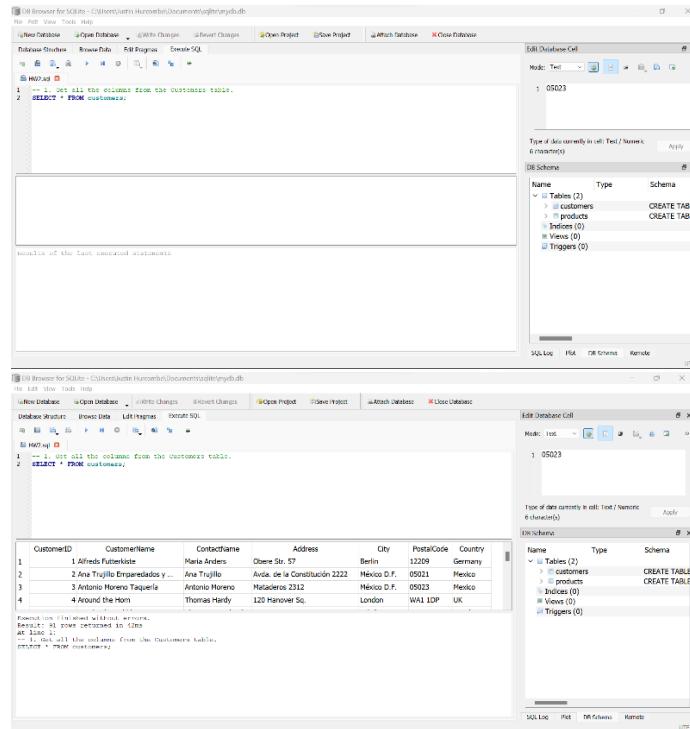


A database is provided. There are two tables in it: customers and products.

Use these two tables to finish the following query requests:

1. Get all the columns from the Customers table.



The screenshot shows the DB Browser for SQLite interface. A query window contains the following SQL code:

```
1 SELECT * FROM customers;
```

The results pane shows the following data from the customers table:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Oberstr. 57	Berlin	12209	Germany
2	2 An Trujillo Emparedados y ...	Ana Trujillo	Ave de la Constitución 2222	México D.F.	05021	Mexico
3	3 Antonio Moreno Taquería	Antonio Moreno	Madero 2312	México D.F.	05023	Mexico
4	4 Around the Horn	Thomas Hardy	123 Hanover Sq.	London	WA1 1OP	UK

## 2. Get all Cities from the Customers table.

The image contains three vertically stacked screenshots of DB Browser for SQLite. Each screenshot shows a database window with a toolbar at the top and a central pane for executing SQL queries. The queries are as follows:

```
1 -- 1. Get all the columns from the Customers table.
2 SELECT * FROM customers;
3 -- 2. Get all Cities from the Customers table.
4 SELECT City FROM customers;
```

The results of the first query show a table with columns: CustomerID, CustomerName, ContactName, Address, City, PostalCode, and Country. The results of the second query show a table with the City column only, listing Berlin, Mexico D.F., Mexico D.F., and London.

The third screenshot shows the results of the second query again, with the City column showing Berlin.

## 3. Select all the different values from the Country column in the Customers table.

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql

```

1 -- 1. Get all the columns from the Customers table.
2 SELECT * FROM customers;
3 -- 2. Get all cities from the Customers table.
4 SELECT city FROM customers;
5 -- 3. Select all the different values from the Country column in the Customers table.
6 SELECT DISTINCT country FROM customers;

```

**City**

88 Resende
89 Seattle
90 Helsinki
91 Walla

Execution finished without errors.  
Result: 91 rows returned in 0ms  
At line 3:  
-- 2. Get all Cities from the Customers table.  
SELECT city FROM customers;

**DB Schema**

Name	Type	Schema
Tables (2)		
> customers	CREATE TABLE	
> products	CREATE TABLE	
Indices (0)		
Views (0)		
Triggers (0)		

SQL Log Plot DB Schema Remote

UTF-8

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql

```

1 1. Get all the columns from the Customers table.
2 SELECT * FROM customers;
3 -- 2. Get all cities from the Customers table.
4 SELECT city FROM customers;
5 -- 3. Select all the different values from the Country column in the Customers table.
6 SELECT DISTINCT country FROM customers;

```

**Country**

1 Germany
2 Mexico
3 Mexico
4 UK
5 Sweden
6 France
7 Spain
8 Canada

Execution finished without errors.  
Result: 22 rows returned in 0ms  
At line 3:  
-- 3. Select all the different values from the Country column in the Customers table.  
SELECT DISTINCT country FROM customers;

**DB Schema**

Name	Type	Schema
Tables (2)		
> customers	CREATE TABLE	
> products	CREATE TABLE	
Indices (0)		
Views (0)		
Triggers (0)		

SQL Log Plot DB Schema Remote

UTF-8

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql

```

1 1. Get all the columns from the Customers table.
2 SELECT * FROM customers;
3 -- 2. Get all cities from the Customers table.
4 SELECT city FROM customers;
5 -- 3. Select all the different values from the Country column in the Customers table.
6 SELECT DISTINCT country FROM customers;

```

**Country**

16 Venezuela
17 Ireland
18 Belgium
19 Norway
20 Denmark
21 Finland
22 Poland

Execution finished without errors.  
Result: 22 rows returned in 0ms  
At line 3:  
-- 3. Select all the different values from the Country column in the Customers table.  
SELECT DISTINCT country FROM customers;

**DB Schema**

Name	Type	Schema
Tables (2)		
> customers	CREATE TABLE	
> products	CREATE TABLE	
Indices (0)		
Views (0)		
Triggers (0)		

SQL Log Plot DB Schema Remote

UTF-8

4. Select all records where the City column has the value "Berlin".

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sql\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Program Execute SQL

Mode: Text

1. Get all the columns from the Customers table.

2. SELECT \* FROM customers;

3. Get all cities from the Customers table.

4. SELECT city FROM customers;

5. Select all the different values from the Country column in the Customers table.

6. SELECT DISTINCT Country FROM customers;

7. Select all records where the City column has the value "Berlin"

8. SELECT \* FROM customers WHERE City = "Berlin";

Execution finished without errors.  
Results: 29 rows returned in 2ms  
At line 1:  
1. Get all the different values from the Country column in the Customers table.  
SELECT DISTINCT Country FROM customers;

DB Schema

Name Type Schema

- Tables (2)
  - customers
  - products
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sql\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Program Execute SQL

Mode: Text

1. Get all the columns from the Customers table.

2. SELECT \* FROM customers;

3. Get all cities from the Customers table.

4. SELECT city FROM customers;

5. Select all the different values from the Country column in the Customers table.

6. SELECT DISTINCT Country FROM customers;

7. Select all records where the City column has the value "Berlin"

8. SELECT \* FROM customers WHERE City = "Berlin";

Execution finished without errors.  
Results: 1 rows returned in 2ms  
At line 1:  
1. Select all records where the City column has the value "Berlin"  
SELECT \* FROM customers WHERE City = "Berlin";

DB Schema

Name Type Schema

- Tables (2)
  - customers
  - products
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

## 5. Select all records where the CustomerID column has the value 32.

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sql\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Program Execute SQL

Mode: Text

1. Get all the columns from the Customers table.

2. SELECT \* FROM customers;

3. Get all cities from the Customers table.

4. SELECT city FROM customers;

5. Select all the different values from the Country column in the Customers table.

6. SELECT DISTINCT Country FROM customers;

7. Select all records where the City column has the value "Berlin"

8. SELECT \* FROM customers WHERE City = "Berlin";

9. Select all records where the CustomerID column has the value 32.

10. SELECT \* FROM customers WHERE CustomerID = 32;

Execution finished without errors.  
Results: 1 rows returned in 2ms  
At line 8:  
1. Select all records where the CustomerID column has the value 32.  
SELECT \* FROM customers WHERE CustomerID like "%32%";

DB Schema

Name Type Schema

- Tables (2)
  - customers
  - products
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sql\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Program Execute SQL

Mode: Text

1. Get all the columns from the Customers table.

2. SELECT \* FROM customers;

3. Get all cities from the Customers table.

4. SELECT city FROM customers;

5. Select all the different values from the Country column in the Customers table.

6. SELECT DISTINCT Country FROM customers;

7. Select all records where the City column has the value "Berlin"

8. SELECT \* FROM customers WHERE City = "Berlin";

9. Select all records where the CustomerID column has the value 32.

10. SELECT \* FROM customers WHERE CustomerID = 32;

Execution finished without errors.  
Results: 1 rows returned in 2ms  
At line 8:  
1. Select all records where the CustomerID column has the value 32.  
SELECT \* FROM customers WHERE CustomerID = 32;

DB Schema

Name Type Schema

- Tables (2)
  - customers
  - products
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

