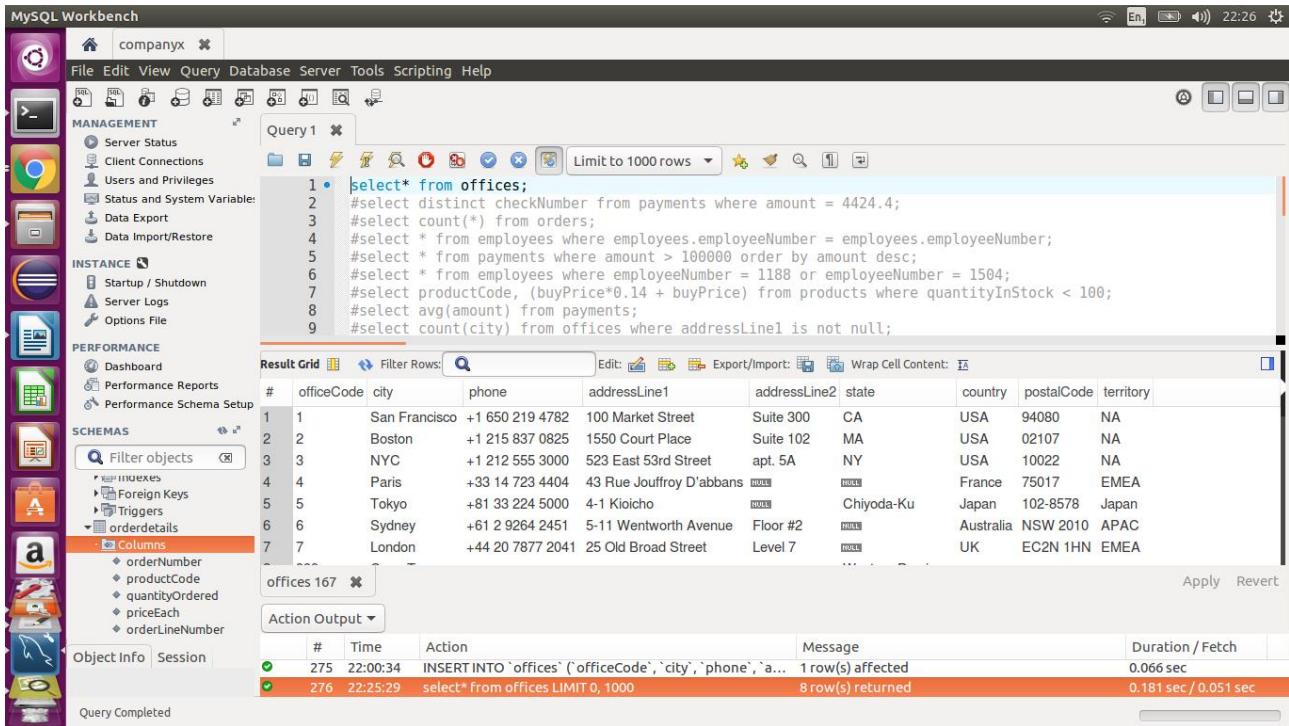


- 1) Show all information in the **offices** relation.

