



Bangladesh University of Business and Technology

Submitted by :

Name : Md. Al - Amin
ID : 22234103066
Section : 06
Intake : 50
Course name : Database system
Course code : CSE 208

Submitted to :

Sworna Akter
Lecturer,BUBT

Database and table create statement:

```
create database dbs;
```

```
use dbs;
```

```
create table customer(
```

```
    Customerid int,
```

```
    Customername varchar(50),
```

```
    Address varchar(50),
```

```
    City varchar(50),
```

```
    postalcode int ,
```

```
    country varchar(50)
```

```
);
```

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0008 seconds.)

```
create database dbs;
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

⚠ Error: #1046 No database selected

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)

```
use dbs;
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

⚠ Error: #1046 No database selected

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0148 seconds.)

```
create table customer( Customerid int, Customername varchar(50), Address varchar(50), City varchar(50), postalcode int ,  
country varchar(50) );
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

Inserting value in database:

```
use dbs;
```

```
INSERT INTO customer (Customerid, Customername, Address,City,postalcode,country )
```

```
VALUES (1,"Alfreds Fuuterkiste","obere str.57","Berlin",12209,"Germany"),
```

(2,"Ana Trujillo Emparedados y helados","Avda de la Construction 2222","Mexico D.F.",05021,"Germany"),

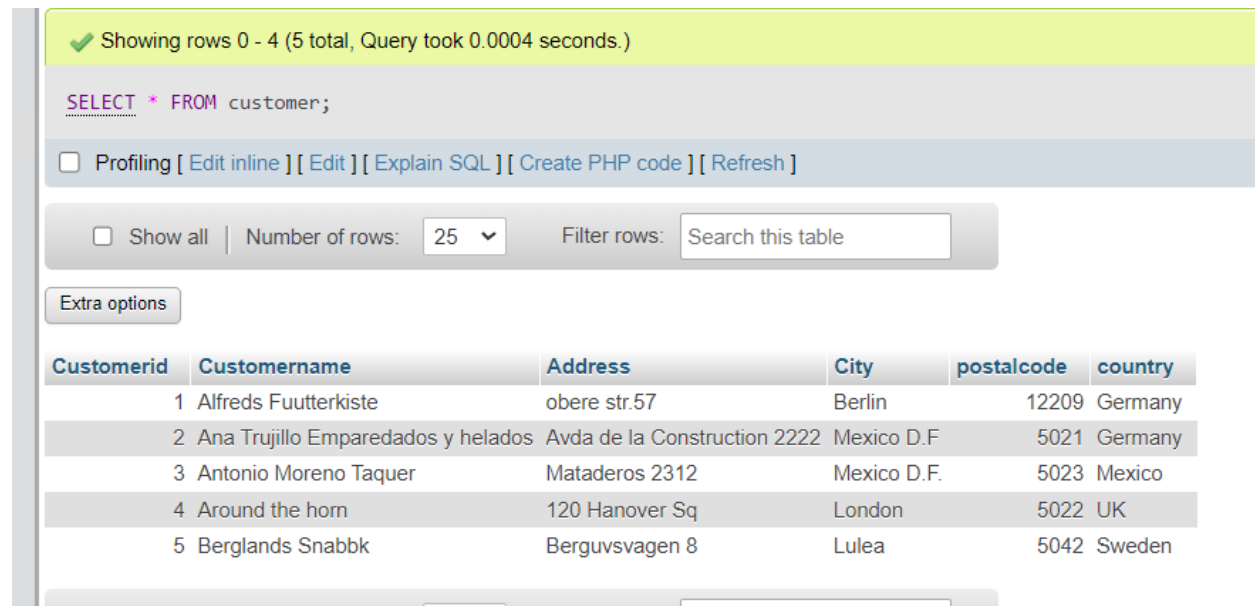
(3,"Antonio Moreno Taquer","Mataderos 2312","Mexico D.F.",05023,"Mexico"),

(4,"Around the horn","120 Hanover Sq","London",05022,"UK"),

(5,"Berglands Snabbk","Berguvsvagen 8","Lulea",05042,"Sweden");

Show table :

SELECT * FROM customer;



Showing rows 0 - 4 (5 total, Query took 0.0004 seconds.)

SELECT * FROM customer;

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

Customerid	Customername	Address	City	postalcode	country
1	Alfreds Futterkiste	obere str.57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F.	5021	Germany
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico
4	Around the horn	120 Hanover Sq	London	5022	UK
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden

Another table insertion in dbs database:

create table supplier(supplierid int,

Suppliername varchar(50),

Address varchar(50),

City varchar(50),

PostalCode int,

Country varchar(50));

Insert into supplier(Supplierid,Suppliername,Address,City,PostalCode,Country)

values (11,"Akash","33 New street","Plymouth",1225,"UK"),

```
(12,"Dipu","11 Windsoroad","Nottingham",1326,"England"),
(13,"Tushar","81 Broadway","Portsmouth",1427,"UK"),
(14,"Fardin","38 George road","Llverepool",1528,"Scotland");
```

✓ Showing rows 0 - 3 (4 total, Query took 0.0005 seconds.)

```
SELECT * from supplier;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 ▾ Filter rows:

Extra options

supplierid	Suppliername	Address	City	PostalCode	Country
11	Akash	33 New street	Plymouth	1225	UK
12	Dipu	11 Windsoroad	Nottingham	1326	England
13	Tushar	81 Broadway	Portsmouth	1427	UK
14	Fardin	38 George road	Llverepool	1528	Scotland

Customer(CustomerID,CustomerName,Address,City,PostalCode,Country)

Supplier(SupplierID,SupplierName,Address,City,PostalCode,Country)

- 1.Finding customer name and city :
- 2.Select customer information where city is London :
- 3.Add a column email and phone number in customer table :
- 4.Delete column Email from customer table:
- 5.Changing column name phonenummer to email:
- 6.Return the city from both customer and supplier table:
- 7.Add value in email column

1.Finding customer name and city :

SELECT Customername, City FROM Customer;

✓ Showing rows 0 - 4 (5 total, Query took 0.0004 seconds)

```
SELECT Customername, City FROM Customer;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)]

☐ Show all | Number of rows: ▼

Extra options

Customername	City
Alfreds Futterkiste	Berlin
Ana Trujillo Emparedados y helados	Mexico D.F
Antonio Moreno Taquer	Mexico D.F.
Around the horn	London
Berglunds Snabbk	Lulea

2.Select customer information where city is London :

SELECT * from customer where City= 'London';

✓ Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)

```
SELECT * from customer where City= 'London';
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: ▼ | Filter rows:

Extra options

Customerid	Customername	Address	City	postalcode	country
4	Around the horn	120 Hanover Sq	London	5022	UK

3.Adding email and phone number in customer table :

ALTER table customer

ADD Email varchar(50);

ALTER table customer

ADD Phonenumbr int;

Customerid	Customername	Address	City	postalcode	country	Email	Phonenumbr
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	NULL	NULL
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	NULL	NULL
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	NULL	NULL
4	Around the horn	120 Hanover Sq	London	5022	UK	NULL	NULL
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	NULL	NULL

4.Deleting column Email from customer table:

ALTER Table customer

DROP Email ;

Customerid	Customername	Address	City	postalcode	country	Phonenumbr
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	NULL
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	NULL
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	NULL
4	Around the horn	120 Hanover Sq	London	5022	UK	NULL
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	NULL

5.Changing column name phonenumbr to email:

ALTER TABLE Customer

CHANGE COLUMN Phonenumbr Email varchar(50);

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	NULL
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	NULL
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	NULL
4	Around the horn	120 Hanover Sq	London	5022	UK	NULL
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	NULL

6.Return the city from both customer and supplier table:

SELECT City FROM customer

UNION ALL

SELECT City FROM Supplier;

City
Berlin
Mexico D.F
Mexico D.F.
London
Lulea
Plymouth
Nottingham
Portsmouth
Llverepool

7.Add value in email column :

update customer set Email= "a@gmail.com" where Customerid=1;

update customer set Email= "b@gmail.com" where Customerid=2;

update customer set Email= "c@gmail.com" where Customerid=3;

update customer set Email= "d@gmail.com" where Customerid=4;

update customer set Email= "e@gmail.com" where Customerid=5;

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	b@gmail.com
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com

Sigma operation:

All the column selected

SELECT * FROM customer

WHERE Customerid=1;

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com

Only one column selected

SELECT City FROM customer

WHERE Customerid=1;

City
Berlin

Pi operation

select Customername from customer;

Customername
Alfreds Fuuterkiste
Ana Trujillo Emparedados y helados
Antonio Moreno Taquer
Around the horn
Berglands Snabbk

Union operation:

If selection does not contain unique column

SELECT Country from customer

UNION

SELECT Country from supplier;

Country
Germany
Mexico
UK
Sweden
England
Scotland

If selection contain unique column

City
Berlin
Mexico D.F.
Mexico D.F.
London
Lulea
Plymouth
Nottingham
Portsmouth
Llverepool

SET Difference

If we perform the EXCEPT operator on the above two tables to retrieve the names, it will display the distinct records only from the first table which are not in common with the records of the second table.

