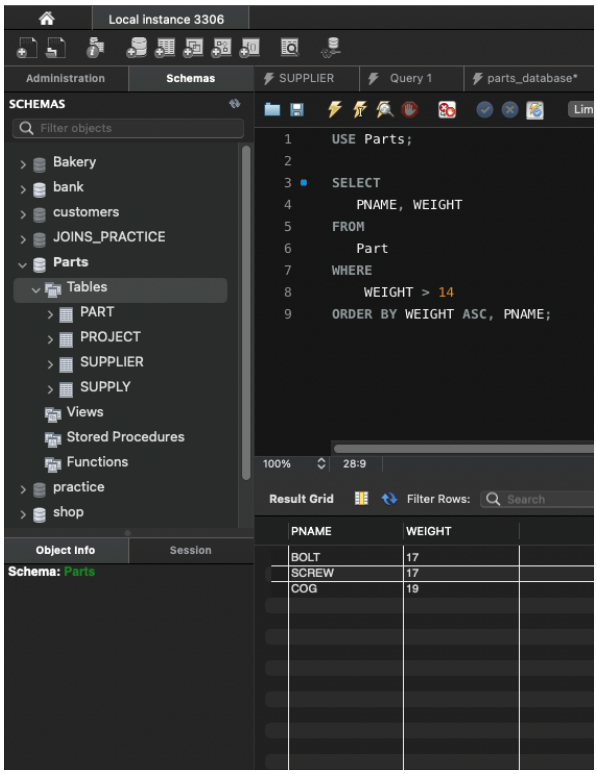
CFG-NANODEGREE
FULL-STACK
SUBMITTED TO:
INSTRUCTOR: MR. SCOTT ADAMS

PAULA MANESE-FULL STACK CLASS

HOMEWORK WEEK 1 - SQL

USE PARTS DB TO WRITE THE FOLLOWING QUERIES

1. Find the name of each part where the weight is more than 14.

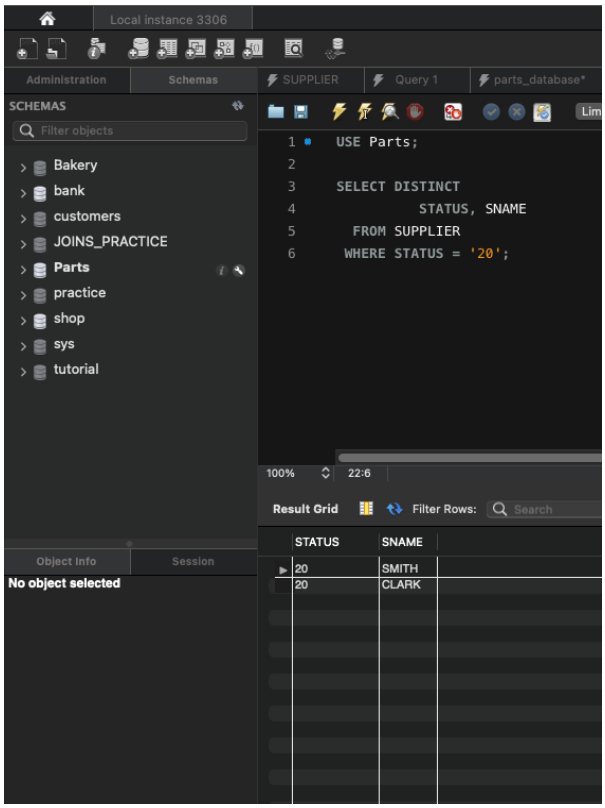


ANSWER:

```
USE Parts;

SELECT
  PNAME, WEIGHT
FROM
  Part
WHERE
   WEIGHT > 14
ORDER BY WEIGHT ASC, PNAME;
```

2. Find all **unique** supplier(s) where their status is equal to 20.



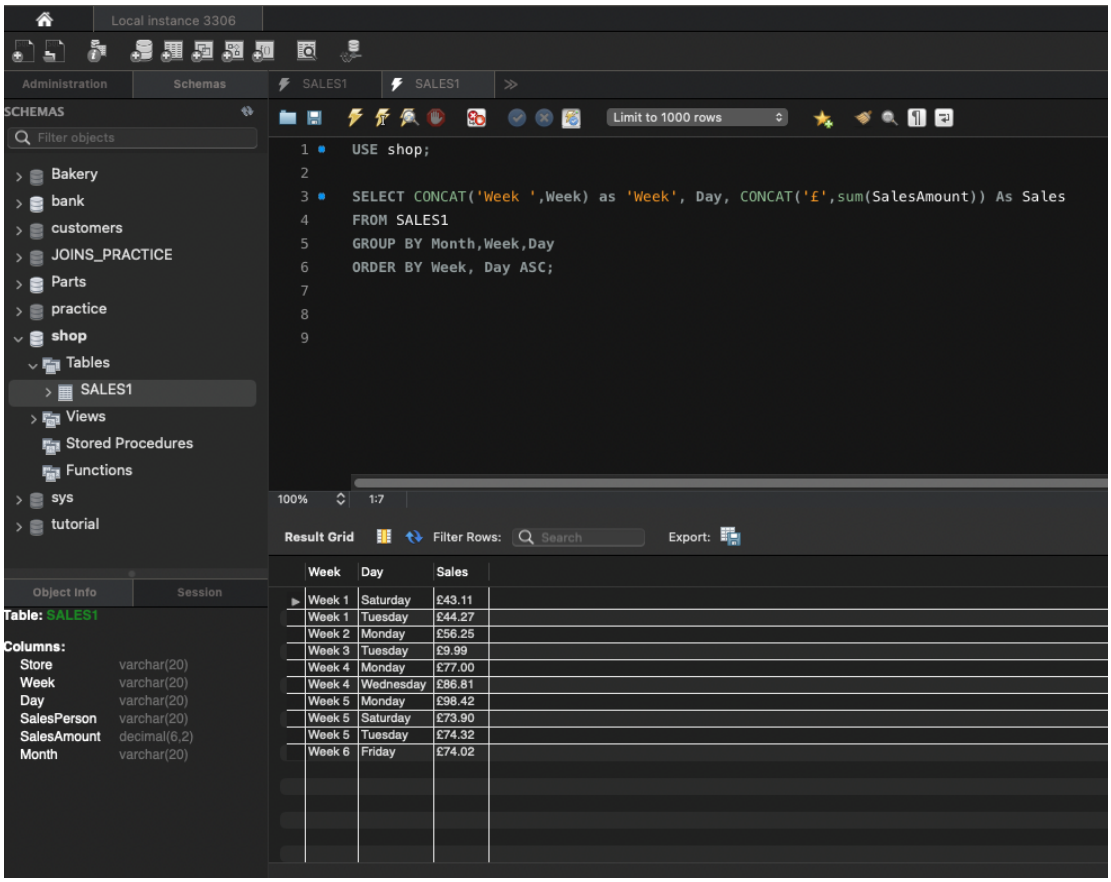
ANSWER:

```
USE Parts;
SELECT DISTINCT
        STATUS, SNAME
  FROM SUPPLIER
 WHERE STATUS = '20';
```

TASK 2

USE SHOP SALES DB TO WRITE THE FOLLOWING QUERIES

1. Find out how many sales (amount) were recorded each week, per day (where available)
○ **This would look like:**
Week 1, Tuesday, £x
Week 1, Wednesday, £x
Week 2, Monday, £x
Week 2, Friday, £x

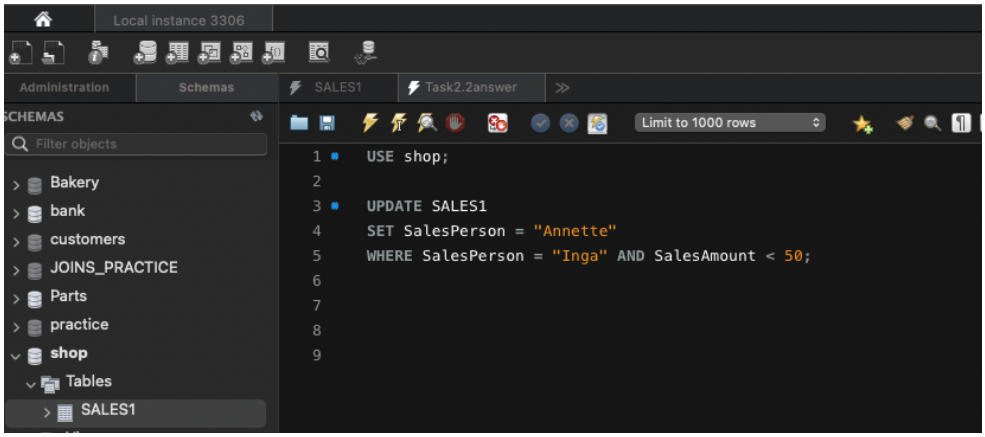


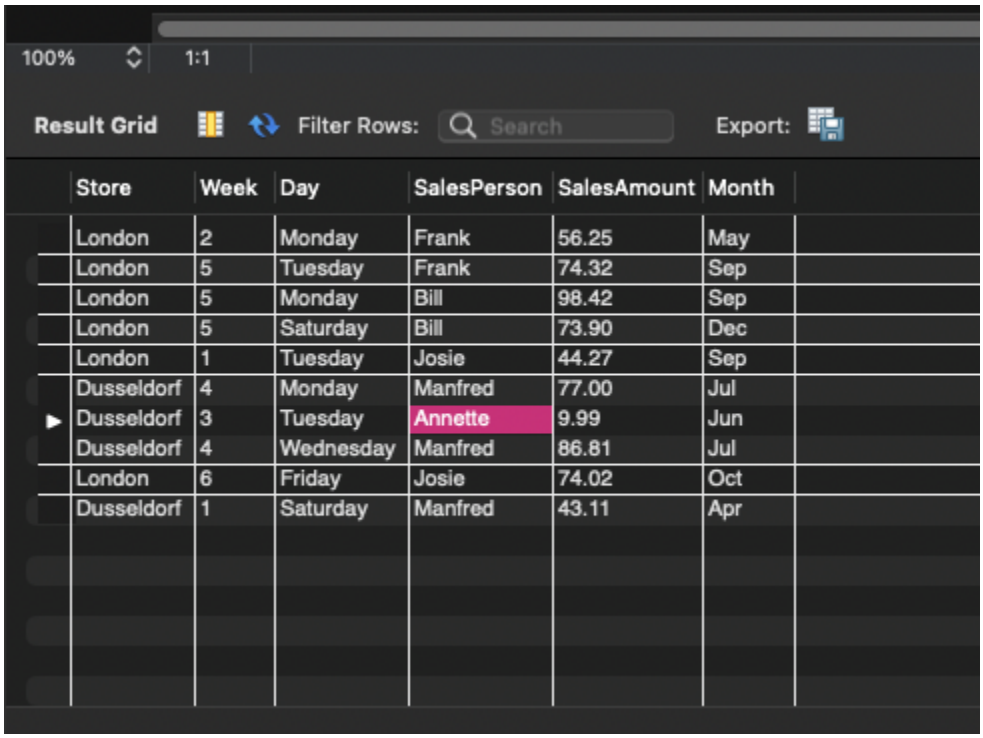
ANSWER:

USE shop;

SELECT CONCAT('Week ',Week) as 'Week', Day, CONCAT('£',sum(SalesAmount)) As Sales
FROM SALES1
GROUP BY Month,Week,Day
ORDER BY Week, Day ASC;

2. Change the name of salesperson Inga to be Annette instead, but only where Ignas Sales are <50.





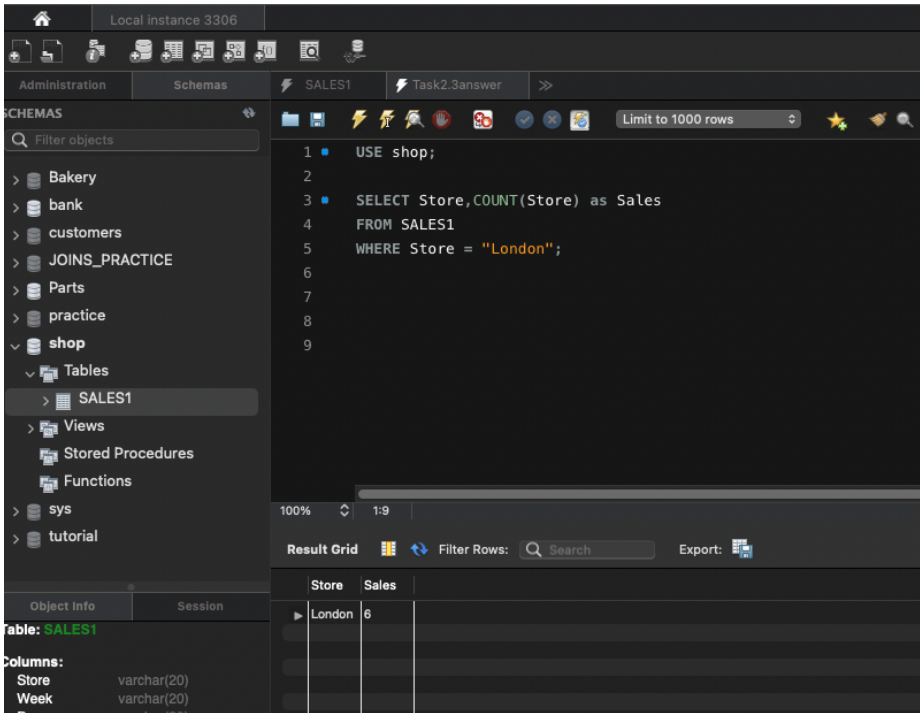
Store	Week	Day	SalesPerson	SalesAmount	Month
London	2	Monday	Frank	56.25	May
London	5	Tuesday	Frank	74.32	Sep
London	5	Monday	Bill	98.42	Sep
London	5	Saturday	Bill	73.90	Dec
London	1	Tuesday	Josie	44.27	Sep
Dusseldorf	4	Monday	Manfred	77.00	Jul
Dusseldorf	3	Tuesday	Annette	9.99	Jun
Dusseldorf	4	Wednesday	Manfred	86.81	Jul
London	6	Friday	Josie	74.02	Oct
Dusseldorf	1	Saturday	Manfred	43.11	Apr