select* from offices; #1 show all information in office table

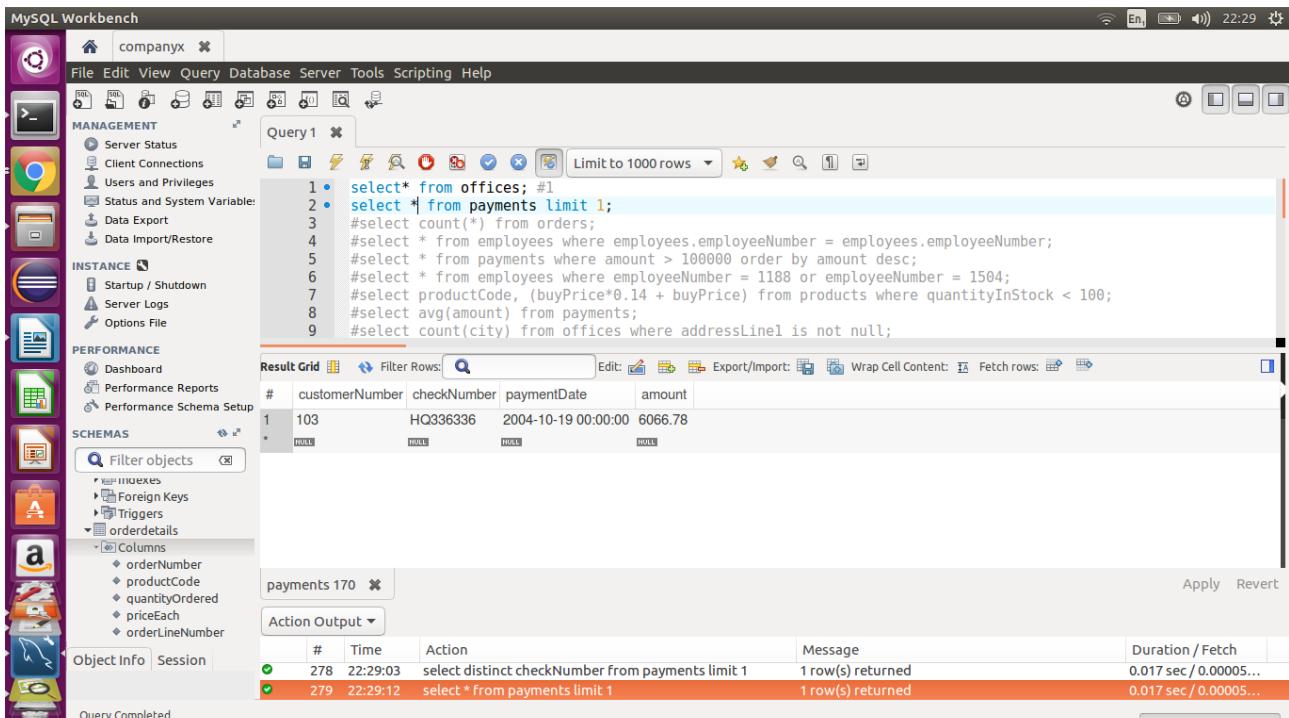


The screenshot shows the MySQL Workbench interface with the 'companyx' database selected. In the left sidebar, under 'SCHEMAS', the 'orderdetails' schema is expanded, and the 'Columns' tab is selected. The main area displays a query results grid for the 'offices' table. The grid has columns: #, officeCode, city, phone, addressLine1, addressLine2, state, country, postalCode, and territory. The data shows 7 rows of office information across various cities like San Francisco, Boston, NYC, Paris, Tokyo, Sydney, and London, with details like phone numbers and addresses. Below the grid is an 'Action Output' table showing two recent actions: an insertion into 'offices' and a query execution.

#	officeCode	city	phone	addressLine1	addressLine2	state	country	postalCode	territory
1	1	San Francisco	+1 650 219 4782	100 Market Street	Suite 300	CA	USA	94080	NA
2	2	Boston	+1 215 837 0825	1550 Court Place	Suite 102	MA	USA	02107	NA
3	3	NYC	+1 212 555 3000	523 East 53rd Street	apt. 5A	NY	USA	10022	NA
4	4	Paris	+33 14 723 4404	43 Rue Joffroy D'abbans	NULL	NULL	France	75017	EMEA
5	5	Tokyo	+81 33 224 5000	4-1 Kioicho	NULL	Chiyoda-Ku	Japan	102-8578	Japan
6	6	Sydney	+61 2 9264 2451	5-11 Wentworth Avenue	Floor #2	NULL	Australia	NSW 2010	APAC
7	7	London	+44 20 7877 2041	25 Old Broad Street	Level 7	NULL	UK	EC2N 1HN	EMEA

- 2) Show any one tuple in the **payments** relation (just one, no more).

select * from payments limit 1; #2 show any row in payments



The screenshot shows the MySQL Workbench interface with the 'companyx' database selected. In the left sidebar, under 'SCHEMAS', the 'orderdetails' schema is expanded, and the 'Columns' tab is selected. The main area displays a query results grid for the 'payments' table. The grid has columns: #, customerNumber, checkNumber, paymentDate, and amount. A single row is shown for customer number 103 with a check number of HQ336336, payment date of 2004-10-19, and amount of 6066.78. Below the grid is an 'Action Output' table showing two recent actions: a distinct check number selection and a general query execution.

#	customerNumber	checkNumber	paymentDate	amount
1	103	HQ336336	2004-10-19 00:00:00	6066.78

- 3) Show how many tuples there are in the **orders** relation.

select count(*) from orders; #3 count how many rows in orders table

The screenshot shows the MySQL Workbench interface with a database named 'companyx'. In the left sidebar, under 'SCHEMAS', the 'orderdetails' table is selected, and its columns are listed: orderNumber, productCode, quantityOrdered, priceEach, and orderLineNumber. The main window contains a query editor with the following SQL code:

```
1 • select * from offices; #1
2 • select * from payments limit 1;
3 • select count(*) from orders;
4 • #select * from employees where employees.employeeNumber = employees.employeeNumber;
5 • #select * from payments where amount > 100000 order by amount desc;
6 • #select * from employees where employeeNumber = 1188 or employeeNumber = 1504;
7 • #select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100;
8 • #select avg(amount) from payments;
9 • #select count(city) from offices where addressLine1 is not null;
```

The 'Result Grid' shows the output of the third query: '# count(*)' followed by the value '326'. Below the grid is an 'Action Output' table with two rows:

#	Time	Action	Message	Duration / Fetch
279	22:29:12	select * from payments limit 1	1 row(s) returned	0.017 sec / 0.00005...
280	22:29:50	select count(*) From orders LIMIT 0, 1000	1 row(s) returned	0.037 sec / 0.00003...

