATE OR ALTER PROC uspAddDelivery(@CustomerName VARCHAR(60), @PizzaShopName
VARCHAR(60), @SerialNumber VARCHAR(60),
@DDateTime DATETIME)
AS
DECLARE @CID INT, @PID INT, @DID INT
```

```
IF NOT EXISTS (SELECT * FROM Customers WHERE Cname = @CustomerName)
BEGIN
RAISERROR('Invalid customer name.', 16, 1)
RETURN
END
IF NOT EXISTS (SELECT * FROM PizzaShops WHERE Pname = @PizzaShopName)
BEGIN
RAISERROR('Invalid pizza shop name.', 16, 1)
RETURN
END
IF NOT EXISTS (SELECT * FROM Drones WHERE SerialNumber = @SerialNumber)
BEGIN
RAISERROR('Invalid drone serial number.', 16, 1)
RETURN
END
SELECT @CID = (SELECT CustomerId FROM Customers WHERE CName = @CustomerName),
@PID = (SELECT PizzaShopId FROM PizzaShops WHERE PName = @PizzaShopName),
@DID = (SELECT Droneld FROM Drones WHERE SerialNumber = @SerialNumber)
INSERT Deliveries (CustomerId, PizzaShipId, DroneId, DDateTime) VALUES(@CID, @PID, @DID,
@DDateTime)
GO
EXEC uspAddDelivery 'c3', 'p1', 's5', '1-1-2020'
INSERT Manufacturers (ManufacturerId, MName) VALUES
(1, 'man1'),
(2, 'man2'),
```



```
SELECT M.MName
FROM(
SELECT M.ManufacturerId, COUNT(*) AS 'NoDrones'
FROM Models M INNER JOIN Drones D ON M.ModelId = D.ModelId
GROUP BY M.ManufacturerId
) MD INNER JOIN Manufacturers M ON MD.ManufacturerId = M.ManufacturerId
WHERE MD.NoDrones =
(SELECT MAX(Aux.NoDrones)
FROM (
SELECT M.ManufacturerId, COUNT(*) AS 'NoDrones'
FROM Models M INNER JOIN Drones D ON M.Modelid = D.Modelid
GROUP BY M.ManufacturerId
) Aux
)
GO
SELECT * FROM ManufacturersNames
--d
GO
CREATE FUNCTION atLeastDDeliveries (@D INT)
RETURNS TABLE
AS
RETURN
SELECT C.CName
FROM Customers C INNER JOIN Deliveries D ON C.CustomerId = D.CustomerId
GROUP BY C.CName
HAVING COUNT(*) >= @D
```

AS

GO

SELECT * FROM atLeastDDeliveries(1)

SELECT * FROM atLeastDDeliveries(2)

SELECT * FROM atLeastDDeliveries(3)