ANSWER:

```
USE shop;

UPDATE SALES1
SET SalesPerson = "Annette"
WHERE SalesPerson = "Inga" AND SalesAmount < 50;
```

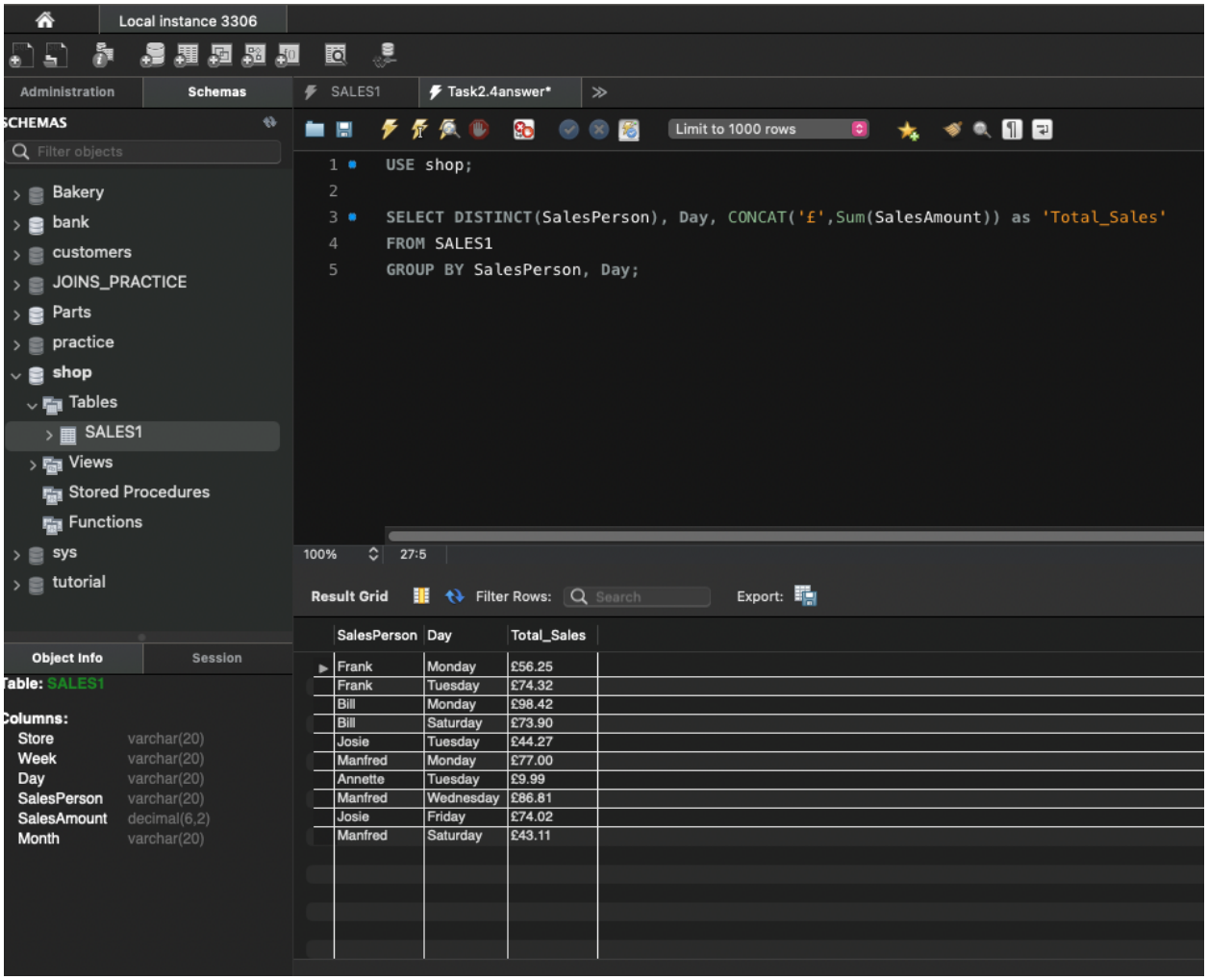
3.Find out how many sales the London office logged
○ Note we're looking for quantity here -
(e.g. if London did 6 sales, then output would be 6)



ANSWER:
USE shop;

SELECT Store,COUNT(Store) as Sales
FROM SALES1
WHERE Store = "London";