4) Show all **employees** tuples where **reportsTo** is the same as **employeeNumber**.

select * from employees where employees.reportsTo = employees.employeeNumber; #4

show employees that have same employeeNumber as reportsto

The screenshot shows the MySQL Workbench interface with a database named 'companyx'. In the left sidebar, under 'SCHEMAS', the 'employees' table is selected, and its columns are listed: employeeNumber, lastName, firstName, extension, email, officeCode, reportsTo, and jobTitle. The main window contains a query editor with the following SQL code:

```
1 • select * from offices; #1
2 • select * from payments limit 1;
3 • select count(*) from orders; #3
4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • #select * from payments where amount > 100000 order by amount desc;
6 • #select * from employees where employeeNumber = 1188 or employeeNumber = 1504;
7 • #select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100;
8 • #select avg(amount) from payments;
9 • #select count(city) from offices where addressLine1 is not null;
```

The 'Result Grid' shows the output of the fourth query: '# employees' followed by the value '173'. Below the grid is an 'Action Output' table with two rows:

#	Time	Action	Message	Duration / Fetch
281	22:30:29	select * from employees where employees.employeeNu... 23 row(s) returned		0.00050 sec / 0.000...
282	22:31:33	select * from employees where employees.reportsTo = e... 0 row(s) returned		0.00046 sec / 0.000...

- 5) Show all information in the **payments** relation for payments exceeding 100 000, in decreasing order (i.e. from highest payment downwards).

```
select * from payments where amount > 100000 order by amount desc; #5 show
information where payments are more than R100000 from highest to lowest
```

```
select * from payments where amount > 100000 order by amount desc; #5
```

The screenshot shows the MySQL Workbench interface. The left sidebar contains various management and performance tabs like Server Status, Data Export, and Performance Reports. The main area has a 'Query 1' tab open with the following SQL code:

```

1 • select* from offices; #1
2 • select * from payments limit 1; #2
3 • select count(*) from orders; #3
4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • select * from payments where amount > 100000 order by amount desc; #5
6 • #select * from employees where employeeNumber = 1188 or employeeNumber = 1504;
7 • #select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100;
8 • #select avg(amount) from payments;
9 • #select count(city) from offices where addressLine1 is not null;

```

The 'Result Grid' shows the output of the last query, which retrieves 5 rows of payment data:

#	customerNumber	checkNumber	paymentDate	amount
1	141	JE105477	2005-03-18 00:00:00	120166.58
2	141	ID10962	2004-12-31 00:00:00	116208.4
3	124	KH131716	2003-08-15 00:00:00	111654.4
4	148	KM172879	2003-12-26 00:00:00	105743
5	124	AE215433	2005-03-05 00:00:00	101244.59

The 'Action Output' pane at the bottom shows two log entries:

#	Time	Action	Message	Duration / Fetch
282	22:31:33	select * from employees where employees.reportsTo = e...	0 row(s) returned	0.00046 sec / 0.000...
283	22:31:51	select * from payments where amount > 100000 order b...	5 row(s) returned	0.0076 sec / 0.000...

- 6) Show all information in the **employees** relation for **employeeNumbers** 1188 and 1504.

```
select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6
display all information for employees with employee number 118 and 1504
```

```
select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6
```

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

INSTANCE

SCHEMAS

Query 1

```

1 • select* from offices; #1
2 • select * from payments limit 1; #2
3 • select count(*) from orders; #3
4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • select * from payments where amount > 100000 order by amount desc; #5
6 • select * from employees where employeeNumber = 1188 or employeeNumber = 1504;
7 #select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100;
8 #select avg(amount) from payments;
9 #select count(city) from offices where addressLine1 is not null;

```

Result Grid

#	employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle
1	1188	Firrelli	Julie	x2173	jfirrelli@classicmodelcars.com	2	1143	Sales Rep
2	1504	Jones	Barry	x102	bjones@classicmodelcars.com	7	1102	Sales Rep

Action Output

#	Time	Action	Message	Duration / Fetch
283	22:31:51	select * from payments where amount > 100000 order b...	5 row(s) returned	0.0076 sec / 0.0000...
284	22:33:09	select * from employees where employeeNumber = 118...	2 row(s) returned	0.00062 sec / 0.0000...

- 7) Show the **productCode** of all **products** having their **quantityInStock** below 100, along with their total price. The total price is the **buyPrice** plus VAT (VAT is 14% of **buyPrice**).

select **productCode**, (**buyPrice***0.14 + **buyPrice**) from **products** where **quantityInStock** < 100; #7

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

INSTANCE

SCHEMAS

Query 1

```

1 • Execute the statement under the keyboard cursor
2 • select * from payments limit 1; #2
3 • select count(*) from orders; #3
4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • select * from payments where amount > 100000 order by amount desc; #5
6 • select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6
7 • select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100; #7
8 #select avg(amount) from payments;
9 #select count(city) from offices where addressLine1 is not null;

```

Result Grid

#	productCode	(buyPrice*0.14 + buyPrice)
1	S12_1099	108.6876
2	S24_2000	42.5448

Action Output

#	Time	Action	Message	Duration / Fetch
284	22:33:09	select * from employees where employeeNumber = 118...	2 row(s) returned	0.00062 sec / 0.0000...
285	22:33:55	select productCode, (buyPrice*0.14 + buyPrice) from pr...	2 row(s) returned	0.017 sec / 0.00005...

