

```
1 • create database litmus7;
2 • use litmus7;
3
4 • CREATE TABLE Cutomer(
5     CustomerID INT PRIMARY KEY,
6     Name VARCHAR(100),
7     Email VARCHAR(100),
8     City VARCHAR(100),
9     SignupDate DATE );
10
11 • RENAME TABLE Cutomer TO Customers;
12
13 • CREATE TABLE Orders(
14     OrderID INT PRIMARY KEY,
15     CustomerID INT,
16     OrderDate DATE,
17     TotalAmount DECIMAL(10,2),
18     FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID));
19
20 • CREATE TABLE Products(
21     ProductID INT PRIMARY KEY,
22     ProductName VARCHAR(100),
23     Category VARCHAR(50),
24     Price DECIMAL(10,2));
25
```

25

```
26 • CREATE TABLE OrderDetails(  
27     OrderDetailID INT PRIMARY KEY,  
28     OrderID INT,  
29     ProductID INT,  
30     Quantity INT,  
31     Price DECIMAL(10,2),  
32     FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),  
33     FOREIGN KEY (ProductID) REFERENCES Products(ProductID));  
34  
35
```

```
36 • INSERT INTO Customers (CustomerID, Name, Email, City, SignupDate) VALUES  
37     (1, 'Alan', 'alan@ex.com', 'Kochi', '2024-01-15'),  
38     (2, 'Bob', 'bob@ex.com', 'Mumbai', '2023-02-20'),  
39     (3, 'Donna', 'donna@ex.com', 'Mumbai', '2024-03-10'),  
40     (4, 'Sharath', 'sharath@ex.com', 'Delhi', '2023-03-10');  
41  
42
```

```
43 • INSERT INTO Products (ProductID, ProductName, Category, Price) VALUES  
44     (201, 'Laptop', 'Electronics', 45000.00),  
45     (202, 'Mouse', 'Electronics', 500.00),  
46     (203, 'Notebook', 'Stationary', 100.00),  
47     (204, 'Water Bottle', 'School', 250.00),  
48     (205, 'Desk Lamp', 'Furniture', 1200.00),  
49     (206, 'Smartwatch', 'Electronics', 9999.00);  
50  
51
```

```
52 • INSERT INTO Orders (OrderID, CustomerID, OrderDate, TotalAmount) VALUES
53     (101, 1, '2024-05-15', 800.00),      -- 1 Mouse and 3 Notebook
54     (102, 2, '2024-06-01', 46000.00),   -- 1 Laptop 4 Water Bottle
55     (103, 1, '2024-06-05', 1000.00),    -- 2 Mouse
56     (104, 3, '2024-04-22', 1700.00);    -- 5 Notebook and 1 Desk Lamp
57
58
59 • INSERT INTO OrderDetails (OrderDetailID, OrderID, ProductID, Quantity, Price) VALUES
60     (301, 101, 202, 1, 500.00),         -- Mouse
61     (302, 101, 203, 3, 100.00),         -- Notebook
62     (303, 102, 201, 1, 45000.00),       -- Laptop
63     (304, 102, 204, 4, 250.00),         -- Water Bottle
64     (305, 103, 202, 2, 500.00),         -- Mouse
65     (306, 104, 203, 5, 100.00),         -- Notebook
66     (307, 104, 205, 1, 1200.00);       -- Desk Lamp
67
```

```
68      #list of all customers
69  ●    SELECT * FROM CUSTOMERS;
```

Result Grid



Filter Rows:

Edit:



Export/Import:



	CustomerID	Name	Email	City	SignupDate
▶	1	Alan	alan@ex.com	Kochi	2024-01-15
	2	Bob	bob@ex.com	Mumbai	2023-02-20
	3	Donna	donna@ex.com	Mumbai	2024-03-10
	4	Sharath	sharath@ex.com	Delhi	2023-03-10
●	NULL	NULL	NULL	NULL	NULL

```
68      #list of all customers
69  •    SELECT * FROM Orders where OrderDate <= 2025-05-09;
70  ✖    result is null since i forgot to add a corresponding entry
```

Result Grid |   Filter Rows: | Edit:    | Export/Import:   | Wrap Cell Content: 

	OrderID	CustomerID	OrderDate	TotalAmount
•	NULL	NULL	NULL	NULL

```
68      #product name and prices
```

```
69  •    SELECT ProductName , Price from Products;
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	ProductName	Price
▶	Laptop	45000.00
	Mouse	500.00
	Notebook	100.00
	Water Bottle	250.00
	Desk Lamp	1200.00
	Smartwatch	9999.00

```
69 • SELECT count(Category),Category FROM Products GROUP BY Category;
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	count(Category)	Category
▶	3	Electronics
	1	Stationary
	1	School
	1	Furniture

```
68 #customers from mumbai
69 * SELECT * FROM Customers where City = 'Mumbai';
70
```

Result Grid



Filter Rows:

Edit:



Export/Import:



Wrap Cell Content:



	CustomerID	Name	Email	City	SignupDate
▶	2	Bob	bob@ex.com	Mumbai	2023-02-20
	3	Donna	donna@ex.com	Mumbai	2024-03-10
•	NULL	NULL	NULL	NULL	NULL


```
68      #orders with total amount > 5000
69  •    SELECT * FROM Orders where TotalAmount > 5000;
70
```

Result Grid



Filter Rows:

Edit:



Exp

	OrderID	CustomerID	OrderDate	TotalAmount
▶	102	2	2024-06-01	46000.00
•	NULL	NULL	NULL	NULL

```
68      #customers signing up after 2024-01-01
69  *    SELECT * FROM Customers where SignupDate > '2024-01-01';
70
```

Result Grid



Filter Rows:





Edit:



Export/Import

	CustomerID	Name	Email	City	SignupDate
▶	1	Alan	alan@ex.com	Kochi	2024-01-15
	3	Donna	donna@ex.com	Mumbai	2024-03-10
●	NULL	NULL	NULL	NULL	NULL

```
69      #orders with customer names
70  •    SELECT o.CustomerID, o.OrderID, c.Name
71      from Orders o
72      join Customers c on c.CustomerID = o.CustomerID;
73
```

Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

	CustomerID	OrderID	Name
▶	1	101	Alan
	1	103	Alan
	2	102	Bob
	3	104	Donna

```

69      #products bought in order (assuming date)
70  *  SELECT od.OrderID, o.OrderDate, p.ProductName, od.Quantity, od.Price
71      FROM OrderDetails od
72      JOIN Orders o ON od.OrderID = o.OrderID
73      JOIN Products p ON od.ProductID = p.ProductID
74      ORDER BY o.OrderDate;

```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	OrderID	OrderDate	ProductName	Quantity	Price
►	104	2024-04-22	Notebook	5	100.00
	104	2024-04-22	Desk Lamp	1	1200.00
	101	2024-05-15	Mouse	1	500.00
	101	2024-05-15	Notebook	3	100.00
	102	2024-06-01	Laptop	1	45000.00
	102	2024-06-01	Water Bottle	4	250.00
	103	2024-06-05	Mouse	2	500.00

```
69      #customers who didnt place order
70  *   SELECT c.Name , c.CustomerID
71      FROM Customers c
72      LEFT JOIN Orders o on o.CustomerID = c.CustomerID
73      WHERE o.OrderID is NULL;
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	Name	CustomerID
▶	Sharath	4

```
69      #total amt spent by customer
70  *   SELECT c.Name , sum(o.TotalAmount) as Expenditure
71      FROM Customers c
72      JOIN Orders o on o.CustomerID = c.CustomerID
73      GROUP BY c.Name;
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	Name	Expenditure
▶	Alan	1800.00
	Bob	46000.00
	Donna	1700.00

```
69      #product sold the most
70  •    SELECT p.ProductName, sum(od.Quantity) as MostSold
71      FROM OrderDetails od
72      JOIN Products p ON p.ProductID = od.ProductID
73      GROUP BY p.ProductID, p.ProductName
74      ORDER BY MostSold DESC
75      LIMIT 1;
76
77
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:

	ProductName	MostSold
▶	Notebook	8

```
69      #average order value
70  •   SELECT c.Name, avg(o.TotalAmount) as AverageOrderExpense
71      FROM Customers c
72      JOIN Orders o ON o.CustomerID = c.CustomerID
73      GROUP BY c.Name;
74
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	Name	AverageOrderExpense
▶	Alan	900.000000
	Bob	46000.000000
	Donna	1700.000000


```

69 • SELECT SUM(od.Price * od.Quantity) , p.Category
70 FROM OrderDetails od
71 JOIN Products p ON p.ProductID = od.ProductID
72 group by p.Category;