## 6. Select all records from the Customers table, sort the result alphabetically by the column City.

The screenshots show the DB Browser for SQLite interface with three separate database windows. Each window displays the following SQL query and its results:

```

2 SELECT * FROM customers;
3   1. Select all entries from the Customers table.
4   2. Select all the different values from the Country column in the Customers table.
5   3. Select all records where the City column has the value "Berlin".
6   4. Select all records where the City column has the value "Köln".
7   5. Select all records where the CustomerID column has the value 32.
8   6. Select all records from the Customers table, sort the result alphabetically by the column City.
9   7. Select all records from the Customers table, sort the result alphabetically by the column City.
10  8. Select all records from the Customers table, sort the result alphabetically by the column City.
11  9. Select all records from the Customers table, sort the result alphabetically by the column City.
12  10. Select all records from the Customers table, sort the result alphabetically by the column City.
13  11. Select all records from the Customers table, sort the result alphabetically by the column City.
14  12. Select all records from the Customers table, sort the result alphabetically by the column City.
15  13. Select all records from the Customers table, sort the result alphabetically by the column City.
16  14. Select all records from the Customers table, sort the result alphabetically by the column City.
17  15. Select all records from the Customers table, sort the result alphabetically by the column City.
18  16. Select all records from the Customers table, sort the result alphabetically by the column City.
19  17. Select all records from the Customers table, sort the result alphabetically by the column City.
20  18. Select all records from the Customers table, sort the result alphabetically by the column City.
21  19. Select all records from the Customers table, sort the result alphabetically by the column City.
22  20. Select all records from the Customers table, sort the result alphabetically by the column City.
23  21. Select all records from the Customers table, sort the result alphabetically by the column City.
24  22. Select all records from the Customers table, sort the result alphabetically by the column City.
25  23. Select all records from the Customers table, sort the result alphabetically by the column City.
26  24. Select all records from the Customers table, sort the result alphabetically by the column City.
27  25. Select all records from the Customers table, sort the result alphabetically by the column City.
28  26. Select all records from the Customers table, sort the result alphabetically by the column City.
29  27. Select all records from the Customers table, sort the result alphabetically by the column City.
30  28. Select all records from the Customers table, sort the result alphabetically by the column City.
31  29. Select all records from the Customers table, sort the result alphabetically by the column City.
32  30. Select all records from the Customers table, sort the result alphabetically by the column City.
33  31. Select all records from the Customers table, sort the result alphabetically by the column City.
34  32. Select all records from the Customers table, sort the result alphabetically by the column City.
35  33. Select all records from the Customers table, sort the result alphabetically by the column City.
36  34. Select all records from the Customers table, sort the result alphabetically by the column City.
37  35. Select all records from the Customers table, sort the result alphabetically by the column City.
38  36. Select all records from the Customers table, sort the result alphabetically by the column City.
39  37. Select all records from the Customers table, sort the result alphabetically by the column City.
40  38. Select all records from the Customers table, sort the result alphabetically by the column City.
41  39. Select all records from the Customers table, sort the result alphabetically by the column City.
42  40. Select all records from the Customers table, sort the result alphabetically by the column City.
43  41. Select all records from the Customers table, sort the result alphabetically by the column City.
44  42. Select all records from the Customers table, sort the result alphabetically by the column City.
45  43. Select all records from the Customers table, sort the result alphabetically by the column City.
46  44. Select all records from the Customers table, sort the result alphabetically by the column City.
47  45. Select all records from the Customers table, sort the result alphabetically by the column City.
48  46. Select all records from the Customers table, sort the result alphabetically by the column City.
49  47. Select all records from the Customers table, sort the result alphabetically by the column City.
50  48. Select all records from the Customers table, sort the result alphabetically by the column City.
51  49. Select all records from the Customers table, sort the result alphabetically by the column City.
52  50. Select all records from the Customers table, sort the result alphabetically by the column City.
53  51. Select all records from the Customers table, sort the result alphabetically by the column City.
54  52. Select all records from the Customers table, sort the result alphabetically by the column City.
55  53. Select all records from the Customers table, sort the result alphabetically by the column City.
56  54. Select all records from the Customers table, sort the result alphabetically by the column City.
57  55. Select all records from the Customers table, sort the result alphabetically by the column City.
58  56. Select all records from the Customers table, sort the result alphabetically by the column City.
59  57. Select all records from the Customers table, sort the result alphabetically by the column City.
60  58. Select all records from the Customers table, sort the result alphabetically by the column City.
61  59. Select all records from the Customers table, sort the result alphabetically by the column City.
62  60. Select all records from the Customers table, sort the result alphabetically by the column City.
63  61. Select all records from the Customers table, sort the result alphabetically by the column City.
64  62. Select all records from the Customers table, sort the result alphabetically by the column City.
65  63. Select all records from the Customers table, sort the result alphabetically by the column City.
66  64. Select all records from the Customers table, sort the result alphabetically by the column City.
67  65. Select all records from the Customers table, sort the result alphabetically by the column City.
68  66. Select all records from the Customers table, sort the result alphabetically by the column City.
69  67. Select all records from the Customers table, sort the result alphabetically by the column City.
70  68. Select all records from the Customers table, sort the result alphabetically by the column City.
71  69. Select all records from the Customers table, sort the result alphabetically by the column City.
72  70. Select all records from the Customers table, sort the result alphabetically by the column City.
73  71. Select all records from the Customers table, sort the result alphabetically by the column City.
74  72. Select all records from the Customers table, sort the result alphabetically by the column City.
75  73. Select all records from the Customers table, sort the result alphabetically by the column City.
76  74. Select all records from the Customers table, sort the result alphabetically by the column City.
77  75. Select all records from the Customers table, sort the result alphabetically by the column City.
78  76. Select all records from the Customers table, sort the result alphabetically by the column City.
79  77. Select all records from the Customers table, sort the result alphabetically by the column City.
80  78. Select all records from the Customers table, sort the result alphabetically by the column City.
81  79. Select all records from the Customers table, sort the result alphabetically by the column City.
82  80. Select all records from the Customers table, sort the result alphabetically by the column City.
83  81. Select all records from the Customers table, sort the result alphabetically by the column City.
84  82. Select all records from the Customers table, sort the result alphabetically by the column City.
85  83. Select all records from the Customers table, sort the result alphabetically by the column City.
86  84. Select all records from the Customers table, sort the result alphabetically by the column City.
87  85. Select all records from the Customers table, sort the result alphabetically by the column City.
88  86. Select all records from the Customers table, sort the result alphabetically by the column City.
89  87. Select all records from the Customers table, sort the result alphabetically by the column City.
90  88. Select all records from the Customers table, sort the result alphabetically by the column City.
91  89. Select all records from the Customers table, sort the result alphabetically by the column City.

```

The results of the query are displayed in a table format, showing columns: CustomerID, CustomerName, ContactName, Address, City, PostalCode, and Country. The data is sorted alphabetically by City. The first few rows of the results are as follows:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Great Lakes Food Market	Howard Snyder	2732 Baker Blvd.	Eugene	97403	USA
2	Bólido Comidas preparadas	Martin Sommer	C Araquil 67	Madrid	28023	Spain
3	17 Drachenblut Delikatessen	Sven Ottlieb	Walserweg 21	Aachen	52066	Germany
4	65 Rattlersnake Canyon Grocery	Paula Wilson	2817 Milton Dr.	Albuquerque	87110	USA
5	55 Old World Delicatessen	Rene Phillips	2743 Bering St.	Anchorage	99508	USA
6	29 Galería del gastrónomo	Eduardo SaaVEDRA	Rambla de Cataluña 23	Barcelona	08022	Spain
7	46 LILA-Supermercado	Carlos González	Carrera 52 con Ave. Bolívar #22-91	Barquisimeto	3508	Venezuela
8	49 Magazzini Alimentari Riuniti	Giovanni Rovelli	Via Ludovico il Moro 22	Bergamo	24100	Italy
9	A-Amadeus Compañía	Media Andrade	Obra Social 32	Madrid	28000	Spain

## 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Mode: Text

NULL

Type of data currently in cell: NULL Apply

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

**HWSql**

```

4 SELECT City FROM customers;
5   -- 3. Select all the different values from the Country column in the Customers table.
6 SELECT DISTINCT Country FROM customers;
7   -- 4. Select all records where the City column has the value "Berlin"
8 SELECT * FROM customers WHERE city = "Berlin";
9   -- 5. Select all records where the CustomerID column has the value 32.
10 SELECT * FROM customers WHERE customerid = 32;
11   -- 6. Select all records from the Customers table, sort the result alphabetically by the column City.
12 SELECT * FROM customers ORDER BY City;
13   -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.
14 SELECT * FROM customers ORDER BY City DESC;
```

Execution finished without errors.  
Results: 10 rows returned in 0ms  
At line 13:  
-- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
SELECT \* FROM customers ORDER BY City DESC;

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Mode: Text

NULL

Type of data currently in cell: NULL Apply

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

**HWSql**

```

3 -- 1. Get all cities from the Customers table.
4 SELECT City FROM customers;
5   -- 2. Get all the different values from the Country column in the Customers table.
6 SELECT DISTINCT Country FROM customers;
7   -- 3. Select all records where the City column has the value "Berlin"
8 SELECT * FROM customers WHERE city = "Berlin";
9   -- 4. Select all records where the CustomerID column has the value 32.
10 SELECT * FROM customers WHERE customerid = 32;
11   -- 5. Select all records from the Customers table, sort the result alphabetically by the column City.
12 SELECT * FROM customers ORDER BY City;
13   -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.
```

Execution finished without errors.  
Results: 10 rows returned in 0ms  
At line 13:  
-- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
SELECT \* FROM customers ORDER BY City DESC;

DB Browser for SQLite - C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Mode: Text

NULL

Type of data currently in cell: NULL Apply

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log Plot DB Schema Remote

UTF-8

**HWSql**

```

2 Get all Cities from the Customers table.
3 SELECT City FROM customers;
4   -- 1. Get all the different values from the Country column in the Customers table.
5 SELECT DISTINCT Country FROM customers;
6   -- 4. Select all records where the City column has the value "Berlin"
7 SELECT * FROM customers WHERE city = "Berlin";
8   -- 5. Select all records where the CustomerID column has the value 32.
9 SELECT * FROM customers WHERE customerid = 32;
10   -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.
11 SELECT * FROM customers ORDER BY City DESC;
```

Execution finished without errors.  
Results: 10 rows returned in 0ms  
At line 13:  
-- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
SELECT \* FROM customers ORDER BY City DESC;

8. Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.

DB Browser for SQLite C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql Lecture 3 Practice.sql

```

9 -- 5. Select all records where the CustomerID column has the value 32.
10 SELECT * FROM customers WHERE CustomerID = 32;
11 -- 6. Select all records from the Customers table, sort the result alphabetically by the column City.
12 SELECT * FROM customers ORDER BY City;
13 -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.
14 SELECT * FROM customers ORDER BY City DESC;
15 /* 8. Select all records from the Customers table, sort the result alphabetically,
16 first by the column Country, then by the column city. */
17 SELECT * FROM customers ORDER BY Country, City;
18

```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
85	49 Magazzini Alimentari Riuniti	Giovanni Rovelli	Via Ludovico il Moro 22	Bergamo	24100	Italy
86	46 LILA-Supermercado	Carlos González	Carrera 52 con Ave. Bolívar ...	Bogotá	3508	Venezuela
87	29 Galería del gastronomo	Eduardo Saavedra	Rambla de Cataluña 23	Barcelona	08022	Spain
88	55 Old World Delicatessen	Rene Phillips	2743 Bering St.	Anchorage	99508	USA
89	65 Rattlesnake Canyon Grocery	Paula Wilson	2817 Milton Dr.	Albuquerque	87110	USA
90	17 Drachenblut Delikatessend	Sven Ottlieb	Walserweg 21	Aachen	52066	Germany
91	8 Bólido Comidas preparadas	Martin Sommer	C Araquil 67	Madrid	28023	Spain

Execution finished without errors.  
Result: 91 rows returned in 3ms  
At line 13:  
-- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
SELECT \* FROM customers ORDER BY Country, City;

DB Browser for SQLite C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql Lecture 3 Practice.sql

```

9 -- 5. Select all records where the CustomerID column has the value 32.
10 SELECT * FROM customers WHERE CustomerID = 32;
11 -- 6. Select all records from the Customers table, sort the result alphabetically by the column City.
12 SELECT * FROM customers ORDER BY City;
13 -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column City.
14 SELECT * FROM customers ORDER BY City DESC;
15 /* 8. Select all records from the Customers table, sort the result alphabetically,
16 first by the column Country, then by the column city. */
17 SELECT * FROM customers ORDER BY Country, City;
18

```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	12 Cactus Comidas para llevar	Patricia Simpson	Cerroto 333	Buenos Aires	1010	Argentina
2	54 Océano Atlántico Ltda.	Yvonne Moncada	Ing. Gustavo Moncada 8585 Pt...	Buenos Aires	1010	Argentina
3	64 Rancho grande	Sergio Gutiérrez	Av. del Libertador 900	Buenos Aires	1010	Argentina
4	20 Ernst Handel	Roland Mendel	Kirchgasse 6	Graz	8010	Austria
5	59 Piccolo und mehr	Georg Pipp	Gesitweg 14	Salzburg	5020	Austria
6	50 Maison Dewey	Catherine Dewey	Rue Joseph-Bens 532	Bruxelles	B-1180	Belgium
7	76 Suprèmes délices	Pascale Cartain	Boulevard Tirou 255	Charleroi	B6000	Belgium

Execution finished without errors.  
Result: 91 rows returned in 3ms  
At line 13:  
-- 8. Select all records from the Customers table, sort the result alphabetically,
first by the column Country, then by the column City.  
SELECT \* FROM customers ORDER BY Country, City;

DB Browser for SQLite C:\Users\Justin Hurcombe\Documents\sqlite\mydb.db

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

HW2.sql Lecture 3 Practice.sql

```

9 -- 5. Select all records where the CustomerID column has the value 32.
10 SELECT * FROM customers WHERE CustomerID = 32;
11 -- 6. Select all records from the Customers table, sort the result alphabetically by the column City.
12 SELECT * FROM customers ORDER BY City;
13 -- 7. Select all records from the Customers table, sort the result reversed alphabetically by the column city.
14 SELECT * FROM customers ORDER BY City DESC;
15 /* 8. Select all records from the Customers table, sort the result alphabetically,
16 first by the column Country, then by the column City. */
17 SELECT * FROM customers ORDER BY Country, City;
18

```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
85	45 Let's Stop N Shop	Jaime Yorres	87 Polk St. Suite 5	San Francisco	94117	USA
86	89 White Clover Markets	Karl Jablonski	305 14th Ave. S. Suite 3B	Seattle	98128	USA
87	43 Lazy K Kountry Store	John Steel	12 Orchestra Terrace	Walla Walla	99362	USA
88	46 LILA-Supermercado	Carlos González	Carrera 52 con Ave. Bolívar ...	Bogotá	3508	Venezuela
89	33 GROSSELLA-Restaurante	Manuel Pereira	51 Ave. Los Palos Grandes	Caracas	1081	Venezuela
90	47 LINO-Delicatessen	Felipe Izquierdo	Ave. 5 de Mayo 100	I. de Margarita	4980	Venezuela
91	35 HILARIÓN-Abastos	Carlos Hernández	Carrera 22 con Ave. Carlos ...	San Cristóbal	5022	Venezuela

Execution finished without errors.  
Result: 91 rows returned in 3ms  
At line 13:  
-- 8. Select all records from the Customers table, sort the result alphabetically,
first by the column Country, then by the column City.  
SELECT \* FROM customers ORDER BY Country, City;

9. Select all records where the City column has the value 'Berlin' and the PostalCode column has the value '12209'.

File Edit View Tools Help

New Database

Database Structure

File Edit View Tools Help

New Database

Database Structure

File Edit View Tools Help

New Database

Database Structure

File Edit View Tools Help

New Database

Database Structure

10. Select all records where the City column has the value 'Berlin' OR 'London'.

DB Browser for SQL Server - C:\Users\Justin\Downloads\MyFirstDatabase.mdf

File Edit View Tools Help

Open Database Open Database Open Changes Save Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Programs Execute SQL

HM2010 [ (return) ] [ (return) ]

17. T. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
18. SELECT \* FROM customers ORDER BY City DESC;  
19. T. To select all records from the Customers table, sort the result alphabetically.  
20. SELECT \* FROM customers ORDER BY Country, City;  
21. T. Select all records from the Customers table where the value 'Berlin' and the PostalCode column has the value '12209'.  
22. SELECT \* FROM customers WHERE City = 'Berlin' AND PostalCode = '12209';  
23. T. Select all records where the city column has the value 'Berlin' OR 'London'.  
24. SELECT \* FROM customers WHERE City = 'Berlin' OR City = 'London';

CustomerID CustomerName ContactName Address City PostalCode Country  
1 Alfreds Futterkiste Maria Anders Obere Str. 57 Berlin 12209 Germany

Execution finished without errors.  
Resultset: 1 rows selected in 0ms  
At line 20  
-- T. Select all records where the City column has the value 'Berlin' OR 'London'.  
SELECT \* FROM customers WHERE City = 'Berlin' OR City = 'London';

DB Browser for SQL Server - C:\Users\Justin\Downloads\MyFirstDatabase.mdf

File Edit View Tools Help

Open Database Open Database Open Changes Save Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Programs Execute SQL

HM2010 [ (return) ] [ (return) ]

17. SELECT \* FROM customers ORDER BY City;  
18. T. Select all records from the Customers table, sort the result reversed alphabetically by the column City.  
19. SELECT \* FROM customers ORDER BY City DESC;  
20. T. Select all records from the Customers table, sort the result alphabetically.  
21. T. First by the column Country, then by the column City. /  
22. SELECT \* FROM customers ORDER BY Country, City;  
23. T. Select all records where the city column has the value 'Berlin' and the PostalCode column has the value '12209'.  
24. SELECT \* FROM customers WHERE City = 'Berlin' AND PostalCode = '12209';  
25. T. Select all records where the city column has the value 'Berlin' OR 'London'.  
26. SELECT \* FROM customers WHERE City = 'Berlin' OR City = 'London';

CustomerID CustomerName ContactName Address City PostalCode Country  
1 Alfreds Futterkiste Maria Anders Obere Str. 57 Berlin 12209 Germany  
2 A Round the Horn Thomas Hardy 120 Hanover Sq. London W1A 1DP UK  
3 B's Beverages Victoria Ashworth Fair味bury Circus London EC2 5NT UK  
4 Consolidated Holdings Elizabeth Brown Berkeley Gardens 12 Brewster London WXL 6LT UK  
5 Eastern Connection Ann Devon 35 King George London WX3 6PW UK  
6 North South Simon Crowther South House 300 Queensbridge London SW1 1RZ UK  
7 Seven Seas Imports Han Kumar 90 Wadhurst Rd. London OX15 4NB UK

Execution finished without errors.  
Resultset: 2 rows selected in 0ms  
At line 25  
-- T. Select all records where the City column has the value 'Berlin' OR 'London'.  
SELECT \* FROM customers WHERE City = 'Berlin' OR City = 'London';

11. Select all records where City is NOT "Berlin".

12. Select all records from the Customers where the PostalCode column is NOT empty.

```
US Browser for SQLServer | C:\Users\yatin\OneDrive\Documents\sql\onebyone\lab1
File Edit View Tools Help
New Database Open Database Write Changes Refresh Changes Open Project Save Project All Databases Close Database
Database Structure Home Data Edit Pages Execute SQL
File Help Database Tools Mode: Text
NIFI
Type of data currently in cell: NULL Apply
DB Schema
Name Type Sch
Tables (2)
> customers
products
Indices (0)
Views (0)
Triggers (0)
1. SELECT * FROM customers ORDER BY Country, City
2. SELECT * FROM customers WHERE city = 'Berlin' AND postalcode = '12209'.
-- 15. Select all 15 records where the city column has the value 'Berlin' or 'London'.
-- 16. Select all records where the city column has the value 'London'.
-- 17. Select all records whose city is NOT 'Berlin'.
23. SELECT * FROM customers WHERE NOT city = 'Berlin'.
24. SELECT * FROM customers WHERE postalcode IS NOT NULL.
25. SELECT * FROM customers WHERE postalcode IS NOT NULL;
Execution finished without errors.
Result: 1 row(s) returned in 0ms
12. Select all records from the Customers where the PostalCode column is NOT empty.
SELECT * FROM customers WHERE PostalCode IS NOT NULL;
```