- 8) What is the average **payment amount** in the database?

select avg(amount) from payments; #8

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Shows the query: `select avg(amount) from payments; #8`. The result grid displays one row with the value 32431.645531135528.
- Action Output:** Shows two log entries:
 - Action: `select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100; #7`, Message: "2 row(s) returned", Duration: 0.017 sec / 0.0005...
 - Action: `select avg(amount) from payments LIMIT 0, 1000`, Message: "1 row(s) returned", Duration: 0.00078 sec / 0.000...

9) In how many cities are **offices** located (how many cities have **offices** in them) ?

select count(city) from offices where addressLine1 is not null;#9

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Shows the query: `select count(city) from offices where addressLine1 is not null; #9`. The result grid displays one row with the value 8.
- Action Output:** Shows two log entries:
 - Action: `select avg(amount) from payments LIMIT 0, 1000`, Message: "1 row(s) returned", Duration: 0.00078 sec / 0.000...
 - Action: `select count(city) from offices where addressLine1 is not null; #9`, Message: "1 row(s) returned", Duration: 0.00042 sec / 0.000...

10) Show all information in the **offices** relation where the **state** is missing/unknown.

select * from offices where state is null;#10

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query editor contains the following SQL code:

```
3 • select count(*) from orders; #3
4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • select * from payments where amount > 100000 order by amount desc; #5
6 • select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6
7 • select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100; #7
8 • select avg(amount) from payments; #8
9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • #select customerNumber, amount from payments where checkNumber like 'Q%';
```

The result grid displays the following data for three offices:

#	officeCode	city	phone	addressLine1	addressLine2	state	country	postalCode	territory
1	4	Paris	+33 14 723 4404	43 Rue Joffroy D'abbans	NULL	NULL	France	75017	EMEA
2	6	Sydney	+61 2 9264 2451	5-11 Wentworth Avenue	Floor #2	NULL	Australia	NSW 2010	APAC
3	7	London	+44 20 7877 2041	25 Old Broad Street	Level 7	NULL	UK	EC2N 1HN	EMEA

The 'offices 179' tab shows the execution history with two entries:

#	Time	Action	Message	Duration / Fetch
287	22:34:57	select count(city) from offices where addressLine1 is no...	1 row(s) returned	0.00042 sec / 0.000...
288	22:35:19	select * from offices where state is null LIMIT 0, 1000	3 row(s) returned	0.00039 sec / 0.000...

11) Show the **customerNumber** and **amount** for all **payments** with a 'Q' as the 2nd character of the **checkNumber** (a check is a cheque!)

select customerNumber, amount from payments where checkNumber like '_Q%'; #11

MySQL Workbench

```

4 • select * from employees where employees.reportsTo = employees.employeeNumber; #4
5 • select * from payments where amount > 100000 order by amount desc; #5
6 • select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6
7 • select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100; #7
8 • select avg(amount) from payments; #8
9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • #select jobtitle from employees;

```

Result Grid

#	customerNumber	amount
1	103	6066.78
2	112	32641.98
3	124	11044.3
4	166	38785.48
5	181	5494.78
6	187	47159.11
7	198	6036.96
...

Action Output

#	Time	Action	Message	Duration / Fetch
288	22:35:19	select * from offices where state is null LIMIT 0, 1000	3 row(s) returned	0.00039 sec / 0.000...
289	22:35:43	select customerNumber, amount from payments where... checkNumber like '_0%'	16 row(s) returned	0.013 sec / 0.00006...

12) What jobTitles exist in the database?

select distinct jobtitle from employees; #12

MySQL Workbench

```

9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • #select max(buyPrice) from products; not properly done
14 • /*select orders.orderNumber, status, quantityOrdered, productName
15 • from orders,orderdetails, products
16 • where productVendor = 'Exoto Designs' having status = 'cancelled';*/
17 • #select * from orderdetails,orders; can't figure it out

```

Result Grid

#	jobtitle
1	President
2	VP Sales
3	VP Marketing
4	Sales Manager (APAC)
5	Sale Manager (EMEA)
6	Sales Manager (NA)
7	Sales Rep

Action Output

#	Time	Action	Message	Duration / Fetch
291	22:36:37	select distinct from employees	Error Code: 1064. You have an error in your SQL ...	0.00025 sec
292	22:36:54	select distinct jobtitle from employees LIMIT 0, 1000	7 row(s) returned	0.00048 sec / 0.000...

13) Show productName and buyPrice of the product(s) with the highest buyPrice.

select productName, buyPrice from products order by buyPrice desc limit 1; #13

The screenshot shows the MySQL Workbench interface. The left sidebar contains various management, performance, and schema-related tools. The main area has a query editor titled "Query 1" containing the following SQL code:

```

9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 /*select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where productVendor = 'Exoto Designs' having status = 'cancelled';*/
17 #select * from orderdetails,orders; can't figure it out

```

The "Result Grid" shows one row of data:

#	productName	buyPrice
1	1962 LanciaA Delta 16V	103.42

The "Action Output" pane shows two log entries:

#	Time	Action	Message	Duration / Fetch
292	22:36:54	select distinct jobtitle from employees LIMIT 0, 1000	7 row(s) returned	0.00048 sec / 0.000...
293	22:38:43	select productName, buyPrice from products order by b...	1 row(s) returned	0.00059 sec / 0.000...

