


SUBQUERIES AND VIEWS

PART 1


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


DESC country;

Result Grid


Filter Rows:

Export:




Wrap Cell Content:



	Field	Type	Null	Key	Default	Extra
▶	Id	int	NO	PRI	NULL	
	Country_name	varchar(50)	YES		NULL	
	Population	bigint	YES		NULL	
	Area	int	YES		NULL	

DESC persons;


Result Grid


Filter Rows:

Export:



Wrap Cell Content:



	Field	Type	Null	Key	Default	Extra
▶	Id	int	NO	PRI	NULL	
	Fname	varchar(50)	YES		NULL	
	Lname	varchar(50)	YES		NULL	
	Population	bigint	YES		NULL	
	Rating	decimal(3,2)	YES		NULL	
	Country_Id	int	YES	MUL	NULL	
	Country_name	varchar(50)	YES		NULL	

select * from country;

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	Id	Country_name	Population	Area
	1	USA	75000000	9833520
	2	India	5000000	3287263
	3	Canada	8000000	9984670
	4	UK	150000000	243610
	5	Australia	400000	7692024
	6	Germany	2000000	357022
	7	France	6900000	551695
	8	Brazil	6000000	8515767
	9	Russia	15500000	17098242
	10	Japan	8650000	377930
	NULL	NULL	NULL	NULL

select * from persons;

Result Grid		Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:	
	Id	Fname	Lname	Population	Rating	Country_Id	Country_name	DOB	
▶	1	Jeeva	Joseph	75000000	4.50	1	USA	1990-01-01	
	2	Arya	Shan	5000000	3.80	2	India	1985-06-15	
	3	Emma	Emmi	8000000	4.90	3	Canada	1983-08-18	
	4	Basim	Johny	150000000	4.20	4	UK	1988-06-05	
	5	Aryan	John	400000	3.90	5	Australia	1980-04-16	
	6	Maria	Grace	2000000	4.70	6	Germany	1988-01-19	
	7	James	Wilson	6900000	2.10	7	France	1999-07-29	
	8	Suzzy	Martin	6000000	4.00	8	Brazil	1985-08-25	
	9	Robert	Williams	15500000	1.70	9	Russia	1982-11-16	
	10	Jane	Taylor	8650000	4.60	10	Japan	1983-02-27	
	11	Aishw...	NULL	75000000	4.50	1	USA	1988-12-25	
	12	Daisy	NULL	15500000	1.70	9	Russia	1990-01-13	
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	


1. Find the number of persons in each country.


SELECT Country_name, COUNT(*) AS Number_of_Persons


FROM Persons

GROUP BY Country_name;

Result Grid

 Filter Rows:

Export: 

Wrap Cell Content: 

	Country_name	Number_of_Persons
▶	USA	2
	India	1
	Canada	1
	UK	1
	Australia	1
	Germany	1
	France	1
	Brazil	1
	Russia	2
	Japan	1

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

```
SELECT Country_name, COUNT(*) AS Number_of_Persons  
FROM Persons  
GROUP BY Country_name  
ORDER BY Number_of_Persons DESC;
```

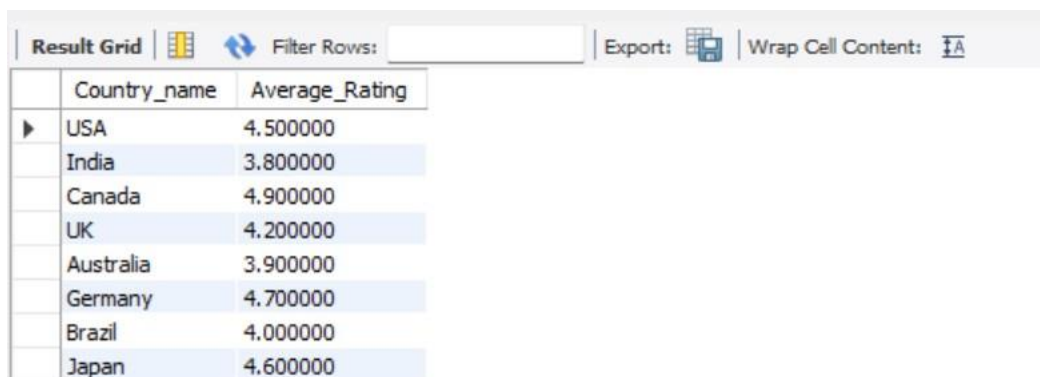


The screenshot shows a database query result grid. At the top, there is a toolbar with 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content' options. The main table has two columns: 'Country_name' and 'Number_of_Persons'. The data is sorted in descending order by the number of persons. The USA and Russia both have 2 persons, while all other countries (India, Canada, UK, Australia, Germany, France, Brazil, and Japan) have 1 person each.

Country_name	Number_of_Persons
USA	2
Russia	2
India	1
Canada	1
UK	1
Australia	1
Germany	1
France	1
Brazil	1
Japan	1

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

```
SELECT Country_name, AVG(Rating) AS Average_Rating  
FROM Persons  
GROUP BY Country_name  
HAVING AVG(Rating) > 3.0;
```



The screenshot shows a database query result grid. At the top, there is a toolbar with 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content' options. The main table has two columns: 'Country_name' and 'Average_Rating'. The data is sorted in descending order by the average rating. The USA has the highest average rating at 4.500000, followed by Canada at 4.900000, Germany at 4.700000, Japan at 4.600000, UK at 4.200000, Brazil at 4.000000, Australia at 3.900000, and India at 3.800000.

Country_name	Average_Rating
USA	4.500000
India	3.800000
Canada	4.900000
UK	4.200000
Australia	3.900000
Germany	4.700000
Brazil	4.000000
Japan	4.600000

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

```
SELECT DISTINCT Country_name  
FROM Persons  
WHERE Rating IN (SELECT Rating FROM Persons WHERE Country_name = 'USA');
```

Result Grid			Filter Rows: <input type="text"/>	Export:	Wrap Cell Content:
	Country_name				
▶	USA				

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

```
SELECT Country_name, Population  
FROM Country  
WHERE Population > (SELECT AVG(Population) FROM Country);
```

Result Grid			Filter Rows: <input type="text"/>	Export:	Wrap Cell Content:
	Country_name	Population			
▶	USA	75000000			
	UK	150000000			

PART 2

-- Create a database named Product

```
CREATE DATABASE Product;
```

```
USE Product;
```

-- create a table called Customer

**-- fields in the Product database: Customer_Id - Make PRIMARY KEY First_name Last_name
Email Phone_no Address City State Zip_code Country**

```
CREATE TABLE Customer (
```

```
    Customer_Id INT PRIMARY KEY,
```

```
    First_name VARCHAR(50),
```

```
    Last_name VARCHAR(50),
```

```
    Email VARCHAR(100),
```

```
    Phone_no VARCHAR(15),
```

```
    Address VARCHAR(100),
```

```
    City VARCHAR(50),
```

```
    State VARCHAR(50),
```

```
    Zip_code VARCHAR(10),
```

```
    Country VARCHAR(50)
```

```
);
```

```
DESC customer;
```

Result Grid						
		Filter Rows:				
		Export:				
		Wrap Cell Content:				
	Field	Type	Null	Key	Default	Extra
▶	Customer_Id	int	NO	PRI	NULL	
	First_name	varchar(50)	YES		NULL	
	Last_name	varchar(50)	YES		NULL	
	Email	varchar(100)	YES		NULL	
	Phone_no	varchar(15)	YES		NULL	
	Address	varchar(100)	YES		NULL	
	City	varchar(50)	YES		NULL	
	State	varchar(50)	YES		NULL	
	Zip_code	varchar(10)	YES		NULL	
	Country	varchar(50)	YES		NULL	

-- INSERT VALUES IN TO CUSTOMER TABLE

INSERT INTO Customer (Customer_Id, First_name, Last_name, Email, Phone_no, Address, City, State, Zip_code, Country)

VALUES

(1, 'John', 'Jacob', 'johnjacob@example.com', '1234567890', '123 Elm St', 'New York', 'New York', '10001', 'US'),

(2, 'Annie', 'Smith', 'Anniesmith@example.com', '0987654321', '456 Oak St', 'Los Angeles', 'California', '90001', 'US'),

(3, 'Michael', 'Johnson', 'mjohnson@example.com', '2345678901', '789 Pine St', 'Chicago', 'Illinois', '60601', 'US'),

(4, 'Emily', 'Cooper', 'Emilycooper@example.com', '3456789012', '101 Maple Ave', 'Melbourne', 'Victoria', '77001', 'Australia'),

(5, 'William', 'Joe', 'williamj@example.com', '4567890123', '202 Birch Rd', 'Mumbai', 'Maharashtra', '75201', 'India'),

(6, 'Linda', 'Wilson', 'lwilson@example.com', '5678901234', '303 street', 'Vancouver', 'British Columbia', '19101', 'Canada'),

(7, 'Jaime', 'Taylor', 'jtaylor@example.com', '6789012345', '404 Walnut Ln', 'San Diego', 'California', '92101', 'US'),

(8, 'Basim', 'jass', 'bjass@example.com', '7890123456', '505 Ash Dr', 'Dallas', 'Texas', '75201', 'US'),

(9, 'Robert', 'Andy', 'robert@example.com', '8901234567', '606 Spruce Ct', 'San Francisco', 'California', '94101', 'US'),

(10, 'Patty', 'Thomas', 'pthomas@example.com', '9012345678', '707 Redwood St', 'Bangalore', 'Karnataka', '73301', 'India');

SELECT * FROM customer;

Result Grid										
Filter Rows:										
Edit: Export/Import: Wrap Cell Content:										
	Customer_Id	First_name	Last_name	Email	Phone_no	Address	City	State	Zip_code	Country
▶	1	John	Jacob	johnjacob@example.com	1234567890	123 Elm St	New York	New York	10001	US
	2	Annie	Smith	Anniesmith@example.com	0987654321	456 Oak St	Los Angeles	California	90001	US
	3	Michael	Johnson	mjohnson@example.com	2345678901	789 Pine St	Chicago	Illinois	60601	US
	4	Emily	Cooper	Emilycooper@example.com	3456789012	101 Maple Ave	Melbourne	Victoria	77001	Australia
	5	William	Joe	williamj@example.com	4567890123	202 Birch Rd	Mumbai	Maharashtra	75201	India
	6	Linda	Wilson	lwilson@example.com	5678901234	303 street	Vancouver	British Columbia	19101	Canada
	7	Jaime	Taylor	jtaylor@example.com	6789012345	404 Walnut Ln	San Diego	California	92101	US
	8	Basim	jass	bjass@example.com	7890123456	505 Ash Dr	Dallas	Texas	75201	US
	9	Robert	Andy	robert@example.com	8901234567	606 Spruce Ct	San Francisco	California	94101	US
	10	Patty	Thomas	pthomas@example.com	9012345678	707 Redwood St	Bangalore	Karnataka	73301	India
✱	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

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

CREATE VIEW customer_info AS

SELECT CONCAT(First_name, ' ', Last_name) AS Full_name, Email

FROM Customer;

-- perform the SELECT operation for the customer_info view

SELECT * FROM customer_info;

Result Grid		
Filter Rows:		
Export: Wrap Cell Content:		
	Full_name	Email
▶	John Jacob	johnjacob@example.com
	Annie Smith	Anniesmith@example.com
	Michael Johnson	mjohnson@example.com
	Emily Cooper	Emilycooper@example.com
	William Joe	williamj@example.com
	Linda Wilson	lwilson@example.com
	Jaime Taylor	jtaylor@example.com
	Basim jass	bjass@example.com
	Robert Andy	robert@example.com
	Patty Thomas	pthomas@example.com

customer_info 7 x

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

```
CREATE VIEW US_Customers AS
```

```
SELECT *
```

```
FROM Customer
```

```
WHERE Country = 'US';
```

-- perform the SELECT operation for the US_Customers view

```
SELECT * FROM US_Customers;
```

Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content:										
	Customer_Id	First_name	Last_name	Email	Phone_no	Address	City	State	Zip_code	Country
▶	1	John	Jacob	johnjacob@example.com	1234567890	123 Elm St	New York	New York	10001	US
	2	Annie	Smith	Anniesmith@example.com	0987654321	456 Oak St	Los Angeles	California	90001	US
	3	Michael	Johnson	mjohnson@example.com	2345678901	789 Pine St	Chicago	Illinois	60601	US
	7	Jaime	Taylor	jtaylor@example.com	6789012345	404 Walnut Ln	San Diego	California	92101	US
	8	Basim	jass	bjass@example.com	7890123456	505 Ash Dr	Dallas	Texas	75201	US
	9	Robert	Andy	robert@example.com	8901234567	606 Spruce Ct	San Francisco	California	94101	US

US_Customers 8 x

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

```
CREATE VIEW Customer_details AS
```

```
SELECT CONCAT(First_name, ' ', Last_name) AS Full_name, Email, Phone_no, State
```

```
FROM Customer;
```

-- perform the SELECT operation for the Customer_details view

```
SELECT * FROM Customer_details;
```


Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content:				
	Full_name	Email	Phone_no	State
▶	John Jacob	johnjacob@example.com	1234567890	New York
	Annie Smith	Anniesmith@example.com	0987654321	California
	Michael Johnson	mjohnson@example.com	2345678901	Illinois
	Emily Cooper	Emilycooper@example.com	3456789012	Victoria
	William Joe	williamj@example.com	4567890123	Maharashtra
	Linda Wilson	lwilson@example.com	5678901234	British Columbia
	Jaime Taylor	jtaylor@example.com	6789012345	California
	Basim jass	bjass@example.com	7890123456	Texas
	Robert Andy	robert@example.com	8901234567	California
	Patty Thomas	pthomas@example.com	9012345678	Karnataka

Customer_details 9 x

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

```
set sql_safe_updates =0;
```

```
-- Update specific phone numbers based on customer IDs
```

```
UPDATE Customer_details
```

```
SET phone_no = '1234567890'
```

```
WHERE state = 'California';
```

-- perform the SELECT operation for the Customer_details view

```
SELECT * FROM Customer_details;
```

Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content:				
	Full_name	Email	Phone_no	State
▶	John Jacob	johnjacob@example.com	1234567890	New York
	Annie Smith	Anniesmith@example.com	1234567890	California
	Michael Johnson	mjohnson@example.com	2345678901	Illinois
	Emily Cooper	Emilycooper@example.com	3456789012	Victoria
	William Joe	williamj@example.com	4567890123	Maharashtra
	Linda Wilson	lwilson@example.com	5678901234	British Columbia
	Jaime Taylor	jtaylor@example.com	1234567890	California
	Basim jass	bjass@example.com	7890123456	Texas
	Robert Andy	robert@example.com	1234567890	California
	Patty Thomas	pthomas@example.com	9012345678	Karnataka

Customer_details 12 x


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

```
SELECT State, COUNT(*) AS Number_of_Customers  
FROM Customer  
GROUP BY State  
HAVING COUNT(*) > 5;
```

Result Grid		 Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 
State	Number_of_Customers			





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

```
SELECT State, COUNT(*) AS Number_of_Customers  
FROM Customer_details  
GROUP BY State;
```

Result Grid		 Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 
	State	Number_of_Customers		
	New York	1		
	California	3		
	Illinois	1		
	Victoria	1		
	Maharashtra	1		
	British Columbia	1		
	Texas	1		
	Karnataka	1		

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

```
SELECT *  
FROM Customer_details  
ORDER BY State ASC;
```

Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 
	Full_name	Email	Phone_no	State	
▶	Linda Wilson	lwilson@example.com	5678901234	British Columbia	
	Annie Smith	Anniesmith@example.com	1234567890	California	
	Jaime Taylor	jtaylor@example.com	1234567890	California	
	Robert Andy	robert@example.com	1234567890	California	
	Michael Johnson	mjohnson@example.com	2345678901	Illinois	
	Patty Thomas	pthomas@example.com	9012345678	Karnataka	
	William Joe	williamj@example.com	4567890123	Maharashtra	
	John Jacob	johnjacob@example.com	1234567890	New York	
	Basim jass	bjass@example.com	7890123456	Texas	
	Emily Cooper	Emilycooper@example.com	3456789012	Victoria	

Customer_details 18 ×