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure with 'Tables (2)' expanded, showing 'customers' and 'products'.
- Table View:** The 'customers' table is selected, displaying columns: CustomerID, CustomerName, ContactName, Address, City, PostalCode, and Country.
- Script Selection Context Menu:** A context menu is open over the table rows, with the 'Script Selection' option highlighted. Other visible options include 'Script Selection as' (with 'CREATE TABLE' and 'CREATE TABLE and Data' choices), 'Script Selection into', 'Script Selection as (New Query)', and 'Script Selection into (New Query)'.
- SQL Editor:** The main pane shows a query window with the following T-SQL code:

```
17 -- SELECT * FROM customers ORDER BY Country, City;
18 -- 1. Select all records where the City column has the value 'Berlin' and the PostalCode column has the value '12209'.
19 -- 2. Select all records where the City column has the value 'Berlin' OR City = 'London'.
20 -- 3. select all records where the City column has the value 'Berlin' OR City = 'London';
21 -- 4. SELECT * FROM customers WHERE City = 'Berlin' OR City = 'London';
22 -- 5. SELECT * FROM customers WHERE NOT City = 'Berlin';
23 -- 6. SELECT * FROM customers WHERE NOT City = 'Berlin' AND NOT City = 'London';
24 -- 7. SELECT * FROM customers WHERE PostalCode IS NOT NULL;
```
- Status Bar:** Shows 'SQL Log' and 'Plan'.

### 13. Select all records where the value of the City column starts with the letter "a".

```

13. Select all records where the value of the City column starts with the letter "a".
-- 1. Select all records where the value of the City column starts with the letter "a".
SELECT * FROM customers WHERE City LIKE 'A%'

CustomerID CustomerName ContactName Address City PostalCode Country
1 17 Drachenblut Delikatessen Sven Ottieb Wallenweg 21 Aachen 52066 Germany
2 55 Old World Delicatessen Rene Phillips 2743 Bering St. Anchorage 99508 USA
3 65 Rattlesnake Canyon Grocery Paula Wilson 2817 Milton Dr. Albuquerque 87110 USA

```

### 14. Select all records where the value of the City column ends with the letter "a".

```

14. Select all records where the value of the City column ends with the letter "a".
-- 1. Select all records where the value of the City column ends with the letter "a".
SELECT * FROM customers WHERE City LIKE '%a'

CustomerID CustomerName ContactName Address City PostalCode Country
1 17 Drachenblut Delikatessen Sven Ottieb Wallenweg 21 Aachen 52066 Germany
2 55 Old World Delicatessen Rene Phillips 2743 Bering St. Anchorage 99508 USA
3 65 Rattlesnake Canyon Grocery Paula Wilson 2817 Milton Dr. Albuquerque 87110 USA

14. Select all records where the value of the City column ends with the letter "a".
-- 2. Select all records where the value of the City column ends with the letter "a".
SELECT * FROM customers WHERE City LIKE '%a'

CustomerID CustomerName ContactName Address City PostalCode Country
1 28 Furia Bacalhau e Frutos do Mar Lino Rodriguez Jardim das rosas n. 32 Lisboa 1675 Portugal
2 29 Galeria do gastronomo Eduardo Saavedra Ramble de Cataluña 23 Barcelona 08022 Spain
3 30 Godos Cocina Típica José Pedro Freyre C Romero 33 Sevilla 41001 Spain
4 43 Lary K Kountry Store John Stark 12 Orchestra Terrace Wala Wala 99362 USA
5 47 LINO-Delicatessen Felipe Izquierdo Ave. 5 de Mayo Pofenmar/ 1 de Margaria 9800 Venezuela
6 60 Princess Isabel Vinhos Isabel de Castro Estrada da saude n. 58 Lisboa 1750 Portugal
7 66 Reggiani Casafici Maurizio Moroni Strada Provinciale 124 Reggio Emilia 41000 Italy
8 91 Wolski Zbyszek ul. Filrowa 68 Wala 01012 Poland

```

## 15. Select all records where the value of the City column contains the letter "a".

The screenshots show the execution of the following T-SQL query in three separate database contexts:

```

21 -- 14. Select all records where the value of the City column contains the letter "a".
22 SELECT * FROM customers WHERE City LIKE '%a%'
23 -- 15. Select all records where the value of the City column starts with the letter "a".
24 SELECT * FROM customers WHERE City LIKE 'a%'
25 -- 16. Select all records where the value of the City column ends with the letter "a".
26 SELECT * FROM customers WHERE City LIKE '%a'
27 -- 17. Select all records where the value of the City column contains the letter "a".
28 SELECT * FROM customers WHERE City LIKE '*a*'
29 -- 18. Select all records where the value of the City column starts with letter "a" and
30 -- ends with the letter "b".
31 SELECT * FROM customers WHERE City LIKE 'ab%'
32 -- 19. Select all records where the value of the City column ends with "ab".
33 SELECT * FROM customers WHERE City LIKE '%ab'

```

Execution results are shown for each query, indicating successful completion with 25 rows returned.

## 16. Select all records where the value of the City column starts with letter "a" and ends with the letter "b".

The screenshot shows the execution of the following T-SQL query in one database context:

```

28 -- 14. Select all records where the value of the City column ends with the letter "a".
29 SELECT * FROM customers WHERE City LIKE '%a%'
30 -- 15. Select all records where the value of the City column contains the letter "a".
31 SELECT * FROM customers WHERE City LIKE 'a%'
32 -- 16. Select all records where the value of the City column starts with letter "a" and
33 -- ends with the letter "b".
34 SELECT * FROM customers WHERE City LIKE 'ab%'

```

Execution finished without errors. Result: query executed successfully. Took 0ms.

There were no records where city started with the letter a and ended with the letter b.

**17. Select all records where the value of the City column does NOT start with the letter "a".**

```

18. Select all records where the value of the City column contains the letter "a".
19. SELECT * FROM customers WHERE city LIKE "%a%";
20. -- 16. Select all records where the value of the City column starts with letter "a" and
21. -- ends with letter "a"/
22. SELECT * FROM customers WHERE city LIKE "a%";
23. -- 17. Select all records where the value of the City column does NOT start with the letter "a";
24. SELECT * FROM customers WHERE city NOT LIKE "a%";
25. -- 18. Select all records where the value of the City column does NOT start with the letter "a";
26. SELECT * FROM customers WHERE city NOT LIKE "%a%";

CustomerID CustomerName ContactName Address City PostalCode Country
1 17 Drachenhaus Sven Ottieb Waisenweg 21 Aachen 52066 Germany
2 55 Old World Delicatessen Rene Phillips 2743 Spring St Anchorage 99508 USA
3 65 Rattanakosin Candy Company Paula Wilson 2817 Milton Dr Albuquerque 87110 USA

Execution finished without errors.
Results 3 rows returned in 2ms.
At line 26
18. Select all records where the value of the City column starts with the letter "a".
source * from customers where city like "a%";

CustomerID CustomerName ContactName Address City PostalCode Country
1 1 Alfredo Futterkiste Maria Anders Obere Str. 57 Berlin 12209 Germany
2 2 Ana Trujillo Emparedados y ... Ana Trujillo Avda de la Constitución 2222 México D.F. 05021 Mexico
3 3 Antonio Moreno Taqueria Antonio Moreno Matadero 2312 México D.F. 05023 Mexico
4 4 Around the Horn Thomas Hardy 120 Hanover Sq. London WA1 1DP UK
5 5 Berglunds snabbköp Christina Berglund Berguvsvägen 8 Luleå 955 22 Sweden
6 6 Blauer See Delikatessen Hanna Moos Forststr. 57 Mannheim 68300 Germany
7 7 Blondel père et fils Frédérique Cœure 24 place Kléber Strasbourg 67000 France

Execution finished without errors.
Results 7 rows returned in 2ms.
At line 26
-- 17. Select all records where the value of the city column does NOT start with the letter "a";
SELECT * FROM customers WHERE city NOT LIKE "%a%";

CustomerID CustomerName ContactName Address City PostalCode Country
82 85 Vins et alcools Chevalier Paul Henrot 59 rue de l'Abbaye Reims 51100 France
83 86 Die Wandmühle Kuh Rita Müller Adenauerallee 900 Stuttgart 70563 Germany
84 87 Wartian Herku Pirkko Koskitalo Torikatu 38 Oulu 90100 Finland
85 88 Wellington Importadora Paula Parente Rua do Mercado 12 Resende 08737 363 Brazil
86 89 White Clover Markets Kari Jämsenki 305 14th Ave. S. Suite 3B Seattle 98128 USA
87 90 Wilman Käla Matti Karttunen Kesäkatu 45 Helsinki 21240 Finland
88 91 WorldCafe Zbyszek ul. Fibrova 68 Wiala 01012 Poland

Execution finished without errors.
Results 8 rows returned in 2ms.
At line 26
-- 18. Select all records where the value of the City column does NOT start with the letter "a";
SELECT * FROM customers WHERE city NOT LIKE "%a%";
```

**18. Select all records where the second letter of the City is an "a".**

```

DB Browser for SQLite - C:\Users\Martin\Hausarbeiten\Documents\sqlitetutorial.db
File Edit View Tools Help
New Database Open Database Write Changes Read Changes Open Project Save Project Attach Database Close Database
Database Structure Recent Dbs Help Pages Execute SQL
File Edit View Tools Help
New Database Open Database Write Changes Read Changes Open Project Save Project Attach Database Close Database
Database Structure Recent Dbs Help Pages Execute SQL

```

Execution finished without errors.  
Results: 20 rows returned in 4ms  
# 16. Select all records where the value of the City column starts with letter "a" and  
# ends with the letter "a".  
34 **ELECT \* FROM customers WHERE City LIKE "%aa%"**  
35 - 17. Select all records where the value of the City column does NOT start with the letter "a".  
36 **ELECT \* FROM customers WHERE City NOT LIKE "aa%"**  
37 - 18. Select all records where the second letter of the City is an "a".  
38 **ELECT \* FROM customers WHERE City LIKE "%a%"**

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68300	Germany
2	Bon app'	Laurence Lebihan	12 rue des Bouchers	Marseille	13008	France
3	17 brachenblut Delikatessen	Sven Ottlieb	Walserweg 21	Aachen	52066	Germany
4	18 Du monde entier	Jeanne Labrune	67 rue des Cinquante Otages	Nantes	44000	France
5	22 FISSA Fabrica Inter. Salchichas...	Diego Roel	C Monzalvar 85	Madrid	28034	Spain
6	26 Franco restoration	Carine Schmitt	54 rue Royale	Nantes	44000	France
7	29 Galeria del gastronomo	Eduardo Saavedra	Rambla de Catalunya 23	Barcelona	08022	Spain
...	...	...	...	...	...	...