- 14) Show **orderNumber**, **status**, **quantityOrdered** and **productName** for all **products** from **productVendor** 'Exoto Designs' that have **status** 'Cancelled'.

select orders.orderNumber, status, quantityOrdered, productName
 from orders,orderdetails, products
 where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode =
 products.productCode and productVendor = 'Exoto Designs' having status = 'cancelled';#14

MySQL Workbench Screenshot:

```

9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   #select * from orderdetails,orders; can't figure it out

```

Result Grid:

#	orderNumber	status	quantityOrdered	productName
1	10167	Cancelled	34	1941 Chevrolet Special Deluxe Cab...
2	10179	Cancelled	45	1997 BMW F650 ST
3	10248	Cancelled	42	1904 Buick Runabout
4	10260	Cancelled	23	1992 Porsche Cayenne Turbo Silver
5	10262	Cancelled	34	1941 Chevrolet Special Deluxe Cab...

Action Output:

#	Time	Action	Message	Duration / Fetch
294	22:39:09	select orders.orderNumber, status, quantityOrdered, p...	1000 row(s) returned	0.019 sec / 0.00025 ...
295	22:44:28	select orders.orderNumber, status, quantityOrdered, p...	5 row(s) returned	0.142 sec / 0.00004...

15) Show the **productCode** of all **products** that have never been ordered.

select productCode from products where productCode not in(select productCode from orderdetails);

#15

MySQL Workbench Screenshot:

```

10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   #select productCode from products where productCode not in(select productCode from orderdetails);
18   #select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;

```

Result Grid:

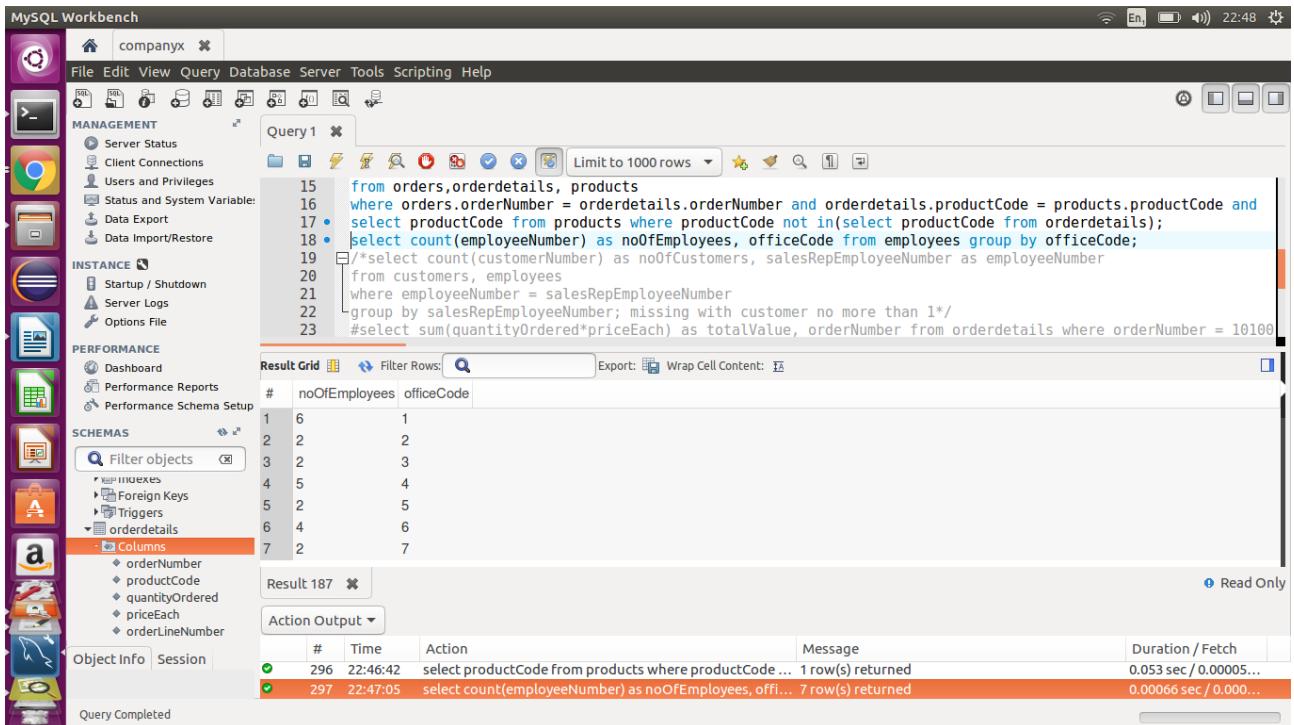
#	productCode
1	S18_3233
	NULL

Action Output:

#	Time	Action	Message	Duration / Fetch
295	22:44:28	select orders.orderNumber, status, quantityOrdered, p...	5 row(s) returned	0.142 sec / 0.00004...
296	22:46:42	select productCode from products where productCode ...	1 row(s) returned	0.053 sec / 0.00005...

16) Show how many **employees** there are in each office (give **officeCode** and value each time).

select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode; #16



The screenshot shows the MySQL Workbench interface. The left sidebar contains various management, instance, performance, and schema-related tools. The main area has a query editor titled "Query 1" containing the following SQL code:

```

15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   •    select productCode from products where productCode not in(select productCode from orderdetails);
18   •    select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;
19   /*select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber
20   from customers, employees
21   where employeeNumber = salesRepEmployeeNumber
22   group by salesRepEmployeeNumber; missing with customer no more than 1*/
23   #select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where orderNumber = 10100

```

The "Result Grid" shows the following data:

#	noOfEmployees	officeCode
1	6	1
2	2	2
3	2	3
4	5	4
5	2	5
6	4	6
7	2	7

The "Action Output" pane shows two log entries:

#	Time	Action	Message	Duration / Fetch
296	22:46:42	select productCode from products where productCode ...	1 row(s) returned	0.053 sec / 0.00005...
297	22:47:05	select count(employeeNumber) as noOfEmployees, offi...	7 row(s) returned	0.00066 sec / 0.000...

17) Show how many **customers** each employee is associated with (as **salesRepEmployeeNumber**), but only for employees who are the **salesRepEmployeeNumber** for at least 1 customer. Give **employeeNumber** and value each time.

select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber
from customers, employees
where employeeNumber = salesRepEmployeeNumber
group by salesRepEmployeeNumber;#17

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- Filter objects
- Indexes
- Foreign Keys
- Triggers
- orderdetails
- Columns

Object Info Session Query Completed

Query 1

```

10 • select * from offices where state is null; #10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;

```

Result Grid

#	noOfCustomers	employeeNumber
1	6	1165
2	6	1166
3	6	1188
4	6	1216
5	7	1286
6	8	1323
7	6	1337
...		

Result 189

Action Output

#	Time	Action	Message	Duration / Fetch
298	22:50:13	select count(employeeNumber) as noOfEmployees, offi... 7 row(s) returned	0.00055 sec / 0.000...	
299	22:50:21	select count(customerNumber) as noOfCustomers, sale... 15 row(s) returned	0.0012 sec / 0.000...	

Read Only

- 18) What was the total value of **orderNumber** 10100 i.e. the total of (**quantityOrdered** * **priceEach**) over all its orderlines?

select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where
orderNumber = 10100; #18

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- Filter objects
- Indexes
- Foreign Keys
- Triggers
- orderdetails
- Columns

Object Info Session Query Completed

Query 1

```

10 • select * from offices where state is null; #10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;

```

Result Grid

#	totalValue	orderNumber
1	10223.829999999998	10100

Result 190

Action Output

#	Time	Action	Message	Duration / Fetch
299	22:50:21	select count(customerNumber) as noOfCustomers, sale... 15 row(s) returned	0.0012 sec / 0.000...	
300	22:51:18	select sum(quantityOrdered*priceEach) as totalValue, ... 1 row(s) returned	0.00059 sec / 0.000...	

Read Only

19) Show the **productName** of the product/s with the largest **quantityInStock**.

select productName from products order by quantityInStock desc limit 1; #19

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons.
- Left Sidebar (MANAGEMENT):** Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore.
- Left Sidebar (INSTANCE):** Startup / Shutdown, Server Logs, Options File.
- Left Sidebar (PERFORMANCE):** Dashboard, Performance Reports, Performance Schema Setup.
- Left Sidebar (SCHEMAS):** Filter objects, indexes, Foreign Keys, Triggers, orderdetails (selected), Columns (selected). Sub-items under orderdetails include orderNumber, productCode, quantityOrdered, priceEach, and orderLineNumber.
- Central Area (Query Editor):** Shows the following SQL code:

```
10 • select * from offices where state is null; #10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobTitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16  where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17  select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;
```
- Result Grid:** Shows the result of the query:

#	productName
1	2002 Suzuki XREO
- Action Output:** Shows the execution log:

#	Time	Action	Message	Duration / Fetch
301	22:51:33	select max(quantityInStock) from products LIMIT 0, 1000	1 row(s) returned	0.0065 sec / 0.0000...
302	22:52:26	select productName from products order by quantityInStock desc limit 1;	1 row(s) returned	0.00059 sec / 0.0000...

20) Show the **employeeNumber** of **employees** who **reportsTo** the same person as does **employeeNumber** 1313 (i.e. who have the same boss as 1313).

select employeeNumber from employees where employeeNumber = 1313 and employeeNumber = reportsTo; #20

The screenshot shows the MySQL Workbench interface. The left sidebar contains various management, instance, performance, and schema-related tools. The main area has a query editor titled 'Query 1' with the following SQL code:

```

10 • select * from offices where state is null; #10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;

```

The results grid shows one row of data:

#	employeeNumber
#	employeeNumber

The bottom right corner of the results grid indicates 'Read Only'.

