

Consider the Country table and Persons table that you created earlier and perform the following:

1. Find the number of persons in each country.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with a list of tables including 'country' and 'persons'. The main query editor contains the following SQL code:

```
1 use d41;
2 select country_name, count(country_name) as No_of_persons from country group by country_name;
```

The 'Result Grid' at the bottom displays the results of the query:

country_name	No_of_persons
USA	1
Canada	1
UK	1
India	1
China	1

The 'Action Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	11:07:47	select country_name, count(country_name) as No_of_persons from country group by co...	11 row(s) returned	0.000 sec / 0.000 sec

2. Find the number of persons in each country sorted from high to low.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with a list of tables including 'country' and 'persons'. The main query editor contains the following SQL code:

```
1 use d41;
2 select country_name, count(country_name) as No_of_persons from country group by country_name;
3 select * from country;
4
5 select country_name, count(country_name) as No_of_persons from country group by country_name order by No_of_persons desc;
```

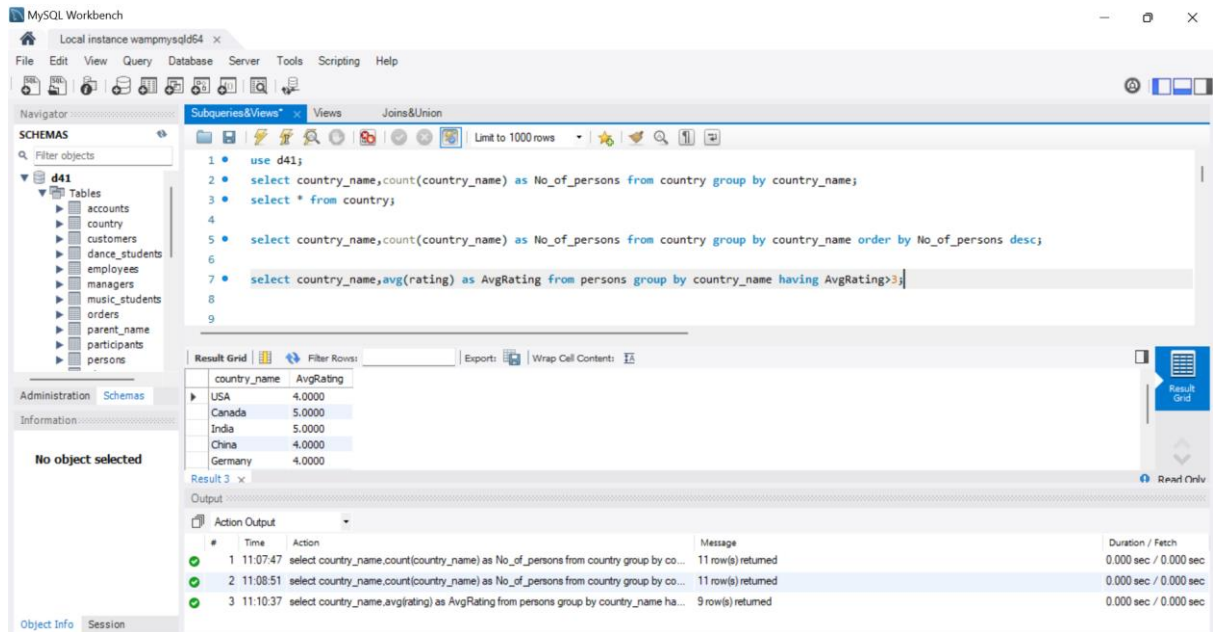
The 'Result Grid' at the bottom displays the results of the query:

country_name	No_of_persons
USA	1
Canada	1
UK	1
India	1
China	1

The 'Action Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	11:07:47	select country_name, count(country_name) as No_of_persons from country group by co...	11 row(s) returned	0.000 sec / 0.000 sec
2	11:08:51	select country_name, count(country_name) as No_of_persons from country group by co...	11 row(s) returned	0.000 sec / 0.000 sec