Execution finished without errors.  
Results: 20 rows returned in 4ms  
# 16. Select all records where the second letter of the City is an "a".  
38 **ELECT \* FROM customers WHERE City LIKE "%a%"**

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
14	46 LILA-Supermercado	Carlos González	Carrera 52 con Ave. Bolívar #7-...	Bogotá	11101	Colombia
15	57 Paris spécialités	Marie Bertrand	265 boulevard Charonne	Paris	75012	France
16	59 Piccolo und mehr	Georg Röpke	Görlitzer 14	Salzburg	5020	Austria
17	60 Ronner y torrijos	Alejandra Camino	Gran Vía 1	Madrid	28001	Spain
18	74 Spécialités du monde	Dominique Perier	25 rue Lauriston	Paris	75016	France
19	75 Spät-Rai Beer & Ale	Art Braunschweiger	P.O. Box 555	Lander	83520	USA
20	91 Wolski	Zygmunt	ul. Filtry 68	Walla	01012	Poland

19. Select all records where the first letter of the City is an "a" or a "c" or an "s".

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Tools, Help, and a user icon. Below the menu are tabs for New Database, Open Database, MySQL Changes, Recent Changes, Open Project, Save Project, Alias Databases, and Close Databases. The main area has tabs for database structure, Home Data, List Progress, and Health SQL. A toolbar below the tabs includes icons for New, Open, Save, Print, Refresh, Undo, Redo, and Help. The central pane displays a query editor with the following SQL code:

```
34 SELECT * FROM customers WHERE City LIKE "%a%";  
35 10. Select all records where the value of the City column does NOT start with the letter "a".  
36 SELECT * FROM customers WHERE City NOT LIKE "%a%";  
37 10. Select all records where the second letter of the City is an "a".  
38 SELECT * FROM customers WHERE City LIKE "%a%a%";  
39 10. Select all records where the first letter of the City is an "a" or "c" or "e".  
40 SELECT * FROM customers WHERE City LIKE "%a%" OR City LIKE "%c%" OR City LIKE "%e%";
```

To the right of the query editor is a panel titled "Edit Database Cell" with a "Mode: Text" dropdown and a toolbar with icons for New, Open, Save, Print, Refresh, Undo, Redo, and Help. Below this is a message "Type of data currently in cell: NULL (Object)". At the bottom right is an "Apply" button. The bottom right corner of the interface shows the MySQL logo.

```
DB Browser for SQLite | C:\Users\asus\OneDrive\Documents\sql\ejercicios\lab

File Edit View Tools Help
New Database Open Database [Will Change] Reset Changes Open Project Save Project Attached Database Close Database
Database Browser Rows (1) Filter (empty) Execute SQL
[ Home ] [ Login ] [ Logout ] [ Help ]

34. SELECT * FROM customers WHERE CITY LIKE "%aa";
35 -- 19. Select all records where the value of the City column does NOT start with the letter "a".
36 SELECT * FROM customers WHERE CITY NOT LIKE "%a";
37 -- 20. Select all records where the first letter of the City is an "a".
38 SELECT * FROM customers WHERE CITY LIKE "a%";
39 -- 19. Select all records where the first letter of the City is an "a" or a "c" or an "s".
40 SELECT * FROM customers WHERE CITY LIKE "%a%" OR CITY LIKE "%c%" OR CITY LIKE "%s%";

CustomerID CustomerName ContactName Address City PostalCode Country
1 7 Blondel père et fils Frédéric Lebeau 24 place Kléber Strasbourg 67000 France
2 15 Comércio Mineiro Pedro Afonso Av. dos Lutitas 23 São Paulo 05430450 Brazil
3 17 Drachenfels Delicatessen Sven Ottie Walserweg 21 Aachen 52066 Germany
4 21 Familia Arquillo José Cruz Rua Orfeu 92 São Paulo 05440010 Brazil
5 30 Godos Cocaína Típica José Pedro Freyre C Romero 33 Sevilla 41101 Spain
6 31 Gourmet Lanchonetes André Fonseca Av. Brasil 442 Campinas 04876796 Brazil
7 33 GROSELLA-Restaurante Manuel Pereira 5º Ave. Los Palados Caracas 1081 Venezuela

SELECT * FROM customers WHERE CITY LIKE "%aa%" OR CITY LIKE "%c%" OR CITY LIKE "%s%"
```

```
File Edit View Tools Help
New Database Open Database <| Write Changes <| New Changes Open Project Save Project <| Match Database <| Close Database
Database Structure Browser Data Edit Phrases Execute SQL
File Edit View Tools Help
New Database Open Database <| Write Changes <| New Changes Open Project Save Project <| Match Database <| Close Database
Edit Database Cell
Mode: text <| 
NULL
Type of data currently in cell: NULL
[!empty] Apply
51 --SELECT * FROM customers WHERE CITY LIKE "%a%" -- all records where the City column does NOT start with the letter "a".
52 --SELECT * FROM customers WHERE CITY NOT LIKE "%a%" -- all records where the second letter of the City is an "%".
53 -- 10. Selected all records where the second letter of the City is an "%".
54 --SELECT * FROM customers WHERE CITY LIKE "%%" -- all records where the first letter of the City is an "a" or a "t" or an "s".
55 -- 10. Selected all records where the first letter of the City is an "a" or a "t" or an "s".
56 --SELECT * FROM customers WHERE CITY LIKE "%a%" OR CITY LIKE "%t%" OR CITY LIKE "%s%"
```

20. Select all records where the first letter of the City starts with anything from an "a" to an "f".

SQL Browser for SQLite - C:\Users\jude\Hunternet\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Import Changes Export Changes Open Project Save Project Attach Database Close Database

Database Status: In Use Data List Tables Recent SQL

Mode: Text

SQL Log Flat DB Schema Results

16. SELECT \* FROM customers WHERE city NOT LIKE "%a%"  
-- 16. select all records where the second letter of the City is an "a".  
17. -- 17. select all records where the value of the City column does NOT start with the letter "a".  
18. SELECT \* FROM customers WHERE city NOT LIKE "%c%"  
-- 18. select all records where the first letter of the City is an "c" or a "C" or an "e".  
19. SELECT \* FROM customers WHERE city LIKE "%e%" OR city LIKE "%E%" OR city LIKE "%c%"  
-- 19. select all records where the first letter of the City starts with anything from an "e" to an "f".  
20. SELECT \* FROM customers WHERE city < "%y"

CustomerID CustomerName ContactName Address City PostalCode Country  
1 17 Drachenfels Delikatessen Sven Ottie Walzenweg 21 Aachen 52066 Germany  
2 55 Old World Delicatessen Rene Phillips 2743 Bering St. Anchorage 99508 USA  
3 65 Rattensack Canyon Grocery Paula Wilson 2817 Hilton Dr. Albuquerque 87110 USA

Execution finished without errors.  
Execution time was 0.000 seconds.  
at line 11  
SELECT \* FROM customers WHERE city like "%a%"

SQL Browser for SQLite - C:\Users\jude\Hunternet\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Import Changes Export Changes Open Project Save Project Attach Database Close Database

Database Status: In Use Data List Tables Recent SQL

Mode: Text

SQL Log Flat DB Schema Results

16. SELECT \* FROM customers WHERE city NOT LIKE "%a%"  
-- 16. select all records where the second letter of the City is an "a".  
17. -- 17. select all records where the value of the City column does NOT start with the letter "a".  
18. SELECT \* FROM customers WHERE city NOT LIKE "%c%"  
-- 18. select all records where the first letter of the City is an "c" or a "C" or an "e".  
19. SELECT \* FROM customers WHERE city LIKE "%e%" OR city LIKE "%E%" OR city LIKE "%c%"  
-- 19. select all records where the first letter of the City starts with anything from an "e" to an "f".  
20. SELECT \* FROM customers WHERE city < "%y"

CustomerID CustomerName ContactName Address City PostalCode Country  
1 8 Bistro Corridos preparados Martin Sommer Oberstraße 57 Berlin 12209 Germany  
2 9 Bistro Corridos preparados Martin Sommer Oberstraße 57 Berlin 12209 Spain  
3 12 Cactus Comidas para levar Patrício Simpson Cemte 3531 Buenos Aires 1010 Argentina  
4 14 Chop Suey Chinese Yang Wang Haussner 29 Bern 3012 Switzerland  
5 17 Drachenfels Delikatessen Sven Ottie Walzenweg 21 Aachen 52066 Germany  
6 24 Fols om & HJ Maria Larsson Åbergatan 24 Brölke 5044 67 Sweden  
7 29 Galeria del gastronomia Eduardo Saavedra Rambla de Cataluña 23 Barcelona 08022 Spain