- 21) Show the **employeeNumber** of all **employees** who are superiors of **employeeNumber** 1313
 (i.e. the person 1313 **reportsTo**, and the employee who that person **reportsTo**, ... all the way up)
- 22) Devise a useful query of your own involving the most interesting usage of SQL you can think of. Explain clearly in a comment what it is meant to find from the database. Also explain how you know the SQL for this query is correct (showing intermediate results if necessary). Marks here will be proportional to the complexity and usefulness of the query you implement.

select * from products where (buyPrice between 30 and 40) and not productLine in ('Classic Cars','Vintage Cars'); #22

The following SQL statement selects all products with a price BETWEEN 30 and 40. In addition; do not show products with a productLine of 'Classic Cars','Vintage Cars'

The screenshot shows the MySQL Workbench interface. The left sidebar contains management, instance, and schema tabs. The main area has a query editor with a list of SQL statements and a results grid.

```

19 • select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber #17
from customers, employees
where employeeNumber = salesRepEmployeeNumber
group by salesRepEmployeeNumber;
20
21
22
23 • select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where orderNumber = 10100;
24 • select productName from products order by quantityInStock desc limit 1; #19
25 select employeeNumber from employees where employeeNumber = 1313 and employeeNumber = reportsTo; #20
26 • select * from products;
27 • select * from products where (buyPrice between 30 and 45) and not productLine in ('Classic Cars','Vintage Car
28 • INSERT INTO `offices`(`officeCode`, `city`, `phone`, `addressLine1`, `addressLine2`, `state`, `country`, `po
29 ('999','Cape Town', "", "", "Western Province", "", "");#23
30 • select * from offices where officeCode = 999; #23
31 • update employees set employeeNumber = max(employees.employeeNumber)+1;
32 • DELETE from orders where orderNumber = '10101'; #25
33

```

Result Grid:

#	productCode	productName	productLine	productScale	productVendor	productDescription	quantityInStock
1	S18_3782	1957 Vespa GS150	Motorcycles	1:18	Studio M Art Models	Features rotating wheels; working k...	7689
2	S24_2000	1960 BSA Gold Star DBD34	Motorcycles	1:24	Highway 66 Mini Classics	Detailed scale replica with working ...	15
3	S24_2R41	1900s Vintage Racer Plane	Planes	1:24	Autumn Studio Design	Hand crafted diecast-like metal hi-nl	5942

Action Output:

#	Time	Action	Message	Duration / Fetch
333	23:47:12	select * from products LIMIT 0, 1000	110 row(s) returned	0.010 sec / 0.00017 ...
334	23:48:14	select * from products where (buyPrice between 30 and ... 13 row(s) returned		0.00098 sec / 0.000...

23) Add a new **office** to the database, giving it **officeCode** 999 (meaning planned for later). This office will be in Cape Town, but no other details are known yet. Make **state** ‘Western Province’.

```

INSERT INTO `offices`(`officeCode`, `city`, `phone`, `addressLine1`, `addressLine2`, `state`,
`country`, `postalCode`, `territory`) VALUES
('999','Cape Town', "", "", "Western Province", "", "");#23
select * from offices where officeCode = 999; #23 populating the table offices with one record

```

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- Filter objects
- Indexes
- Foreign Keys
- Triggers
- orderdetails
- Columns

Object Info Session

Query Completed

Query 1

```

10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;
19 • select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber
20   from customers, employees
21   where employeeNumber = salesRepEmployeeNumber
22   group by salesRepEmployeeNumber;
23 • select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where orderNumber = 10100;
24 • select productName from products order by quantityInStock desc limit 1.

```

Result Grid

#	officeCode	city	phone	addressLine1	addressLine2	state	country	postalCode	territory
1	999	Cape Town	HULL	HULL	HULL	Western Province	HULL	HULL	HULL

offices 200

Action Output

#	Time	Action	Message	Duration / Fetch
314	23:06:20	select max(employees.employeeNumber) as Highest fr...	1 row(s) returned	0.00053 sec / 0.000...
315	23:18:36	select * from offices where officeCode = 999 LIMIT 0, 1000	1 row(s) returned	0.00088 sec / 0.000...

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- Filter objects
- Indexes
- Foreign Keys
- Triggers
- orderdetails
- Columns

Object Info Session

Query Completed

Query 1

```

9 • select count(city) from offices where addressLine1 is not null;#9
10 • select * from offices where state is null;#10
11 • select customerNumber, amount from payments where checkNumber like '_0%'; #11
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName
15   from orders,orderdetails, products
16   where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
17   select productCode from products where productCode not in(select productCode from orderdetails);
18 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode;
19 • select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber
20   from customers, employees
21   where employeeNumber = salesRepEmployeeNumber
22   group by salesRepEmployeeNumber;
23 • select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where orderNumber = 10100;
24 • select productName from products order by quantityInStock desc limit 1;
25 • select employeeNumber from employees where employeeNumber = 1313 and employeeNumber = reportsTo;
26 • INSERT INTO `offices`(`officeCode`, `city`, `phone`, `addressLine1`, `addressLine2`, `state`, `country`, `po
('999', 'Cape Town', '', '', '', 'Western Province', '', '', ''));#
27 #update employees set employeeNumber = '1' where employeeNumber = 1313;
28 • DELETE from offices where officeCode = '999';
29 •
30 •
31 •

```

Action Output

#	Time	Action	Message	Duration / Fetch
307	22:56:17	DELETE from offices where officeCode='999'	1 row(s) affected	0.166 sec
308	22:56:26	INSERT INTO `offices`(`officeCode`, `city`, `phone`, `a...	1 row(s) affected	0.088 sec

24) Employee 1313 is superstitious. Change their employee number in the database, giving them the employee number 1 greater than the largest employee number in the database.

```
update employees set employeeNumber = max(employees.employeeNumber) where
employeeNumber = 1313; #24
```

25) OrderNumber 10101 was never signed by the customer. Remove it from the database.