SELECT Country FROM customer

EXCEPT

SELECT Country from supplier;

Country
Germany
Mexico
Sweden

Cross product or Cartesian join:

for all column

SELECT *

FROM customer

CROSS JOIN supplier;

Customerid	Customername	Address	City	postalcode	country	Email	supplierid	Suppliername	Address	City	PostalCode	Country
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com	11	Akash	33 New street	Plymouth	1225	UK
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com	12	Dipu	11 Windsoroad	Nottingham	1326	England
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com	13	Tushar	81 Broadway	Portsmouth	1427	UK
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com	14	Fardin	38 George road	Liverpool	1528	Scotland
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F.	5021	Germany	b@gmail.com	11	Akash	33 New street	Plymouth	1225	UK
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F.	5021	Germany	b@gmail.com	12	Dipu	11 Windsoroad	Nottingham	1326	England
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F.	5021	Germany	b@gmail.com	13	Tushar	81 Broadway	Portsmouth	1427	UK
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F.	5021	Germany	b@gmail.com	14	Fardin	38 George road	Liverpool	1528	Scotland
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com	11	Akash	33 New street	Plymouth	1225	UK
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com	12	Dipu	11 Windsoroad	Nottingham	1326	England
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com	13	Tushar	81 Broadway	Portsmouth	1427	UK
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com	14	Fardin	38 George road	Liverpool	1528	Scotland
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com	11	Akash	33 New street	Plymouth	1225	UK
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com	12	Dipu	11 Windsoroad	Nottingham	1326	England
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com	13	Tushar	81 Broadway	Portsmouth	1427	UK
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com	14	Fardin	38 George road	Liverpool	1528	Scotland
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com	11	Akash	33 New street	Plymouth	1225	UK
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com	12	Dipu	11 Windsoroad	Nottingham	1326	England
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com	13	Tushar	81 Broadway	Portsmouth	1427	UK
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com	14	Fardin	38 George road	Liverpool	1528	Scotland

For one column

SELECT Customername

FROM customer

CROSS JOIN supplier;

Customername
Alfreds Fuuterkiste
Alfreds Fuuterkiste
Alfreds Fuuterkiste
Alfreds Fuuterkiste
Ana Trujillo Emparedados y helados
Ana Trujillo Emparedados y helados
Ana Trujillo Emparedados y helados
Ana Trujillo Emparedados y helados
Antonio Moreno Taquer
Antonio Moreno Taquer
Antonio Moreno Taquer
Antonio Moreno Taquer
Around the horn
Around the horn
Around the horn
Around the horn
Berglands Snabbk
Berglands Snabbk
Berglands Snabbk
Berglands Snabbk

Creating another two table

CREATE TABLE students_hobby(

ID INT ,

NAME VARCHAR(20) ,

HOBBY VARCHAR(20) ,

AGE INT

);

INSERT INTO students_hobby VALUES

(1, 'Vijay', 'Cricket', 18),

(2, 'Varun', 'Football', 26),

(3, 'Surya', 'Cricket', 19),

(4, 'Karthik', 'Cricket', 25),

(5, 'Sunny', 'Football', 26),

(6, 'Dev', 'Cricket', 23);

Extra options

ID	NAME	HOBBY	AGE
1	Vijay	Cricket	18
2	Varun	Football	26
3	Surya	Cricket	19
4	Karthik	Cricket	25
5	Sunny	Football	26
6	Dev	Cricket	23

CREATE TABLE students(

ID INT ,

NAME VARCHAR(20),

SUBJECT VARCHAR(20),

AGE INT,

HOBBY VARCHAR(20)

);

INSERT INTO students VALUES

(1, 'Naina', 'Maths', 24, 'Cricket'),

(2, 'Varun', 'Physics', 26, 'Football'),

(3, 'Dev', 'Maths', 23, 'Cricket'),

(4, 'Priya', 'Physics', 25, 'Cricket'),

(5, 'Aditya', 'Chemistry', 21, 'Cricket'),

(6, 'Kalyan', 'Maths', 30, 'Football');

ID	NAME	SUBJECT	AGE	HOBBY
1	Naina	Maths	24	Cricket
2	Varun	Physics	26	Football
3	Dev	Maths	23	Cricket
4	Priya	Physics	25	Cricket
5	Aditya	Chemistry	21	Cricket
6	Kalyan	Maths	30	Football

Intersection :

SELECT NAME, AGE, HOBBY FROM students_hobby

INTERSECT

SELECT NAME, AGE, HOBBY FROM students;

NAME	AGE	HOBBY
Varun	26	Football
Dev	23	Cricket

Rename:

RENAME table students to students_subject;

Creating table:

Create Table department (DEPT_NAME Varchar(20), MANAGER_NAME Varchar(255));

INSERT INTO department(DEPT_NAME,MANAGER_NAME) VALUES ("IT", "ROHAN");

INSERT INTO department(DEPT_NAME,MANAGER_NAME) VALUES ("SALES", "RAHUL");

INSERT INTO department(DEPT_NAME,MANAGER_NAME) VALUES ("HR", "TANMAY");

INSERT INTO department(DEPT_NAME,MANAGER_NAME) VALUES ("FINANCE", "ASHISH");

INSERT INTO department(DEPT_NAME,MANAGER_NAME) VALUES ("MARKETING", "SAMAY");

EMP_ID	EMP_NAME	DEPT_NAME
1	SUMIT	HR
2	JOEL	IT
3	BISWA	MARKETING
4	VAIBHAV	IT
5	SAGAR	SALES

Create Table employee(EMP_ID int,EMP_NAME Varchar(20),DEPT_NAME Varchar(255));

INSERT INTO employee(EMP_ID, EMP_NAME, DEPT_NAME) VALUES (1, "SUMIT", "HR");

INSERT INTO employee(EMP_ID, EMP_NAME, DEPT_NAME) VALUES (2, "JOEL", "IT");

INSERT INTO employee(EMP_ID, EMP_NAME, DEPT_NAME) VALUES (3, "BISWA", "MARKETING");

INSERT INTO employee(EMP_ID, EMP_NAME, DEPT_NAME) VALUES (4, "VAIBHAV", "IT");

INSERT INTO employee(EMP_ID, EMP_NAME, DEPT_NAME) VALUES (5, "SAGAR", "SALES")

Extra options	
DEPT_NAME	MANAGER_NAME
IT	ROHAN
SALES	RAHUL
HR	TANMAY
FINANCE	ASHISH
MARKETING	SAMAY

Natural join:

SELECT *

FROM employee

NATURAL JOIN department;

Extra options			
DEPT_NAME	EMP_ID	EMP_NAME	MANAGER_NAME
IT	2	JOEL	ROHAN
IT	4	VAIBHAV	ROHAN
SALES	5	SAGAR	RAHUL
HR	1	SUMIT	TANMAY
MARKETING	3	BISWA	SAMAY

☐ Show all | Number of rows: 25 ▼ | Filter rows: Search this table

CREATE TABLE workers(ID int ,Name varchar(20));

INSERT INTO workers(ID,Name)

VALUES(1,"Farhan"),

(2,"Rana"),

(3,"Hasan"),

(4,"Rakib"),

(5,"Tasnim");

ID	Name
1	Farhan
2	Rana
3	Hasan
4	Rakib
5	Tasnim

CREATE table salary (ID int ,salary int);

```
INSERT INTO salary(ID,salary)
VALUES(1,50000),
(3,25000),
(4,40000),
(6,35000),
(7,30000);
```

Extra options	
ID	salary
1	50000
3	25000
4	40000
6	35000
7	30000

Right join

```
SELECT *
FROM salary
RIGHT JOIN workers
ON salary.ID=workers.ID;
```

ID	salary	ID	Name
1	50000	1	Farhan
3	25000	3	Hasan
4	40000	4	Rakib
NULL	NULL	2	Rana
NULL	NULL	5	Tasnim

Left join

```
SELECT *
FROM salary
LEFT JOIN workers
ON salary.ID=workers.ID;
```

ID	salary	ID	Name
1	50000	1	Farhan
3	25000	3	Hasan
4	40000	4	Rakib
6	35000	NULL	NULL
7	30000	NULL	NULL

Delete a record from workers table:

DELETE FROM WORKERS WHERE ID=1;

ID	Name
2	Rana
3	Hasan
4	Rakib
5	Tasnim

ADD column:

ALTER TABLE workers ADD Salary int;

ID	Name	Salary
2	Rana	NULL
3	Hasan	NULL
4	Rakib	NULL
5	Tasnim	NULL

Delete column :

ALTER table workers DROP COLUMN Salary;

ID	Name
2	Rana
3	Hasan
4	Rakib
5	Tasnim

Rename column(change column name with datatype):

ALTER TABLE workers

CHANGE ID W_ID int;

W_ID	Name
2	Rana
3	Hasan
4	Rakib
5	Tasnim

CREATE TABLE students_info(Id int,Name varchar(20),Age int,Tfee int);

INSERT INTO students_info(Id,Name,Age,Tfee)

VALUES

(1,"Rasel",25,70000),

(2,"Ashik",22,40000),

(3,"Shaon",25,90000),

(4,"Hasib",26,60000),

(5,"Sohel",19,30000);

Id	Name	Age	Tfee
1	Rasel	25	70000
2	Ashik	22	40000
3	Shaon	25	90000
4	Hasib	26	60000
5	Sohel	19	30000

Count: SELECT COUNT(*) FROM students_info;

COUNT(*)
5

Sum: SELECT SUM(Tfee)

FROM students_info;

SUM(Tfee)
290000

Distinct:

```
SELECT DISTINCT (Age)
FROM students_info;
```

Age
25
22
26
19

Avg:

```
SELECT AVG(Age)
FROM students_info;
```

AVG(Age)
23.4000

Max:

```
SELECT MAX(Tfee)
FROM students_info;
```

MAX(Tfee)
90000

MIN

```
SELECT MIN(Tfee)
FROM students_info;
```

MIN(Tfee)
30000

The following SQL statement lists the number of customers in each country:

Group by:

Here, this operations are done for customer table

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	b@gmail.com
3	Antonio Moreno Taquer	Mataderos 2312	Mexico D.F.	5023	Mexico	c@gmail.com
4	Around the horn	120 Hanover Sq	London	5022	UK	d@gmail.com
5	Berglands Snabbk	Berguvsvagen 8	Lulea	5042	Sweden	e@gmail.com

```
SELECT COUNT(Customerid),country
```

```
FROM customer
```

```
GROUP BY Country;
```

COUNT(Customerid)	country
2	Germany
1	Mexico
1	Sweden
1	UK

Like operator:

```
SELECT * from Customer
```

```
WHERE country LIKE 'Germany';
```

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	b@gmail.com

```
SELECT *FROM customer
```

```
where Country like 'G%';
```

Extra options						
Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Fuuterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	b@gmail.com
<input type="checkbox"/> Show all Number of rows: 25 Filter rows: Search this table						

```
SELECT Country FROM customer
```

```
where Country like '%e%';
```

Country
Germany
Germany
Mexico
Sweden

The ascending descending done for below table:

Id	Name	Age	Tfee
1	Rasel	25	70000
2	Ashik	22	40000
3	Shaon	25	90000
4	Hasib	26	60000
5	Sohel	19	30000

Ascending:

```
SELECT *
FROM students_info
order by tfee ASC;
```

Id	Name	Age	Tfee
5	Sohel	19	30000
2	Ashik	22	40000
4	Hasib	26	60000
1	Rasel	25	70000
3	Shaon	25	90000

Descending:

```
SELECT *
FROM students_info
order by tfee DESC;
```

Id	Name	Age	Tfee
3	Shaon	25	90000
1	Rasel	25	70000
4	Hasib	26	60000
2	Ashik	22	40000
5	Sohel	19	30000

Having :

The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions.

This operation is done for customer table

```
SELECT COUNT(Customerid), Country
FROM customer
GROUP BY Country
HAVING COUNT(Customerid) <2;
```

COUNT(Customerid)	Country
1	Mexico
1	Sweden
1	UK

Limit:

```
SELECT * FROM customer
LIMIT 2;
```

Customerid	Customername	Address	City	postalcode	country	Email
1	Alfreds Futterkiste	obere str.57	Berlin	12209	Germany	a@gmail.com
2	Ana Trujillo Emparedados y helados	Avda de la Construction 2222	Mexico D.F	5021	Germany	b@gmail.com

Subquery problem:

Given table

Id	Name	Age	Tfee
1	Rasel	25	70000
2	Ashik	22	40000
3	Shaon	25	90000
4	Hasib	26	60000
5	Sohel	19	30000

Q1.Find second largest?

```
SELECT MAX(Tfee)
FROM students_info
WHERE Tfee NOT IN (SELECT MAX(Tfee) FROM students_info);
```

Or

```
SELECT MAX(Tfee)
FROM students_info
WHERE Tfee < (SELECT MAX(Tfee) FROM students_info);
```

MAX(Tfee)
70000

Q2.Find third largest?

```
SELECT MAX(Tfee)
FROM students_info
WHERE Tfee < (
SELECT MAX(Tfee)
FROM students_info
WHERE Tfee < (SELECT MAX(Tfee) FROM students_info));
```

Extra options
MAX(Tfee)
60000