Execution finished without errors.  
Execution time was 0.000 seconds.  
at line 11  
SELECT \* FROM customers WHERE city like "%a%"  
SELECT \* FROM customers WHERE city < "%y"

SQL Browser for SQLite - C:\Users\jude\Hunternet\Documents\sqlite\mydb.db

File Edit View Tools Help

New Database Open Database Import Changes Export Changes Open Project Save Project Attach Database Close Database

Database Status: In Use Data List Tables Recent SQL

Mode: Text

SQL Log Flat DB Schema Results

16. SELECT \* FROM customers WHERE city NOT LIKE "%a%"  
-- 16. select all records where the second letter of the City is an "a".  
17. -- 17. select all records where the value of the City column does NOT start with the letter "a".  
18. SELECT \* FROM customers WHERE city NOT LIKE "%c%"  
-- 18. select all records where the first letter of the City is an "c" or a "C" or an "e".  
19. SELECT \* FROM customers WHERE city LIKE "%e%" OR city LIKE "%E%" OR city LIKE "%c%"  
-- 19. select all records where the first letter of the City starts with anything from an "e" to an "f".  
20. SELECT \* FROM customers WHERE city < "%y"

CustomerID CustomerName ContactName Address City PostalCode Country  
19 51 Old World Delicatessen Rene Phillips 2743 Bering St. Anchorage 99508 USA  
20 63 QUICK Step Horst Kloss Tauchentstraße 10 Cunewalde 01307 Germany  
21 64 Rancho grande Sergio Gutierrez Av. del Libertador 900 Buenos Aires 1010 Argentina  
22 65 Rattensack Canyon Grocery Paula Wilson 2817 Hilton Dr. Albuquerque 87110 USA  
23 71 Saverクト Market Jose Pavanotti 187 Suffolk Ln. Boise 83720 USA  
24 75 Suprèmes délices Pascale Cartain Boulevard Thru 255 Charlotte 88600 Belgium  
25 78 The Cracker Box Liu Wong 55 Grizzly Peak Rd. Butte 59801 USA

Execution finished without errors.  
Execution time was 0.000 seconds.  
at line 11  
SELECT \* FROM customers WHERE city like "%a%"  
SELECT \* FROM customers WHERE city < "%y"

## 21. Select all records where the first letter of the City is NOT an "a" or a "c" or an "f".

18. Run SQL in MySQL Workbench:

```

19. SELECT * FROM customers WHERE City LIKE "%a";
20. -- 19. Select all records where the first letter of the City is an "a" or a "c" or an "s".
21. SELECT * FROM customers WHERE City LIKE "%a" OR City LIKE "%c" OR City LIKE "%s";
22. -- 20. Select all records where the first letter of the City starts with anything from an "a" to an "f".
23. SELECT * FROM customers WHERE City LIKE "%a%" OR City LIKE "%c%" OR City LIKE "%s%";
24. -- 21. Select all records where the first letter of the City is NOT an "a" or a "c" or an "f".
25. SELECT * FROM customers WHERE City NOT LIKE "%a%" OR City NOT LIKE "%c%" OR City NOT LIKE "%s%";
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
19	55 Old World Delicatessen	Rene Phillips	2743 Bering St.	Anchorage	99508	USA
20	63 QUICK Stop	Hans Kloss	Tauherstrasse 10	Cunevalpe	9740	Germany
21	64 Rancho grande	Sergio Gutierrez	Av. del Libertador 900	Buenos Aires	1010	Argentina
22	65 Rattlesnake Canyon Grocery	Paula Wilson	2617 Milton Dr.	Albuquerque	87110	USA
23	71 Saverio's Markets	Jose Pavarotti	187 Saffolk Ln.	Bose	83720	USA
24	76 Suprimes delicatessen	Pascalle Cartain	Boulevard Timo 255	Charieroi	68600	Belgium
25	78 The Cracker Box	Liu Wong	55 Grizzly Peak Rd.	Butte	59801	USA

Execution finished without errors.  
Result: 12 rows returned in 1ms  
At line 11:  
-- 20. Select all records where the first letter of the City starts with anything from an "a" to an "f".  
SELECT \* FROM customers WHERE City LIKE "%a%" OR City LIKE "%c%" OR City LIKE "%s%"

19. Run SQL in MySQL Workbench:

```

20. SELECT * FROM customers WHERE City LIKE "%a";
21. -- 19. Select all records where the first letter of the City is an "a" or a "c" or an "s".
22. SELECT * FROM customers WHERE City LIKE "%a%" OR City LIKE "%c%" OR City LIKE "%s%";
23. -- 20. Select all records where the first letter of the City starts with anything from an "a" to an "f".
24. SELECT * FROM customers WHERE City != "a" OR City != "c" OR City != "s";
25. -- 21. Select all records where the first letter of the City is NOT an "a" or a "c" or an "f".
26. SELECT * FROM customers WHERE City != "%a%" OR City != "%c%" OR City != "%s%";
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Empedrado y ...	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mabedders 2312	México D.F.	05023	Mexico
4	4 Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	5 Berglunds snabbköp	Christine Berglund	Berguvsgatan 8	Luleå	99582	Sweden
6	6 Blauer See Delikatessen	Hanna Moos	Fosdtert 57	Mannheim	68306	Germany
7	7 Blondel père et fils	Frédérique Citeaux	24 place Kléber	Strasbourg	67000	France

Execution finished without errors.  
Result: 81 rows returned in 1ms  
At line 20:  
-- 20. Select all records where the first letter of the City is NOT an "a" or a "c" or an "s".  
SELECT \* FROM customers WHERE City != "%a%" OR City != "%c%" OR City != "%s%"

20. Run SQL in MySQL Workbench:

```

21. SELECT * FROM customers WHERE City LIKE "%a";
22. -- 19. Select all records where the first letter of the City is an "a" or a "c" or an "s".
23. SELECT * FROM customers WHERE City LIKE "%a%" OR City LIKE "%c%" OR City LIKE "%s%";
24. -- 20. Select all records where the first letter of the City starts with anything from an "a" to an "f".
25. -- 21. Select all records where the first letter of the City is NOT an "a" or a "c" or an "f".
26. SELECT * FROM customers WHERE City != "%a%" OR City != "%c%" OR City != "%s%";
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
85	85 Vins et alcools Chevalier	Paul Henrot	59 rue de l'Abbaye	Reims	51100	France
86	86 Die Wunderbare Kuh	Rita Müller	Adeauerallee 900	Stuttgart	70567	Germany
87	87 Wartian Herku	Pirkko Koskitalo	Toriukatu 38	Oulu	80110	Finland
88	88 Wellington Importadora	Julião Parente	Rua do Mercado 12	Resende	68772-983	Brazil
89	89 White Cover Parkers	Karl-Josef Spök	305 14th Ave. S. Suite 3B	Seattle	98103	USA
90	90 Wilman Käsi	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	91 World	Zbynek	ul. Piłsudskiego 66	Walla	01012	Poland

Execution finished without errors.  
Result: 12 rows returned in 1ms  
At line 21:  
-- 21. Select all records where the first letter of the City is NOT an "a" or a "c" or an "s".  
SELECT \* FROM customers WHERE City != "%a%" OR City != "%c%" OR City != "%s%"

## 22. Select all the records where Country is either "Norway" or "France".

21. Run SQL in MySQL Workbench:

```

22. SELECT * FROM customers WHERE Country = "Norway" OR Country = "France";
23. -- 21. Select all the records where the first letter of the City starts with anything from an "a" to an "f".
24. SELECT * FROM customers WHERE City LIKE "%a";
25. -- 21. Select all records where the first letter of the City is NOR an "a" or a "c" or an "f".
26. SELECT * FROM customers WHERE City != "%a%" OR City != "%c%" OR City != "%s%";
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	7 Blondel père et fils	Frédérique Citeaux	24 place Kléber	Strasbourg	67000	France
2	9 Bon app!	Laurence Lebihan	12 rue des Bouchers	Marseille	13008	France
3	18 Du monde entier	Jeanne Labrunie	67 rue des Chiquets Etages Nantes	Nantes	44000	France
4	23 Polles gourmandises	Martine Rancé	184 chaussée de Tournai	Lille	59000	France
5	26 France restauration	Carine Schmitt	54 rue Royale	Nantes	44000	France
6	40 La come d'abondance	Daniel Tonini	67 avenue de l'Europe	Versailles	78000	France
7	41 La maison d'Ase	Annette Roulet	1 rue Alsace-Lorraine	Toulouse	31000	France

Execution finished without errors.  
Result: 12 rows returned in 1ms  
At line 22:  
-- 22. Select all the records where Country is either "Norway" or "France".  
SELECT \* FROM customers WHERE Country IN("Norway","France")

## 23. Select all the records where Country is NOT "Norway" and NOT "France".

DB Browser for SQLite - C:\Users\batin\OneDrive\Documents\sqlite3methods

File Edit Tools Help

Open Database Write Changes Abort Changes Open Project Close Project Attach Database Close Database

Database Structure Insert Data Full Dumper Export SQL

CustomerID CustomerName ContactName Address City PostalCode Country

1 2 Blouin père et fils Frédéric Cléauz 24 place Kléber Strasbourg 67000 France

2 9 Bon app! Laurence Lebihan 12 rue des Bouchers Marseille 13008 France

3 19 Du monde entier Janine Labrune 67 rue des Croquante Oignes Nantes 44000 France

4 21 Fables gourmands Marine Ranch 164 chaussée de Tournai Lille 59000 France

5 26 France restauration Carine Schmitt 54 rue Royale Nantes 44000 France

6 40 La corne d'abondance Daniel Tonin 67 avenue d'Europe Vénissieux 69100 France

7 41 Le maison Alésia Arlette Poulet 1 rue Alsace-Lorraine Toulouse 31000 France

Execution finished without errors.  
Result: 7 rows returned in 1ms.

At line 45:  
-- 21. Select all records where the first letter of the city is NOT as "a" or a "c" or an "f".  
SELECT \* FROM customers WHERE Country IN("Norway","France")  
-- 22. Select all the records where Country is either "Norway" or "France".  
SELECT \* FROM customers WHERE Country IN("Norway","France")  
-- 23. Select all the records where Country is NOT "Norway" and NOT "France".  
SELECT \* FROM customers WHERE Country NOT IN("Norway","France")

DB Schema

Type of column currently in cell: NULL

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log PKI DB Schema Remote

DB Browser for SQLite - C:\Users\batin\OneDrive\Documents\sqlite3methods

File Edit Tools Help

Open Database Write Changes Abort Changes Open Project Close Project Attach Database Close Database

Database Structure Insert Data Full Dumper Export SQL

CustomerID CustomerName ContactName Address City PostalCode Country

1 1 Alfreds Futterkiste Maria Anders Obere Dr. 57 Berlin 12209 Germany

2 2 Avo Träfflo Empredades, Y. Ana Träfflo Avda. de la Constitución 2222 México D.F. 04921 Mexico

3 3 Antonio Moreno Taquería Antonio Moreno Matadero 2312 México D.F. 05023 Mexico

4 4 Around the Horn Thomas Hardy 120 Hanover Sq. London W1A 1DP UK

5 5 Berglunds snabbköp Chrstina Berglund Berguvagen 8 Luleå 993 22 Sweden

6 6 Bauer See Delikassen Hanna Moos Forststr. 57 Mannheim 68306 Germany

7 7 Bólido Convenios preparados Karin Sommer C Anselm 67 Madrid 28023 Spain

Execution finished without errors.  
Result: 7 rows returned in 1ms.

At line 45:  
-- 21. Select all the records where Country is NOT "Norway" and NOT "France".  
SELECT \* FROM customers WHERE Country NOT IN("Norway","France")  
-- 22. Select all the records where Country is either "Norway" or "France".  
SELECT \* FROM customers WHERE Country IN("Norway","France")  
-- 23. Select all the records where Country is NOT "Norway" and NOT "France".  
SELECT \* FROM customers WHERE Country NOT IN("Norway","France")

DB Schema

Type of column currently in cell: NULL

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log PKI DB Schema Remote

DB Browser for SQLite - C:\Users\batin\OneDrive\Documents\sqlite3methods

File Edit Tools Help

New Database Open Database Write Changes Abort Changes Open Project Close Project Attach Database Close Database

Database Structure Insert Data Full Dumper Export SQL

CustomerID CustomerName ContactName Address City PostalCode Country

73 15 Lydia Cofrance Paul Sverre Smøringen 40 Ålesund 6000 Norway

74 88 Öre Handelende Kuh Rita Müller Adressen 800 Stuttgart 70563 Germany

75 87 Wistant Herku Pirkko Koskitalo Tokianniemi 36 Oulu 90110 Finland

76 88 Wellington Importadora Paula Pimenta Rue do Morrodo 12 Resende 08237 363 Brazil

77 89 White Clover Markets Karjalainen Matti Karttunen Kesäkatu 45 Helsinki 00120 USA

78 90 Willow Kale Matti Karttunen Kesäkatu 45 Helsinki 00120 Finland

79 91 Woski Zbyszek ul. Filtrowa 56 Wola 01012 Poland

Execution finished without errors.  
Result: 79 rows returned in 1ms.

At line 45:  
-- 21. Select all the records where Country is NOT "Norway" and NOT "France".  
SELECT \* FROM customers WHERE Country NOT IN("Norway","France")  
-- 22. Select all the records where Country is either "Norway" or "France".  
SELECT \* FROM customers WHERE Country IN("Norway","France")  
-- 23. Select all the records where Country is NOT "Norway" and NOT "France".  
SELECT \* FROM customers WHERE Country NOT IN("Norway","France")

DB Schema

Type of column currently in cell: NULL

DB Schema

Name Type Schema

- Tables (2)
  - customers CREATE
  - products CREATE
- Indices (0)
- Views (0)
- Triggers (0)

SQL Log PKI DB Schema Remote

## 24. List the number of customers in each country.

DB Browser for SQLite - C:\Users\Adam\Downloads\customer\dbs\customer.db

File Edit View Tools Help

Databases Structure Tables Columns Views Changes Scripts Projects New Project Attach Database Close Database

SQL Log SQL Database Remote

```

44 SELECT * FROM customers WHERE city != "em" OR city != "("*
45 -- 22. Select all the records where Country is either "Norway" or "France".
46 SELECT * FROM customers WHERE Country IN("Norway","France")*
47 -- 23. Select all the records where Country is NOT "Norway" and NOT "France".
48 SELECT * FROM customers WHERE Country NOT IN("Norway","France")*
49 -- 24. List the number of customers in each country.
50 SELECT Country, COUNT(*) FROM customers GROUP BY Country

```

CustomerID	COUNT(*)
16	28
17	8
18	5
19	9
20	14
21	7
22	32
23	13
24	4

Preparation started without errors.

Wrote 16 rows to result.

As: customer

-- 21. List the number of customers in each country.

-- 22. List the number of customers in each country.

-- 23. List the number of customers in each country.

-- 24. List the number of customers in each country.

DB Browser for SQLite - C:\Users\Adam\Downloads\customer\dbs\customer.db

File Edit View Tools Help

Databases Structure Tables Columns Views Changes Scripts Projects New Project Attach Database Close Database

SQL Log SQL Database Remote

```

44 SELECT * FROM customers WHERE City != "em" OR City != "("*
45 -- 22. Select all the records where Country is either "Norway" or "France".
46 SELECT * FROM customers WHERE Country IN("Norway","France")*
47 -- 23. Select all the records where Country is NOT "Norway" and NOT "France".
48 SELECT * FROM customers WHERE Country NOT IN("Norway","France")*
49 -- 24. List the number of customers in each country.
50 SELECT Country, COUNT(*) FROM customers GROUP BY Country

```

Country	COUNT(*)
Argentina	3
Austria	2
Belgium	9
Brazil	3
Canada	3
Denmark	2
Finland	2
France	4

Preparation started without errors.

Wrote 16 rows to result.

As: customer

-- 21. List the number of customers in each country.

-- 22. List the number of customers in each country.

-- 23. List the number of customers in each country.

-- 24. List the number of customers in each country.

DB Browser for SQLite - C:\Users\Adam\Downloads\customer\dbs\customer.db

File Edit View Tools Help

Databases Structure Tables Columns Views Changes Scripts Projects New Project Attach Database Close Database

SQL Log SQL Database Remote

```

44 SELECT * FROM customers WHERE city != "em" OR city != "("*
45 -- 22. Select all the records where Country is either "Norway" or "France".
46 SELECT * FROM customers WHERE Country IN("Norway","France")*
47 -- 23. Select all the records where Country is NOT "Norway" and NOT "France".
48 SELECT * FROM customers WHERE Country NOT IN("Norway","France")*
49 -- 24. List the number of customers in each country.
50 SELECT Country, COUNT(*) FROM customers GROUP BY Country

```

Country	COUNT(*)
Portugal	2
Spain	5
Sweden	2
Switzerland	2
UK	7
USA	17
Venezuela	4

Preparation started without errors.

Wrote 16 rows to result.

As: customer

-- 21. List the number of customers in each country.

-- 22. List the number of customers in each country.

-- 23. List the number of customers in each country.

-- 24. List the number of customers in each country.

25. List the number of customers in each country, ordered by the country with the most customers first.

DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db

```

File Edit View Tools Help
New Database Open Database Edit Changes Event Changes Open Project Save Project Attach Database Close Database
Database Explorer Browse SQL Edit Query Locate SQL
DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db
Edit Database Cell
Mode: Text
1 Argentina
Type of data currently in cell: Text
Normal Standard Apply
DB Schema
Name Type Schema
Tables (2)
Customers CREATE
Products CREATE
Indices (0)
Views (0)
Triggers (0)
SQL Log Pwd DB Schema Rename

```

```

16 SELECT * FROM customers WHERE Country IN("Norway", "France");
17 -- 24. List the number of customers where Country is NOT "Norway" and NOT "France".
18 SELECT * FROM customers WHERE Country NOT IN("Norway", "France");
19 -- 24. List the number of customers in each country.
20 SELECT Country, COUNT(*) FROM customers GROUP BY Country;
21 -- 24. List the number of customers in each country, ordered by the country with the most customers first.
22 SELECT Country, COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;

```

Country	COUNT(*)
1 USA	13
2 Germany	11
3 France	11
4 Brazil	9
5 UK	7
6 Spain	5
7 Venezuela	4

Execution finished without errors.  
Rows affected: 22 rows returned in 4ms.