DELETE from orders where orderNumber = '10101'; #25 remove record with order number 10101

The screenshot shows the MySQL Workbench interface. In the center is a query editor titled "Query 1" containing the following SQL code:

```
12 • select distinct jobtitle from employees; #12
13 • select productName, buyPrice from products order by buyPrice desc limit 1; #13
14 • select orders.orderNumber, status, quantityOrdered, productName #14
from orders,orderdetails, products
15 where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode = products.productCode and
16 select productCode from products where productCode not in(select productCode from orderdetails); #15
17 • select count(employeeNumber) as noOfEmployees, officeCode from employees group by officeCode; #16
18 • select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber #17
from customers, employees
19 where employeeNumber = salesRepEmployeeNumber
group by salesRepEmployeeNumber;
20 • select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where orderNumber = 10100;
21 select productName from products order by quantityInStock desc limit 1; #19
22 select employeeNumber from employees where employeeNumber = 1313 and employeeNumber = reportsTo; #20
23 • SELECT * FROM Products WHERE (Price BETWEEN 10 AND 20) AND NOT CategoryID IN (1,2,3);
24 • INSERT INTO `offices`(`officeCode`, `city`, `phone`, `addressLine1`, `addressLine2`, `state`, `country`, `po
('999','Cape Town', '+27 12 456 7890', 'Bergendorp 74', 'Mooiendal', 'Western Province', '+27 12 456 7890');
25 • select *
26 • select * from offices where officeCode = 999; #23
27 • update employees set employeeNumber = max(employees.employeeNumber)+1;
28 • DELETE from orders where orderNumber = '10101';
29
30
31
32
33
34
```

Below the query editor is an "Action Output" table with the following data:

#	Time	Action	Message	Duration / Fetch
316	23:31:03	DELETE from orders where orderNumber = '10100'	1 row(s) affected	0.157 sec
317	23:31:31	DELETE from orders where orderNumber = '10101'	1 row(s) affected	0.077 sec

At the bottom left, it says "Query Completed".

Commands for each question

select* from offices; #1

select * from payments limit 1; #2

select count(*) from orders; #3

select * from employees where employees.reportsTo = employees.employeeNumber; #4

select * from payments where amount > 100000 order by amount desc; #5

select * from employees where employeeNumber = 1188 or employeeNumber = 1504; #6

select productCode, (buyPrice*0.14 + buyPrice) from products where quantityInStock < 100; #7

select avg(amount) from payments; #8

select count(city) from offices where addressLine1 is not null;#9

select * from offices where state is null;#10

select customerNumber, amount from payments where checkNumber like '_Q%'; #11

select distinct jobtitle from employees; #12

select productName, buyPrice from products order by buyPrice desc limit 1; #13

select orders.orderNumber, status, quantityOrdered, productName

from orders,orderdetails, products

```
where orders.orderNumber = orderdetails.orderNumber and orderdetails.productCode =
products.productCode and productVendor = 'Exoto Designs' having status = 'cancelled';#14
select productCode from products where productCode not in(select productCode from orderdetails);
#15
select count(employeeNumber) as noOfEmployees, officeCode from employees group by
officeCode; #16
select count(customerNumber) as noOfCustomers, salesRepEmployeeNumber as employeeNumber
from customers, employees
where employeeNumber = salesRepEmployeeNumber
group by salesRepEmployeeNumber;#17
select sum(quantityOrdered*priceEach) as totalValue, orderNumber from orderdetails where
orderNumber = 10100; #18
select productName from products order by quantityInStock desc limit 1; #19
select employeeNumber from employees where employeeNumber = 1313 and employeeNumber =
reportsTo; #20

select * from products where (buyPrice between 30 and 40) and not productLine in ('Classic
Cars','Vintage Cars'); #22
```

The following SQL statement selects all products with a price BETWEEN 30 and 40.

In addition; do not show products with a productLine of 'Classic Cars','Vintage Cars'

```
INSERT INTO `offices`(`officeCode`, `city`, `phone`, `addressLine1`, `addressLine2`, `state`,
`country`, `postalCode`, `territory`) VALUES
('999','Cape Town', " ", " ", "Western Province", " ", " ");#23
select * from offices where officeCode = 999; #23
update employees set employeeNumber = max(employees.employeeNumber) where
employeeNumber = 1313; #24
DELETE from orders where orderNumber = '10101'; #25
```