3. Find out an average rating for Persons in respective countries if the average is greater than 3.0



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

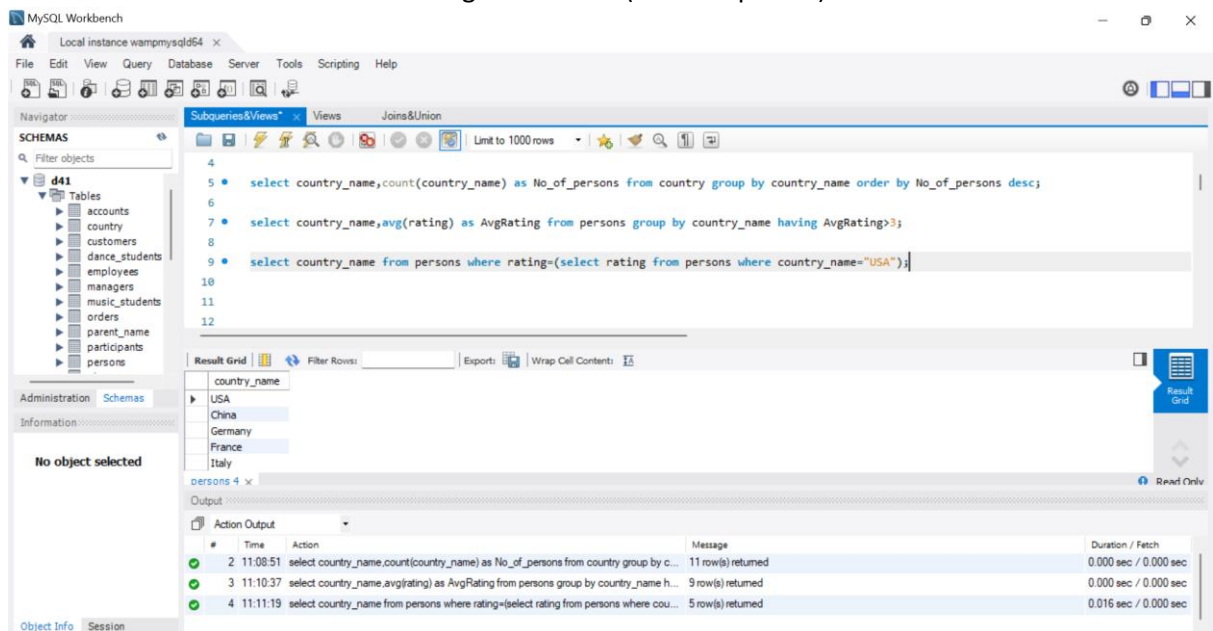
```
1 use d41;
2 select country_name,count(country_name) as No_of_persons from country group by country_name;
3 select * from country;
4
5 select country_name,count(country_name) as No_of_persons from country group by country_name order by No_of_persons desc;
6
7 select country_name,avg(rating) as AvgRating from persons group by country_name having AvgRating>3;
8
9
```

The results grid displays the output of the third query, showing the average rating for each country:

country_name	AvgRating
USA	4.0000
Canada	5.0000
India	5.0000
China	4.0000
Germany	4.0000

The output pane shows the execution of three queries, with the third query returning 9 rows.

4. Find the countries with the same rating as the USA. (Use Subqueries).



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

```
4
5 select country_name,count(country_name) as No_of_persons from country group by country_name order by No_of_persons desc;
6
7 select country_name,avg(rating) as AvgRating from persons group by country_name having AvgRating>3;
8
9 select country_name from persons where rating=(select rating from persons where country_name="USA");
10
11
12
```

The results grid displays the output of the third query, showing the average rating for each country:

country_name
USA
China
Germany
France
Italy

The output pane shows the execution of four queries, with the fourth query returning 5 rows.

5. Select all countries whose population is greater than the average population of all nations.

The screenshot shows the MySQL Workbench interface with three queries executed in the 'Subqueries/Views' tab. The 'Navigator' on the left shows the 'd41' database with tables like 'accounts', 'country', 'customers', etc. The 'Result Grid' shows the results of the queries.