At Line 22, 22 rows returned.  
-- 24. List the number of customers in each country, ordered by the country with the most customers first.  
SELECT Country, COUNT(\*) FROM customers GROUP BY Country ORDER BY COUNT(\*) DESC;

DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db

```

File Edit View Tools Help
New Database Open Database Edit Changes Event Changes Open Project Save Project Attach Database Close Database
Database Explorer Browse SQL Edit Query Locate SQL
DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db
Edit Database Cell
Mode: Text
NULL
Type of data currently in cell: NULL
Normal Standard Apply
DB Schema
Name Type Schema
Tables (2)
Customers CREATE
Products CREATE
Indices (0)
Views (0)
Triggers (0)
SQL Log Pwd DB Schema Rename

```

```

16 SELECT * FROM customers WHERE Country IN("Norway", "France");
17 -- 24. List the number of customers where Country is NOT "Norway" and NOT "France".
18 SELECT * FROM customers WHERE Country NOT IN("Norway", "France");
19 -- 24. List the number of customers in each country.
20 SELECT Country, COUNT(*) FROM customers GROUP BY Country;
21 -- 24. List the number of customers in each country, ordered by the country with the most customers first.
22 SELECT Country, COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;

```

Country	COUNT(*)
1 USA	13
2 Germany	11
3 France	11
4 Brazil	9
5 UK	7
6 Spain	5
7 Venezuela	4

Execution finished without errors.  
Rows affected: 22 rows returned in 4ms.

At Line 22, 22 rows returned.  
-- 24. List the number of customers in each country, ordered by the country with the most customers first.  
SELECT Country, COUNT(\*) FROM customers GROUP BY Country ORDER BY COUNT(\*) DESC;

DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db

```

File Edit View Tools Help
New Database Open Database Edit Changes Event Changes Open Project Save Project Attach Database Close Database
Database Explorer Browse SQL Edit Query Locate SQL
DB Browser for SQLite - C:\Users\Jatin\OneDrive\Documents\sqlite\dbs\customers.db
Edit Database Cell
Mode: Text
NULL
Type of data currently in cell: NULL
Normal Standard Apply
DB Schema
Name Type Schema
Tables (2)
Customers CREATE
Products CREATE
Indices (0)
Views (0)
Triggers (0)
SQL Log Pwd DB Schema Rename

```

```

16 SELECT * FROM customers WHERE Country IN("Norway", "France");
17 -- 24. List the number of customers where Country is NOT "Norway" and NOT "France".
18 SELECT * FROM customers WHERE Country NOT IN("Norway", "France");
19 -- 24. List the number of customers in each country.
20 SELECT Country, COUNT(*) FROM customers GROUP BY Country;
21 -- 24. List the number of customers in each country, ordered by the country with the most customers first.
22 SELECT Country, COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;

```

Country	COUNT(*)
16 Denmark	2
17 Belgium	2
18 Austria	2
19 Poland	1
20 Norway	1
21 Mexico	1
22 Ireland	1

Execution finished without errors.  
Rows affected: 22 rows returned in 4ms.

At Line 22, 22 rows returned.  
-- 24. List the number of customers in each country, ordered by the country with the most customers first.  
SELECT Country, COUNT(\*) FROM customers GROUP BY Country ORDER BY COUNT(\*) DESC;

## 26. Select the record with the smallest Price of products.

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sqlite\mydb.db

```

16 SELECT * FROM customers WHERE Country NOT IN("Norway", "France");
17 -- 24. List the number of customers in each country.
18 SELECT COUNT(*) FROM customers GROUP BY Country;
19 -- 25. List the number of customers in each country, ordered by the country with the most customers first.
20 SELECT Country, COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;
21 -- 26. Select the record with the smallest Price of products.
22 SELECT * FROM products WHERE Price = (SELECT MIN(Price) FROM products);
23

```

MIN(Price)	1	2.5
------------	---	-----

Execution finished without errors.  
Total 1 row returned in 0ms  
At line 23:  
-- 26. select the record with the smallest Price of products.  
SELECT \* FROM products WHERE Price = (SELECT MIN(Price) FROM products);

DB Schema

- Name Type Schema
  - Tables (2)
    - customers CREATE
    - products CREATE
  - Indices (0)
  - Views (0)
  - Triggers (0)

DB Log Plot DB Schema Remote

## 27. Return the number of records that have the Price value set to 18.

DB Browser for SQLite - C:\Users\Justin\OneDrive\Documents\sqlite\mydb.db

```

16 SELECT * FROM customers WHERE Country NOT IN("Norway", "France");
17 -- 24. List the number of customers in each country.
18 SELECT COUNT(*) FROM customers GROUP BY Country;
19 -- 25. List the number of customers in each country, ordered by the country with the most customers first.
20 SELECT Country, COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;
21 -- 26. Select the record with the smallest Price of products.
22 SELECT * FROM products WHERE Price = (SELECT MIN(Price) FROM products);
23 -- 27. Returns the number of records that have the Price value set to 18.
24 SELECT COUNT(*) FROM products WHERE Price = 18;
25

```

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	33 Gelato	15	15	4500 g	2.5

Execution finished without errors.  
Total 1 row returned in 0ms  
At line 25:  
-- 27. Returns the number of records that have the Price value set to 18.  
SELECT COUNT(\*) FROM products WHERE Price = 18;

DB Schema

- Name Type Schema
  - Tables (2)
    - customers CREATE
    - products CREATE
  - Indices (0)
  - Views (0)
  - Triggers (0)

DB Log Plot DB Schema Remote

## 28. Calculate the average Price of all products.

The screenshot shows two separate database windows side-by-side. Both windows have the title 'DB Browser for SQLite - C:\Users\Umar\OneDrive\Documents\sqlite\mydb'. The left window contains the following SQL code:

```

SELECT COUNT(*) FROM customers GROUP BY Country ORDER BY COUNT(*) DESC;
-- 24. Get the records with the most number of products.
SELECT * FROM products WHERE Price = (SELECT MIN(Price) FROM products);
-- 25. Return the number of records that have the Price value set to 10.
SELECT COUNT(*) FROM products WHERE Price = 10;
-- 26. Calculate the average Price of all products.
SELECT ProductName, AVG(UnitPrice) FROM products GROUP BY ProductName;

```

The results of the first query are displayed in a table:

ProductName	Avg(UnitPrice)
Tofu	23.25
Tourtiere	7.45
Tumblerid	9.0
Uncle Bob's Organic Dried Pears	30.0
Valkonen sukaa	16.25
Vege-spread	43.9
Wimmers gute Semmelknödel	33.25
... (truncated)	...

The right window contains the same SQL code, with the results table being empty.

## 29. Calculate the sum of all the Price column values in the Products table.

The screenshot shows three separate database windows side-by-side. The leftmost window has the title 'DB Browser for SQLite - C:\Users\Umar\OneDrive\Documents\sqlite\mydb'. It contains the following SQL code:

```

SELECT * FROM products WHERE Price=(SELECT MIN(Price) FROM products);
-- 25. Returns the number of records that have the Price value set to 10.
SELECT COUNT(*) FROM products WHERE Price = 10;
-- 26. Calculate the average Price of all products.
SELECT ProductName, AVG(UnitPrice) FROM products GROUP BY ProductName;
-- 27. Calculate the sum of all the Price column values in the Products table.
SELECT SUM(UnitPrice) FROM products;

```

The results of the first query are displayed in a table:

ProductName	Avg(UnitPrice)
Tofu	23.25
Tourtiere	7.45
Tumblerid	9.0
Uncle Bob's Organic Dried Pears	30.0
Valkonen sukaa	16.25
Vege-spread	43.9
Wimmers gute Semmelknödel	33.25
... (truncated)	...

The middle window has the title 'DB Browser for SQLite - C:\Users\Umar\OneDrive\Documents\sqlite\mydb'. It contains the same SQL code, with the results table being empty.

The rightmost window has the title 'DB Browser for SQLite - C:\Users\Umar\OneDrive\Documents\sqlite\mydb'. It contains the following SQL code:

```

SELECT * FROM products WHERE Price=(SELECT MIN(Price) FROM products);
-- 27. Returns the number of records that have the Price value set to 10.
SELECT COUNT(*) FROM products WHERE Price = 10;
-- 28. Calculate the average Price of all products.
SELECT ProductName, AVG(UnitPrice) FROM products GROUP BY ProductName;
-- 29. Calculate the sum of all the Price column values in the Products table.
SELECT SUM(UnitPrice) FROM products;

```

The results of the last query are displayed in a table:

Sum(UnitPrice)
2222.71

**30. Select product names and prices and also display a price type column to mark prices higher than 50 as high and price lower than 50 as low.**

```

59  --> 39. Calculate the sum of all the Price column values in the Products Table.
59  SELECT SUM(Price) FROM products;
60  --> 30. Select product names and prices and also display a price type column
61  --> to mark prices higher than 50 as high and price lower than 50 as low.
62  --> 33. SELECT ProductName, Price,
63  --> CASE
64  --> WHEN Price >= 50 THEN "High"
65  --> WHEN Price <= 50 THEN "Low"
66  --> END AS "Price Type"
67  --> FROM products GROUP BY ProductName;
68

```

Execution finished without errors.  
Results: 77 rows returned in 3ms  
SQL Log | SQL | DB Schema | Remote

ProductName	Price	Price Type
Alice Mutton	39	Low
Aniseed Syrup	10	Low
Boston Crab Meat	18.4	Low
Camembert Pierrot	34	Low
Cannanom Tigers	62.5	High
Chais	18	Low
Chang	19	Low

Execution finished without errors.  
Results: 77 rows returned in 3ms  
SQL Log | SQL | DB Schema | Remote

ProductName	Price	Price Type
Alice Mutton	39	Low
Aniseed Syrup	10	Low
Boston Crab Meat	18.4	Low
Camembert Pierrot	34	Low
Cannanom Tigers	62.5	High
Chais	18	Low
Chang	19	Low

Execution finished without errors.  
Results: 77 rows returned in 3ms  
SQL Log | SQL | DB Schema | Remote

ProductName	Price	Price Type
Tourtiere	7.45	Low
Tunabild	9	Low
Uncle Bob's Organic Dried Pears	30	Low
Valkonen saukko	16.25	Low
Vegie-spread	43.9	Low
Wimmers gute Semmelknödel	33.25	Low
Zaanse koeken	9.5	Low