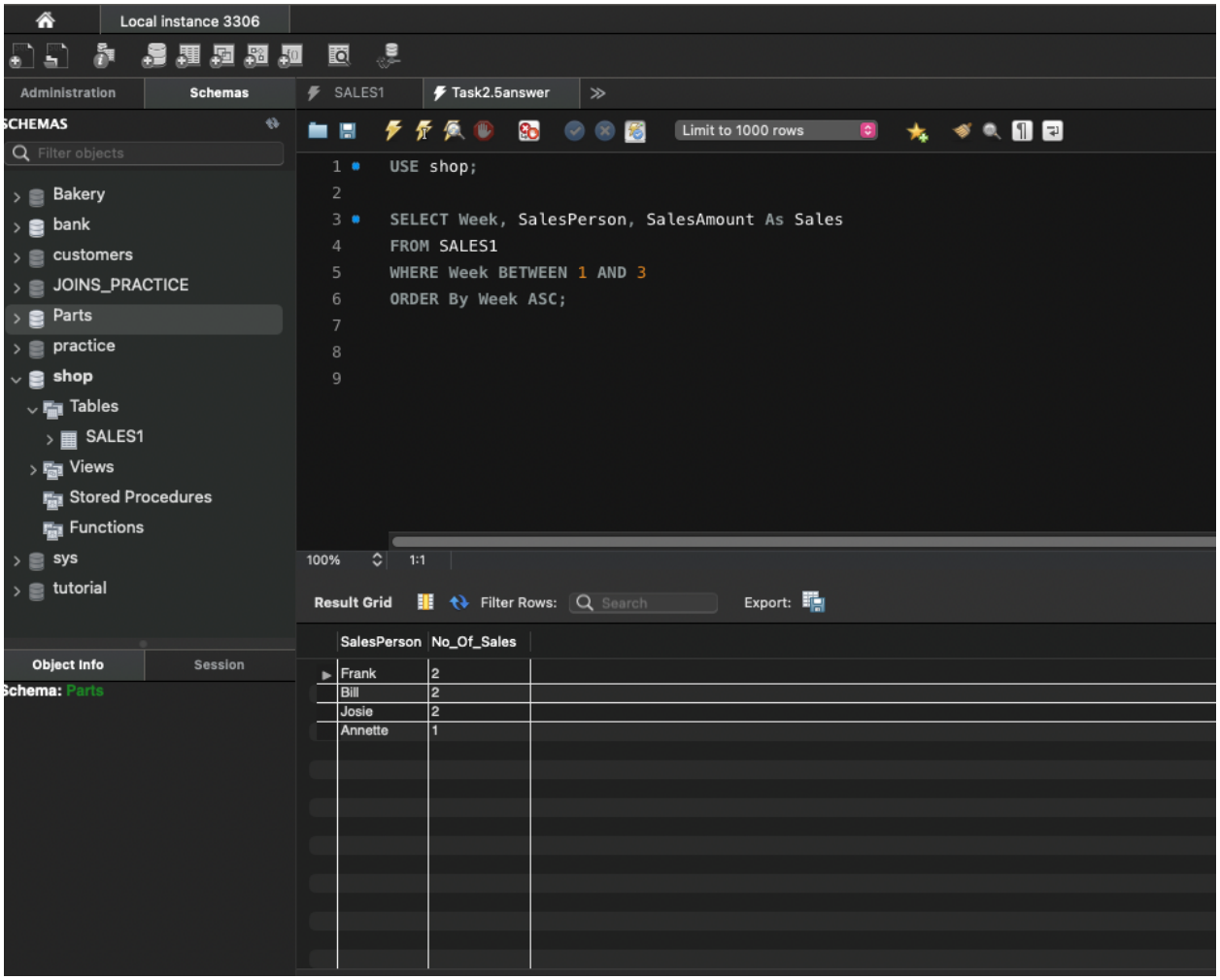
4.Find the total (sum) sales amount by each person by day.



ANSWER:

```
USE shop;
SELECT DISTINCT(SalesPerson), Day, CONCAT('£',Sum(SalesAmount)) as 'Total_Sales'
FROM SALES1
GROUP BY SalesPerson, Day;
```

5. How much (sum) each person sold for between week 1 and week 3.

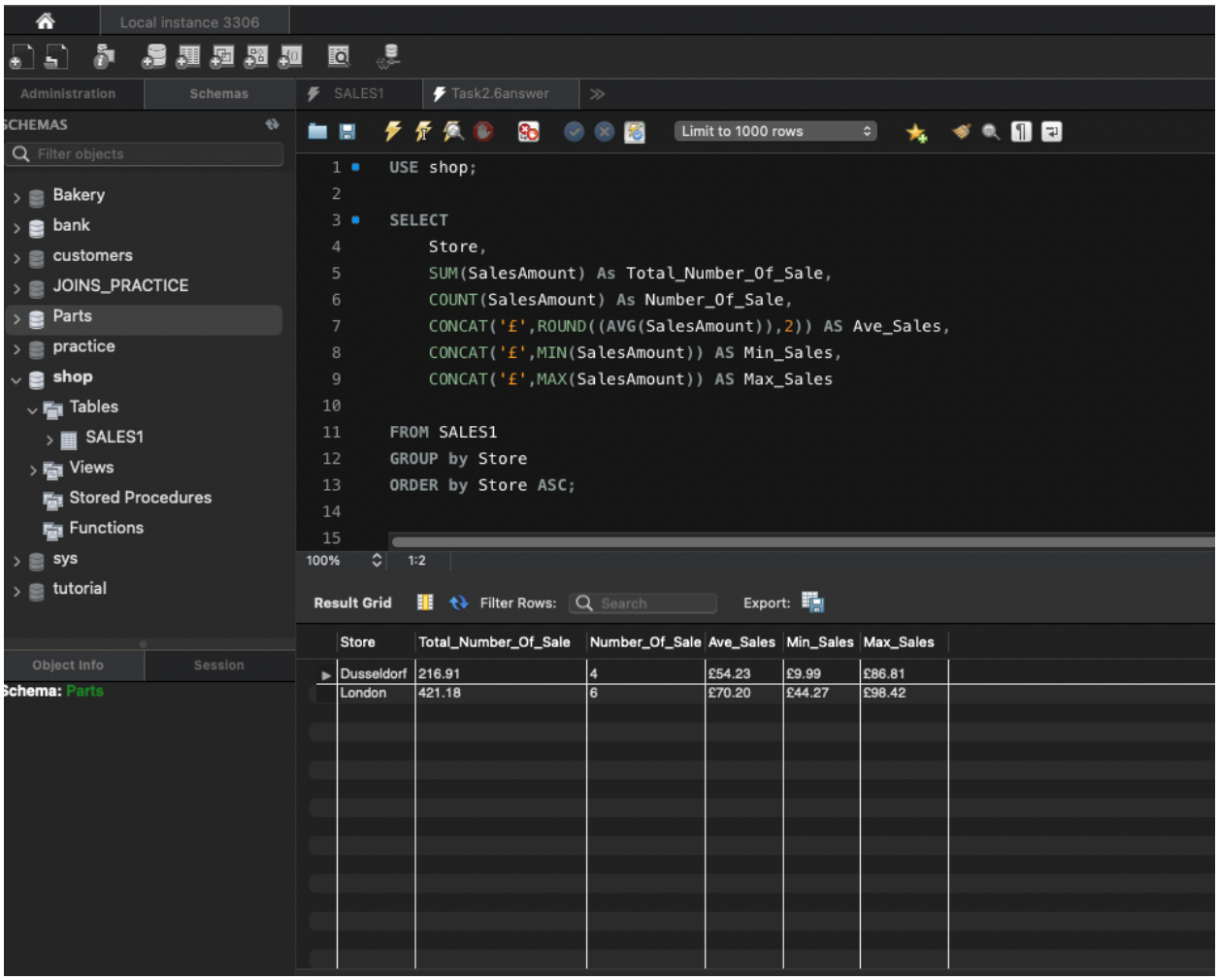


ANSWER:
USE shop;

SELECT Week, SalesPerson, SalesAmount As Sales
FROM SALES1
WHERE Week BETWEEN 1 AND 3
ORDER By Week ASC;

6.For each store:

- The total of their sales;
- The number of sales;
- Their average sales;
- Their lowest sales amount;
- Their highest sales amount.

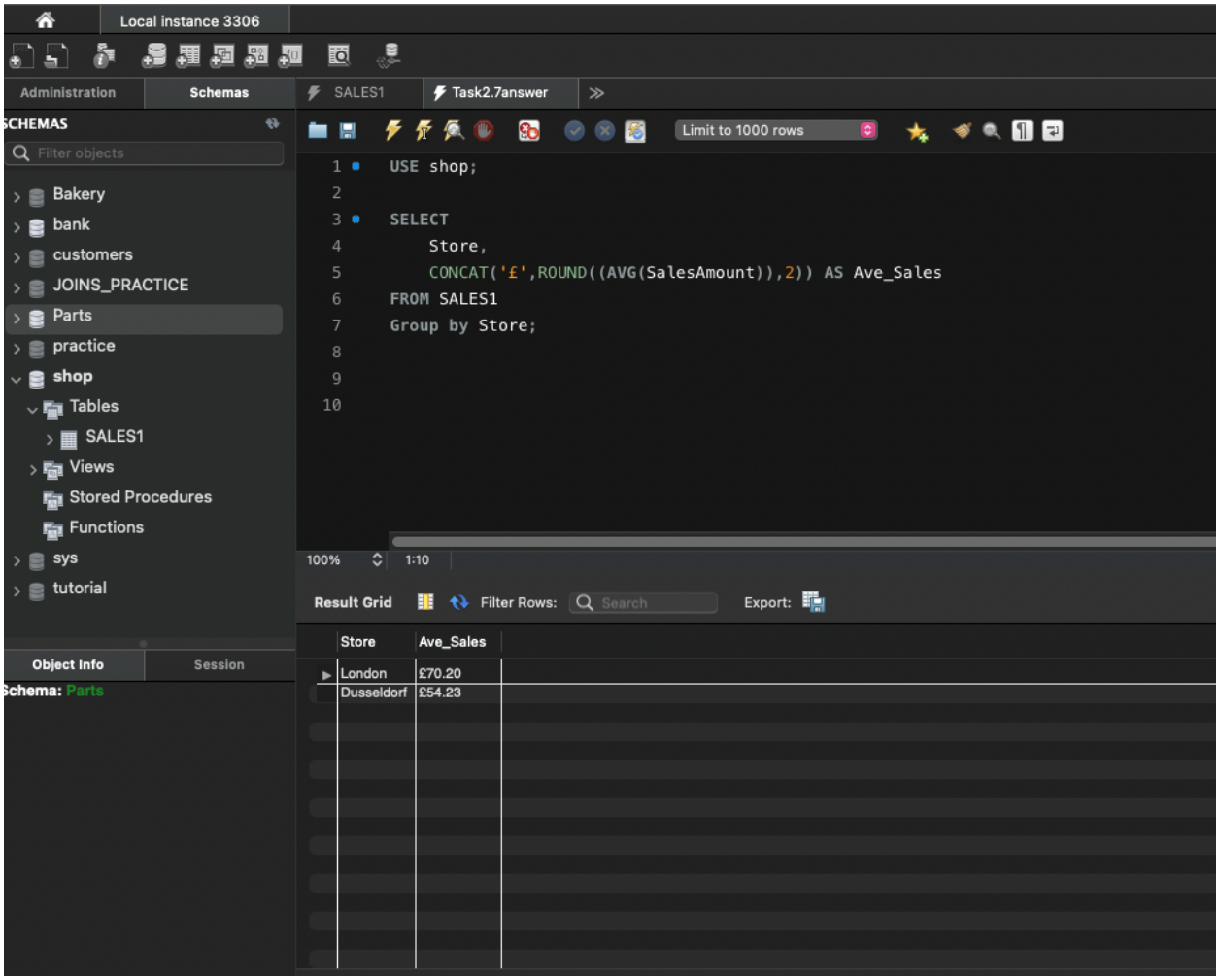



```
USE shop;

SELECT
    Store,
    SUM(SalesAmount) As Total_Number_Of_Sale,
    COUNT(SalesAmount) As Number_Of_Sale,
    CONCAT('£',ROUND((AVG(SalesAmount)),2)) AS Ave_Sales,
    CONCAT('£',MIN(SalesAmount)) AS Min_Sales,
    CONCAT('£',MAX(SalesAmount)) AS Max_Sales

FROM SALES1
GROUP by Store
ORDER by Store ASC;
```

7. Find the average (AVG) monetary sales amount achieved by each store.

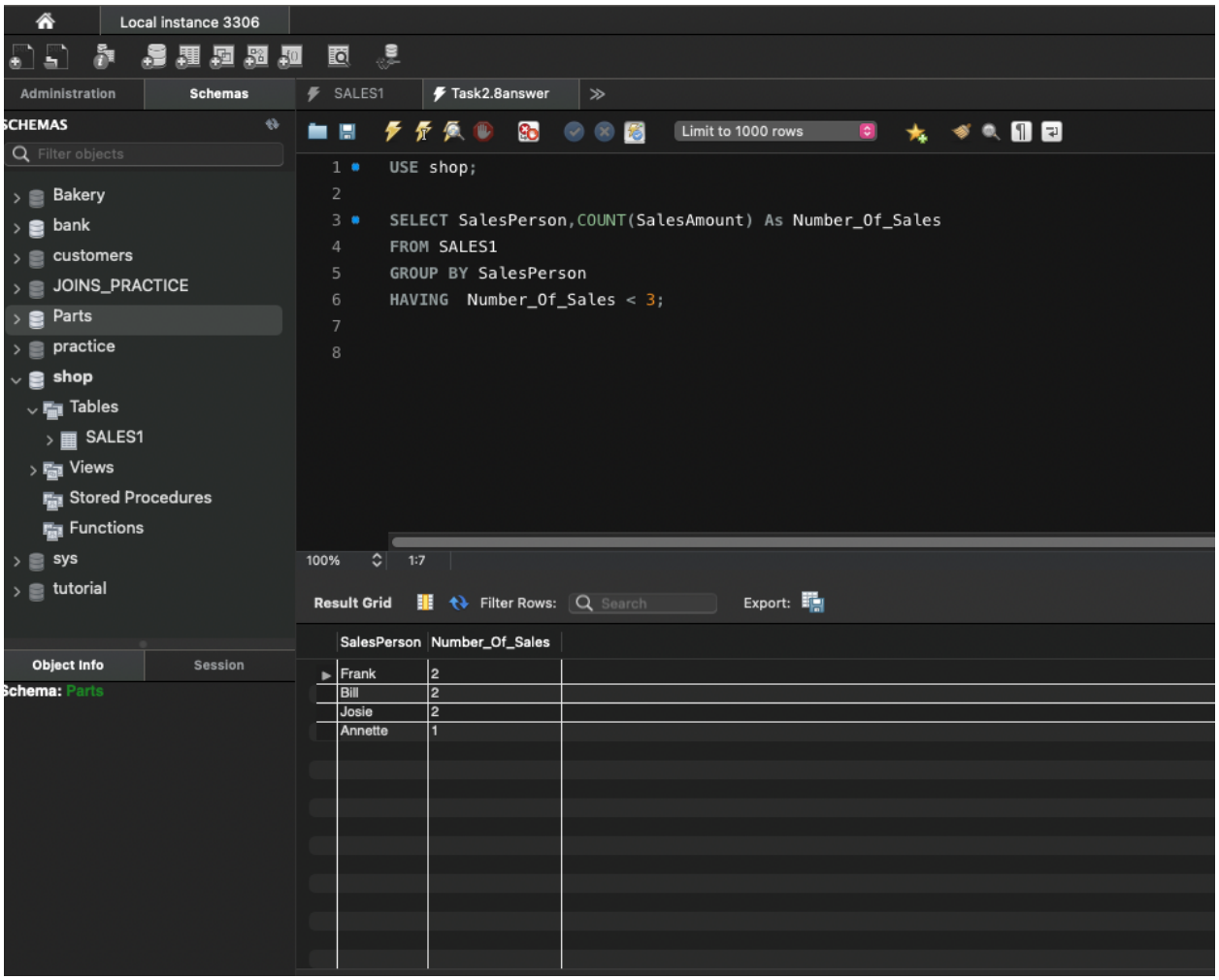


ANSWER:

```
USE shop;

SELECT
   Store,
   CONCAT('£',ROUND((AVG(SalesAmount)),2)) AS Ave_Sales
FROM SALES1
Group by Store;
```

8. Count the number of sales by each person if they had less than 3 sales for the past period.

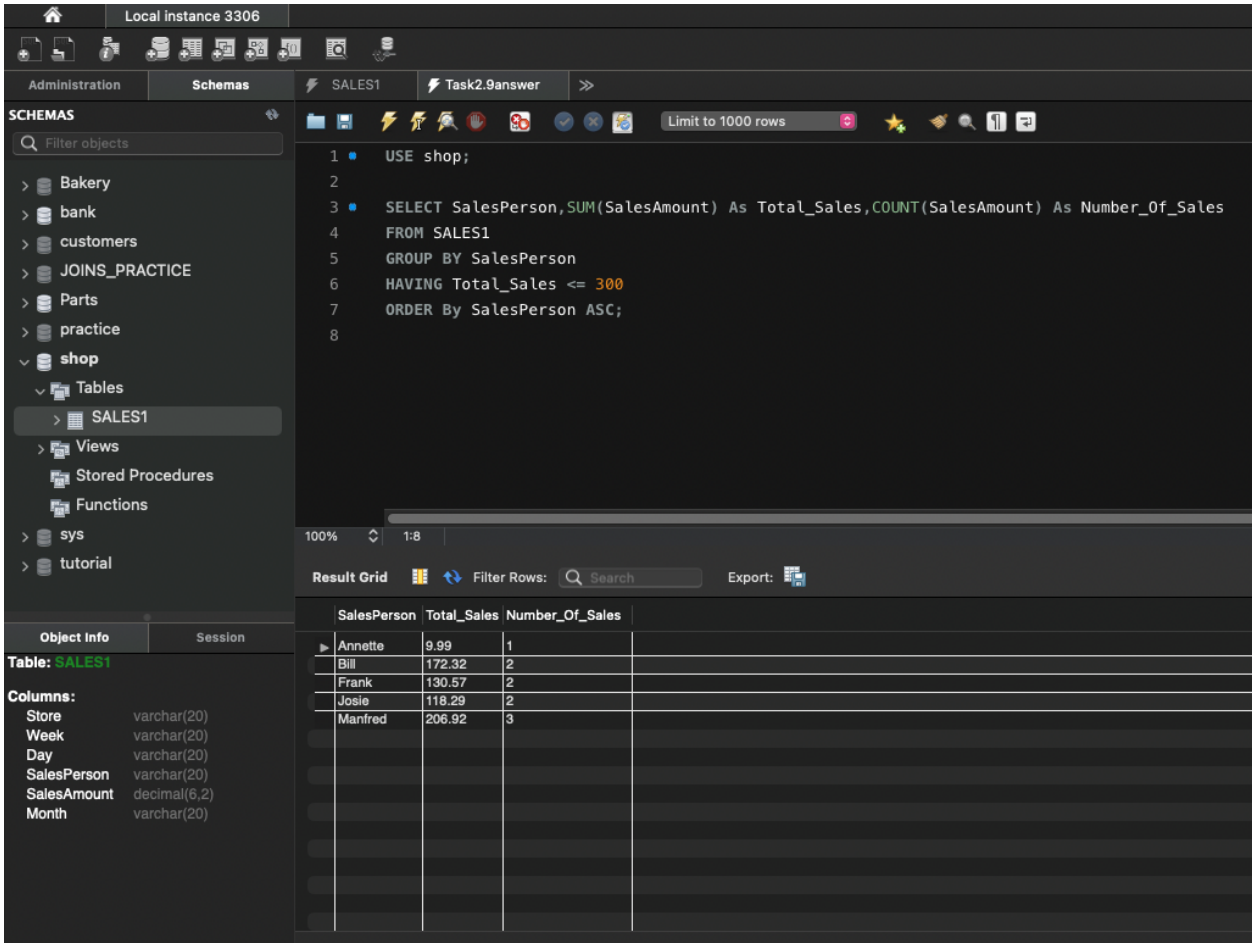


ANSWER:

```
USE shop;

SELECT SalesPerson,COUNT(SalesAmount) As Number_Of_Sales
FROM SALES1
GROUP BY SalesPerson
HAVING  Number_Of_Sales < 3;
```

9. Find the number (count) of sales by each person, but only if they made less than or equal to £300 worth of sales for the past period.



ANSWER:

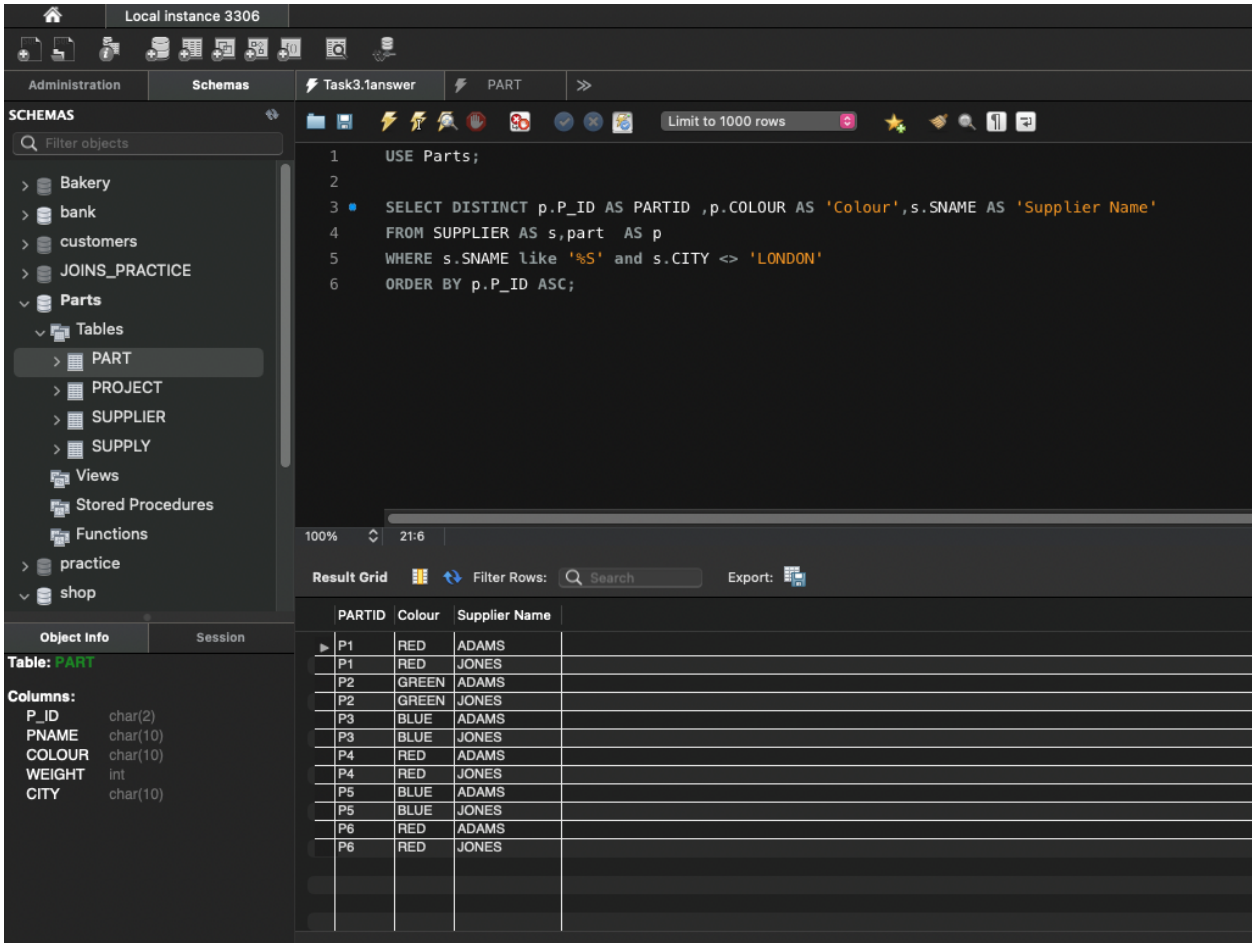
```
USE shop;

SELECT SalesPerson,SUM(SalesAmount) As Total_Sales,COUNT(SalesAmount) As Number_Of_Sales
FROM SALES1
GROUP BY SalesPerson
HAVING Total_Sales <= 300
ORDER By SalesPerson ASC;
```

TASK 3

USE PARTS DB TO WRITE THE FOLLOWING QUERIES

1. Return the PartID, Colour and Supplier name, where the supplier's surname ends in an S, and the Supplier city is not London. Ensure the values are Unique.



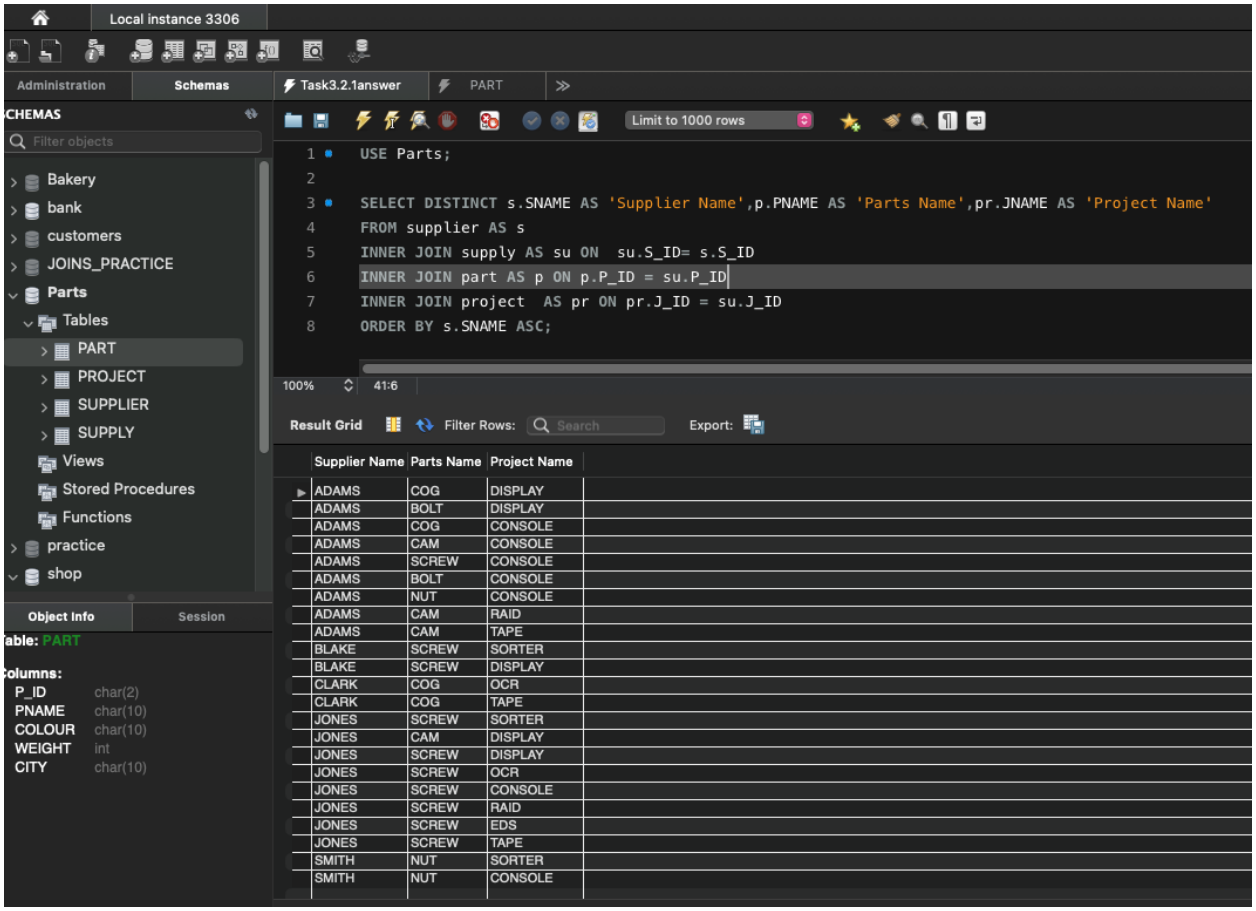
ANSWER:

```
USE Parts;

SELECT DISTINCT p.P_ID AS PARTID ,p.COLOUR AS 'Colour',s.SNAME AS 'Supplier Name'
FROM SUPPLIER AS s,part  AS p
WHERE s.SNAME like '%S' and s.CITY <> 'LONDON'
ORDER BY p.P_ID ASC;
```

2. Return the supplier name, part name and project name for each case where the following conditions are true:

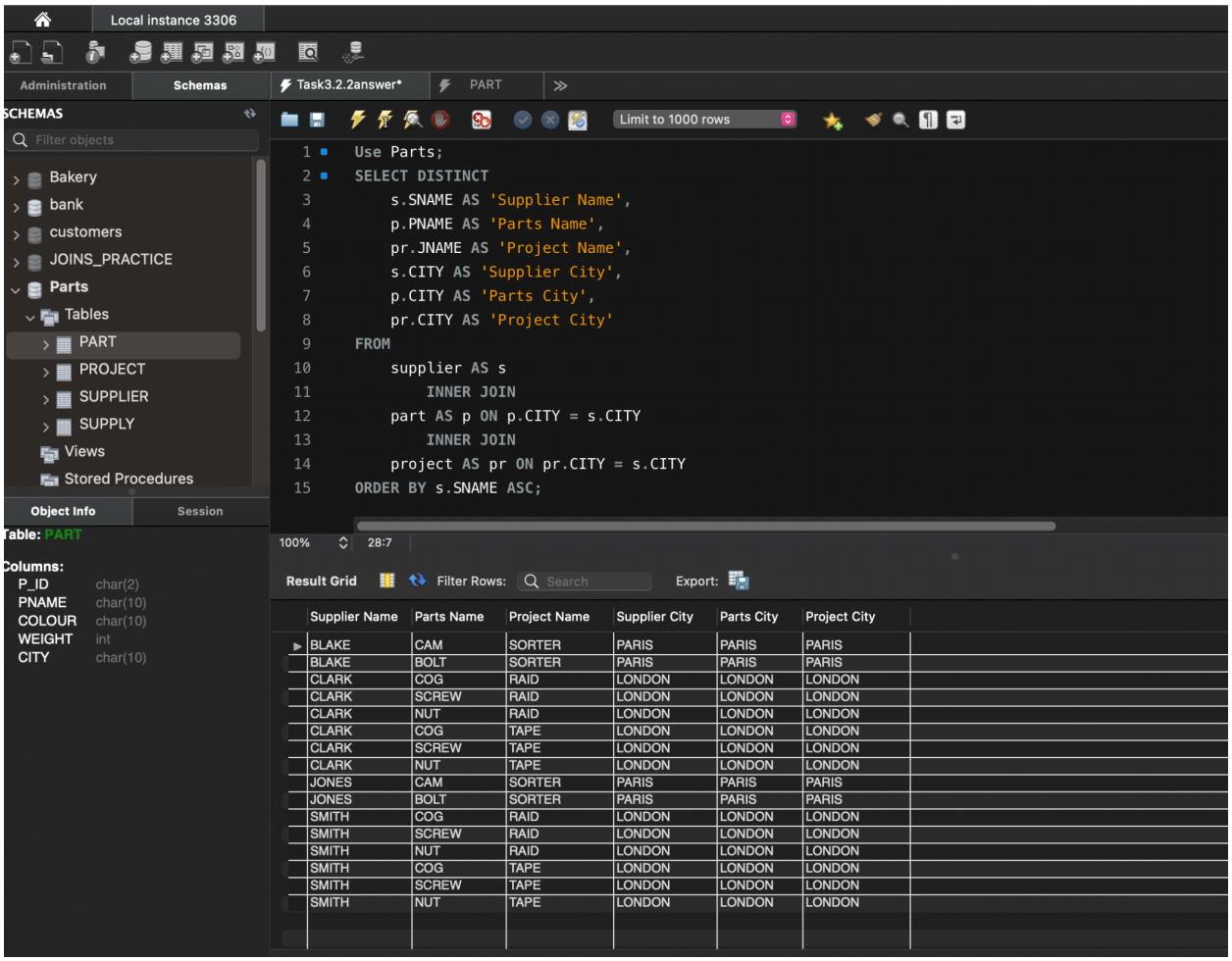
i. The supplier supplies a project with a part;



ANSWER:
USE Parts;

SELECT DISTINCT s.SNAME AS 'Supplier Name',p.PNAME AS 'Parts Name',pr.JNAME AS 'Project Name'
FROM supplier AS s
INNER JOIN supply AS su ON su.S_ID= s.S_ID
INNER JOIN part AS p ON p.P_ID = su.P_ID
INNER JOIN project AS pr ON pr.J_ID = su.J_ID
ORDER BY s.SNAME ASC;

ii. And where the supplier’s city, project city and part city are the same.



ANSWER:

```
Use Parts;

SELECT DISTINCT
    s.SNAME AS 'Supplier Name',
    p.PNAME AS 'Parts Name',
    pr.JNAME AS 'Project Name',
    s.CITY AS 'Supplier City',
    p.CITY AS 'Parts City',
    pr.CITY AS 'Project City'
FROM
    supplier AS s
        INNER JOIN
    part AS p ON p.CITY = s.CITY
        INNER JOIN
    project AS pr ON pr.CITY = s.CITY
ORDER BY s.SNAME ASC;
```