```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	SUM(od.Price * od.Quantity)	Category
▶	46500.00	Electronics
	800.00	Stationary
	1000.00	School
	1200.00	Furniture

```

69      #customers with spending > avg spending
70  *  SELECT c.CustomerID, c.Name, SUM(od.Quantity * od.Price) AS Expense
71      FROM Customers c
72      JOIN Orders o ON o.CustomerID = c.CustomerID
73      JOIN OrderDetails od ON od.OrderID = o.OrderID
74      GROUP BY c.Name , c.CustomerID
75  *  having Expense > (
76      SELECT avg(CustomerTotal)
77  *  FROM (
78      SELECT SUM(od.Quantity * od.Price) as CustomerTotal
79      FROM Orders o
80      JOIN OrderDetails od on od.OrderID = o.OrderID
81      GROUP BY o.CustomerID
82      )as CustomerTotals
83  );

```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	CustomerID	Name	Expense
▶	2	Bob	46000.00

```
68      #products not ordered
69  •    SELECT ProductID, ProductName
70      FROM Products
71  ○    WHERE ProductID NOT IN (
72          SELECT ProductID
73          FROM OrderDetails
74      );
75
```

Result Grid



Filter Rows:

Edit:



	ProductID	ProductName
▶	206	Smartwatch
●	NULL	NULL

```

68      #most recent order for each customer
69  •   SELECT  o.CustomerID , o.OrderDate , o.OrderID
70      from Orders o
71  •   where o.OrderDate = (
72          select max(o1.OrderDate)
73          from Orders o1
74          where o1.CustomerID = o.CustomerID);
75

```

Result Grid					Filter Rows:		Edit:			Export/
	CustomerID	OrderDate	OrderID							
▶	2	2024-06-01	102							
	1	2024-06-05	103							
	3	2024-04-22	104							
●	NULL	NULL	NULL							

```

68 #customer spending rank
69 * SELECT
70     c.CustomerID,
71     c.Name,
72     SUM(od.Quantity * od.Price) AS TotalSpent,
73     RANK() OVER (ORDER BY SUM(od.Quantity * od.Price) DESC) as SpendRank
74 FROM Customers c
75 JOIN Orders o ON o.CustomerID = c.CustomerID
76 JOIN OrderDetails od ON od.OrderID = o.OrderID
77 GROUP BY c.CustomerID, c.Name;
78

```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:





	CustomerID	Name	TotalSpent	SpendingRank
▶	2	Bob	46000.00	1
	1	Alan	1800.00	2
	3	Donna	1700.00	3

```

68      #customer spending rank
69  •  SELECT c.CustomerID, c.Name, SUM(od.Quantity * od.Price) as Spent, RANK() OVER (ORDER BY SUM(od.Quantity * od.Price) DESC) as SpendRank
70      FROM Customers c
71      JOIN Orders o ON o.CustomerID = c.CustomerID
72      JOIN OrderDetails od ON od.OrderID = o.OrderID
73      GROUP BY c.CustomerID, c.Name;

```




Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	CustomerID	Name	Spent	SpendRank
▶	2	Bob	46000.00	1
	1	Alan	1800.00	2
	3	Donna	1700.00	3

```

68      #top 3 customers based on no of orders placed
69  *   SELECT *
70  *   FROM (
71      SELECT c.Name , count(o.OrderID) ,RANK() OVER (ORDER BY count(o.CustomerID) DESC ) as OrderRank
72      FROM Orders o
73      JOIN Customers c on c.CustomerID = o.CustomerID
74      GROUP BY c.Name)
75      ranked where OrderRank <= 3;

```

Result Grid   Filter Rows: | Export:  | Wrap Cell Content: 

	Name	count(o.OrderID)	OrderRank
▶	Alan	2	1
	Bob	1	2
	Donna	1	2

```
67
68 #unique customers for unique product
69 * SELECT p.ProductID, p.ProductName, count(distinct o.CustomerID) as UniqueCustomers
70 FROM OrderDetails od
71 JOIN Orders o ON od.OrderID = o.OrderID
72 JOIN Products p ON od.ProductID = p.ProductID
73 GROUP BY p.ProductID, p.ProductName;
74
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	ProductID	ProductName	UniqueCustomers
▶	201	Laptop	1
	202	Mouse	1
	203	Notebook	2
	204	Water Bottle	1
	205	Desk Lamp	1

Result 9