Query 7: `select country_name, avg(rating) as AvgRating from persons group by country_name having AvgRating > 3;`

Query 9: `select country_name from persons where rating = (select rating from persons where country_name = "USA");`

Query 11: `select country_name from persons where population > (select avg(population) from persons);`

Result Grid:

country_name
USA
India
China

Output:

#	Time	Action	Message	Duration / Fetch
3	11:10:37	select country_name, avg(rating) as AvgRating from persons group by country_name h...	9 row(s) returned	0.000 sec / 0.000 sec
4	11:11:19	select country_name from persons where rating = (select rating from persons where cou...	5 row(s) returned	0.016 sec / 0.000 sec
5	11:13:34	select country_name from persons where population > (select avg(population) from pers...	3 row(s) returned	0.000 sec / 0.000 sec

1. Create a view named `customer_info` for the `Customer` table that displays Customer's Full name and email address. Then perform the `SELECT` operation for the `customer_info` view.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
43 (14, 'Sofia', 'Lopez', 'sofia.lopez@example.com', '4564564567', '282 Banyan St', 'Buenos Aires', 'Buenos Aires', 'C1000', 'Argentina'),
44 (15, 'Nina', 'Petrov', 'nina.petrov@example.com', '5675675678', '383 Willow Ln', 'Sofia', 'Sofia City Province', '1000', 'Bulgaria');
45
46 * select * from customer;
47
48 * create view customer_info as
49   select concat(fname, " ", lname) as fullname, email from customer;
50 * select * from customer_info;
51
```

The **Result Grid** shows the output of the `SELECT * FROM customer;` query:

fullname	email
John Doe	john.doe@example.com
Jane Smith	jane.smith@example.com
Carlos Mendez	carlos.mendez@example.com
Amina Khan	amina.khan@example.com
Li Wei	li.wei@example.com

The **Action Output** table shows the execution of the view creation and selection queries:

#	Time	Action	Message	Duration / Fetch
5	11:13:34	select country_name from persons where population > (select avg(population) from persons);	3 row(s) returned	0.000 sec / 0.000 sec
6	11:14:33	use product	0 row(s) affected	0.000 sec
7	11:14:55	select * from customer_info LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec

2. Create a view named `US_Customers` that displays customers located in the US.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
47
48 * create view customer_info as
49   select concat(fname, " ", lname) as fullname, email from customer;
50 * select * from customer_info;
51
52 * create view us_customers as
53   select concat(fname, " ", lname) as fullname from customer where country="USA";
54 * select * from us_customers;
55
```

The **Result Grid** shows the output of the `SELECT * FROM customer_info;` query:

fullname
John Doe
Jane Smith
Carlos Mendez

The **Action Output** table shows the execution of the view creation and selection queries:

#	Time	Action	Message	Duration / Fetch
6	11:14:33	use product	0 row(s) affected	0.000 sec
7	11:14:55	select * from customer_info LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec
8	11:15:35	select * from us_customers LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

3. Create another view named Customer_details with columns full name(Combine first_name and last_name), email, phone_no, and state.

The screenshot shows the MySQL Workbench interface. The 'Subqueries/Views' tab is active, displaying the following SQL code:

```
51
52 * create view us_customers as
53 select concat(fname, " ", lname) as fullname from customer where country="USA";
54 * select * from us_customers;
55
56 * create view customer_details as
57 select concat(fname, " ", lname) as fullname, email, phno, state from customer;
58 * select * from customer_details;
59
```

The 'Result Grid' shows the output of the queries:

fullname	email	phno	state
John Doe	john.doe@example.com	1234567890	Illinois
Jane Smith	jane.smith@example.com	2468101214	California
Carlos Mendez	carlos.mendez@example.com	3456789012	Texas
Amina Khan	amina.khan@example.com	4567890123	Maharashtra
Li Wei	li.wei@example.com	5678901234	Beijing

The 'Output' tab shows the execution results:

#	Time	Action	Message	Duration / Fetch
7	11:14:55	select * from customer_info LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec
8	11:15:35	select * from us_customers LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
9	11:16:37	select * from customer_details LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec

4. Update phone numbers of customers who live in California for Customer_details view.

The screenshot shows the MySQL Workbench interface. The 'Subqueries/Views' tab is active, displaying the following SQL code:

```
52 * create view us_customers as
53 select concat(fname, " ", lname) as fullname from customer where country="USA";
54 * select * from us_customers;
55
56 * create view customer_details as
57 select concat(fname, " ", lname) as fullname, email, phno, state from customer;
58 * select * from customer_details;
59
60 * set sql_safe_updates=@;
61 * update customer_details set phno="2468101214" where state="California";
62
63
64
65
66
67
```

The 'Output' tab shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	11:07:47	select country_name, count(country_name) as No_of_persons from country group by c...	11 row(s) returned	0.000 sec / 0.000 sec
2	11:08:51	select country_name, count(country_name) as No_of_persons from country group by c...	11 row(s) returned	0.000 sec / 0.000 sec
3	11:10:37	select country_name, avg(rating) as AvgRating from persons group by country_name h...	9 row(s) returned	0.000 sec / 0.000 sec
4	11:11:19	select country_name from persons where rating=(select rating from persons where cou...	5 row(s) returned	0.016 sec / 0.000 sec

5. Count the number of customers in each state and show only states with more than 5 customers.

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
60 • set sql_safe_updates=0;
61 • update customer_details set phno="2468101214" where state="California";
62
63 • select state,count(state) as no_of_customers from customer group by state having no_of_customers>5;
64
65
66
67
68
```

The Result Grid is empty, showing columns 'state' and 'no_of_customers'. The Action Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
9	11:16:37	select "from customer_details LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec
10	11:17:10	update customer_details set phno="2468101214" where state="California"	Error Code: 1175. You are using safe update mode and you tried to update a table wit...	0.000 sec
11	11:18:17	select state,count(state) as no_of_customers from customer group by state having no_...	0 row(s) returned	0.000 sec / 0.000 sec

6. Write a query that will return the number of customers in each state, based on the "state" column in the "customer_details" view.

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
60 • set sql_safe_updates=0;
61 • update customer_details set phno="2468101214" where state="California";
62
63 • select state,count(state) as no_of_customers from customer group by state having no_of_customers>5;
64
65 • select state,count(state) as no_of_customers from customer_details group by state;
66
67
68
```

The Result Grid shows the following data:

state	no_of_customers
Illinois	1
California	1
Texas	1
Maharashtra	1
Beijing	1

The Action Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
10	11:17:10	update customer_details set phno="2468101214" where state="California"	Error Code: 1175. You are using safe update mode and you tried to update a table wit...	0.000 sec
11	11:18:17	select state,count(state) as no_of_customers from customer group by state having no_...	0 row(s) returned	0.000 sec / 0.000 sec
12	11:18:51	select state,count(state) as no_of_customers from customer_details group by state LIM...	15 row(s) returned	0.000 sec / 0.000 sec

7. Write a query that returns all the columns from the "customer_details" view, sorted by the "state" column in ascending order.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
60 * set sql_safe_updates=0;
61 * update customer_details set phno="2468101214" where state="California";
62
63 * select state,count(state) as no_of_customers from customer group by state having no_of_customers>5;
64
65 * select state,count(state) as no_of_customers from customer_details group by state;
66
67 * select * from customer_details order by state asc;
68
```

The Results Grid shows the output of the last query, displaying columns: fullname, email, phno, and state. The data is as follows:

fullname	email	phno	state
Li Wei	li.wei@example.com	5678901234	Beijing
Hans Schmidt	hans.schmidt@example.com	7890123456	Berlin
Sofia Lopez	sofia.lopez@example.com	4564564567	Buenos Aires
Ahmed Ali	ahmed.ali@example.com	2342342345	Cairo
Jane Smith	jane.smith@example.com	2468101214	California

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
11	11:18:17	select state,count(state) as no_of_customers from customer group by state having no...	0 row(s) returned	0.000 sec / 0.000 sec
12	11:18:51	select state,count(state) as no_of_customers from customer_details group by state LIM...	15 row(s) returned	0.000 sec / 0.000 sec
13	11:19:31	select * from customer_details order by state asc LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec