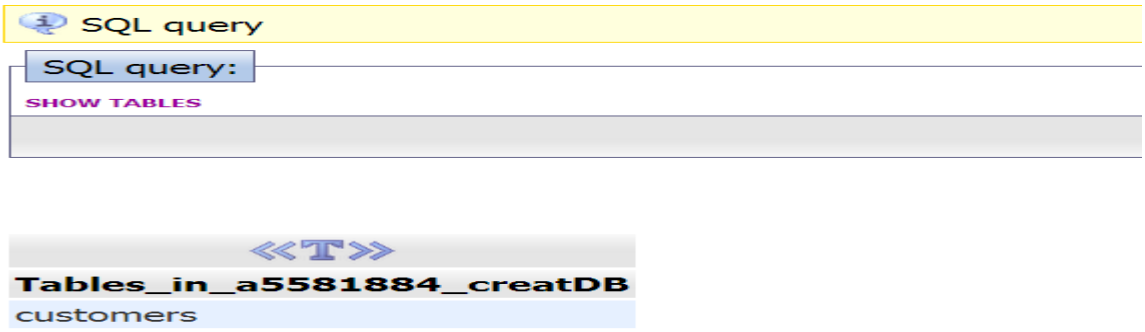
SQL LANGUAGE (MySQL)

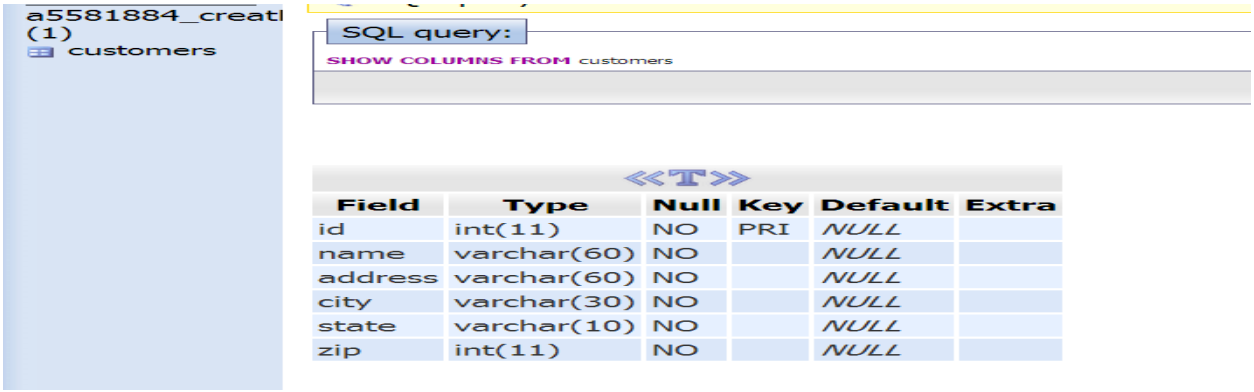
SHOW and SELECT COMMAND:

Install 000webhost.com click phpMyDomain….

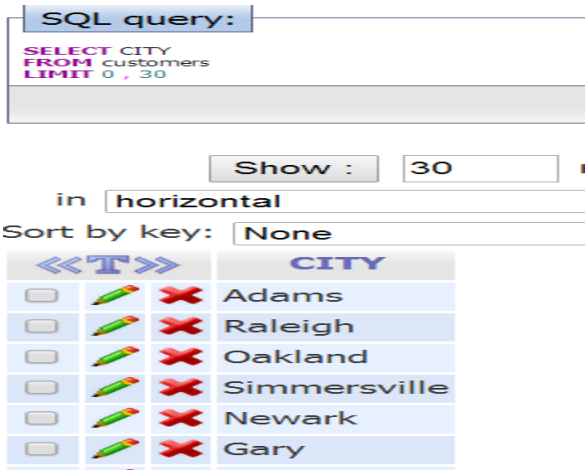
Click SQL and type coding …. It displays the Tables name



It displays the column names from the customers table ….



Displays the column details of the customer table so we use SELECT:



If you are executing multiple queries then you must use semicolon at the end of the queries ….

SELECT city FROM customers;

SELECT state FROM customers;

SQL is **not case sensitive** but use only capitals ……

Displaying multiple columns ……

SELECT name,city FROM customers

Displaying all the columns in the table…..

SELECT * FROM customers

Displaying the column data which eliminates the duplicate of rows … just print unique rows which doesn't repeat twice or more….

SELECT DISTINCT city FROM customers

Displaying the column data with a limit of 5 rows…..

SELECT * FROM customers LIMIT 5

Displaying the column data with a limit of 10 rows but from different position say from 13th row….

Computer reads from 0 so the 13th row is taken as from 14th row…..

SELECT * FROM customers LIMIT 13,10

Sort by key: None

id	name	address	city	state	zip
14	Jeremy White	3954 Brentwood Dr	Seattle	WA	99037
15	Omar Badshah	6801 Regina Cir	Madison	WI	53209
16	Preston Harrison	104 Main St	Denver	CO	81712
17	Manuel Rodriguez	99543 Westin Blvd	Provo	UT	85478
18	David Jones	5488 W 34th St	Boston	MA	2104
19	Nick Flanders	3486 Happy Trails Dr	Springfield	OH	45872
20	Paul Brown	3290 Pennsylvania Ave	Chicago	IL	61208
21	Sara Rehm	7746 Wysong Ave	Detroit	MI	48913
22	Haley Carter	2957 Princess Way	Portland	OR	97532
23	Julian Thomas	5564 Dandy Trail	Santa Fe	NM	81543

Check All / Uncheck All *With selected:*

Sorting the column data

SELECT city FROM customers ORDER BY city

Sorting multiple column data… in this query it first sort the cities and then sort the names according to the cities ….

SELECT state,name FROM customers ORDER BY state,name

state	name
AK	Corey Smith
AK	Ruth Bolen
AL	Crystal Jarvis
AL	Perry Jordan
AL	Thomas Jackson
AR	Katherine Cain
AZ	Debra Talkington
AZ	Sherry Gibbons
CA	Desmond Rafferty
CA	Donna Bradley
CA	Jack Nicholson

Reverse sorting of the values in the column……

SELECT state,zip FROM customers ORDER BY zip DESC

AK	99831
WA	99753
WA	99037
OR	97532
CA	97221
CA	93980
HI	93525
CA	92953

SELECT state,zip FROM customers ORDER BY zip ASC (ascending to descending… it is done default so no need…..)

Display the details of the customer who has the largest ID number and need to display only one name…

SELECT name,id FROM customers ORDER BY id DESC LIMIT 1

name	id
Lucy Bronson	96

Filtering stuffs…. Displaying particular data from the column with a specific data

SELECT id,name FROM customers WHERE id=37

id	name
37	Terry Mitchell

Displaying the values expect 37…..

SELECT id,name FROM customers WHERE id!=37

Displaying the values less than 7

SELECT id,name FROM customers WHERE id < 7

SELECT id,name FROM customers WHERE id <= 7

Displaying the values more than 7

SELECT id,name FROM customers WHERE id > 7

SELECT id,name FROM customers WHERE id >= 7

Displaying the values between the certain range…..

SELECT id,name FROM customers WHERE id BETWEEN 2 AND 5

id	name
2	Noah Parker
3	Kelsey Burger
4	Corey Smith
5	Harry Potter

Displaying the names who resides in STATE……

SELECT name,state FROM customers WHERE state = 'CA'

name	state
Kelsey Burger	CA
Jack Nicholson	CA
Desmond Rafferty	CA
Paula Barker	CA
Donna Bradley	CA
Patsy Cline	CA

AND OR conditions:

Displaying the elements which both condition are to be true…

SELECT id,name,state FROM customers WHERE state='FL'

id	name	state
12	Penny Green	FL
39	James Hamilton	FL
49	Evan Bayh	FL
55	Michael Orlando	FL

SELECT id,name,state FROM customers WHERE state='FL' AND city='Miami'

id	name	state
49	Evan Bayh	FL

Displaying the elements which any one of the condition is to be true…

SELECT name,city,state FROM customers WHERE city='Miami' OR state='FL'

name	city	state
Penny Green	Orlando	FL
James Hamilton	Ft Lauderdale	FL
Evan Bayh	Miami	FL
Michael Orlando	Pensacola	FL
Carol Brown	Ft Myers	FL
Tanya Hicks	Orlando	FL

IN ----- IN OUT option:

SELECT name,state FROM customers WHERE state='CA' OR state='FL' OR state='NY'

This IN option helps in eliminating the OR bunches …..

SELECT name,state FROM customers WHERE state IN ('CA','FL','NY')

This NOT IN option displays the data other than the entered states….

SELECT name,state FROM customers WHERE state NOT IN ('CA','FL','NY')%

SEARCHING IN DATABASE USING WILDCARD FILTER (%) option:

SELECT name FROM items WHERE name LIKE 'new%'

This % acts as the word which starts with “new” displays it …..

name
New gym socks
New ipad stolen from best buy
new curtain for bedroom
newspaper

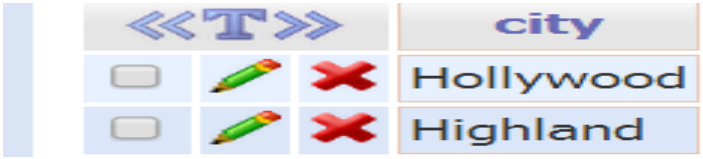
SELECT name FROM items WHERE name LIKE '%computer%'

This displays the before and back words stacked to it….

name
Brand New iMac Computer
awesome alien computer game
supercomputer
computer

SELECT city FROM customers WHERE city LIKE 'h%d'

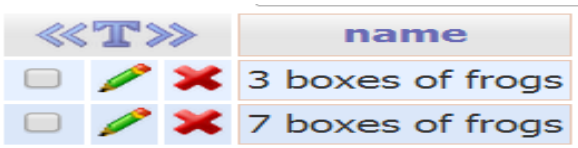
This displays the word which starts with “h” and ends with “d”



city
Hollywood
Highland

SELECT name FROM items WHERE name LIKE '_ boxes of frogs'

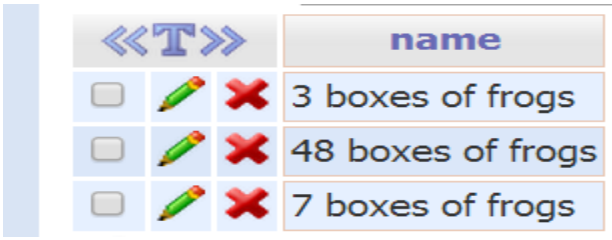
This command contains “_” which displays only one character ….



name
3 boxes of frogs
7 boxes of frogs

SELECT name FROM items WHERE name LIKE '% boxes of frogs'

In table we have 48 boxes of frogs …. In order to display that too we need to use “%” wildcard

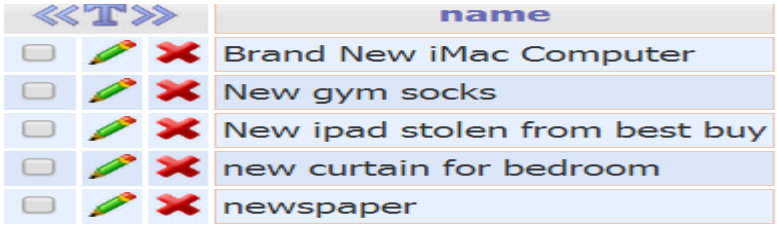


name
3 boxes of frogs
48 boxes of frogs
7 boxes of frogs

SEARCHING IN DATABASE USING WILDCARD FILTER (%) REGULAR EXPRESSIONS:

SELECT name FROM items WHERE name REGEXP 'new'

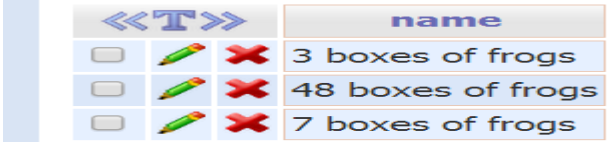
This command displays everything which has a word “new”



name
Brand New iMac Computer
New gym socks
New ipad stolen from best buy
new curtain for bedroom
newspaper

SELECT name FROM items WHERE name REGEXP '.boxes of frogs'

This command “.” Plays as a space before “boxes of frogs”



name
3 boxes of frogs
48 boxes of frogs
7 boxes of frogs

SELECT name FROM items WHERE name REGEXP 'gold | car'

This command works a OR statement and displays the list which has either gold or car…

name
traditional carpet
gold necklace
used car
gold earring

SELECT name FROM items WHERE name REGEXP '[12345] boxes of frogs'

This command displays the set of numbers of “boxes of frogs” you may also use

SELECT name FROM items WHERE name REGEXP '[1-5] boxes of frogs'

name
3 boxes of frogs

SELECT name FROM items WHERE name REGEXP '[^1-5] boxes of frogs'

This command displays the numbers other than the numbers in the set ….

name
48 boxes of frogs
7 boxes of frogs

CREATING CUSTOM COLUMNS:

SELECT CONCAT(city,',',state) AS newAddress FROM customers

This command concate two columns into one column and “AS” displays a name to it ….

newAddress
Adams,NY
Raleigh,NC
Oakland,CA
Simmersville,AK
Newark,NJ
Gary,IN
Augusta,GA

SELECT name,cost,cost+1 AS salePrice FROM items

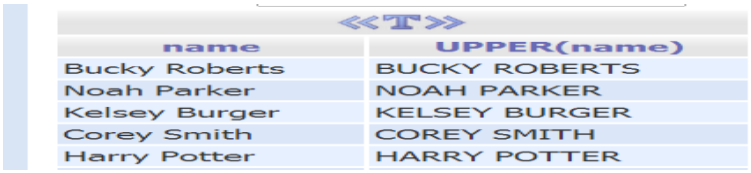
This command adds a new column with addition of price in it … you can use “+ - * /” etc..

name	cost	salePrice
Brand New iMac Computer	149.99	150.990005493164
used diaper from my sister	2.04	3.03999996185303
Fresh apple pie	14.99	15.9899997711182
New gym socks	2.34	3.33999991416931
Weedwacker only slightly used	4.56	5.55999994277954
New ipad stolen from best buy	399	400
Book about having babies	21.34	22.3400001525879
Woman Jeans	49.5	50.5
traditional carpet	25.45	26.4500007629395
3 boxes of frogs	30.49	31.4899997711182

FUNCTIONS:

SELECT name,UPPER(name) FROM customers

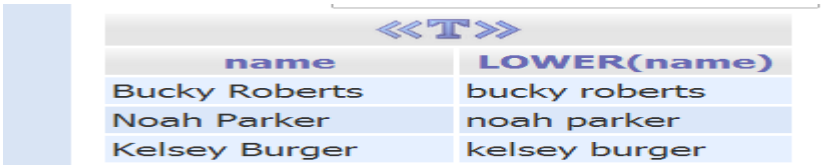
This command change the characters to uppercase….



name	UPPER(name)
Bucky Roberts	BUCKY ROBERTS
Noah Parker	NOAH PARKER
Kelsey Burger	KELSEY BURGER
Corey Smith	COREY SMITH
Harry Potter	HARRY POTTER

SELECT name,LOWER(name) FROM customers

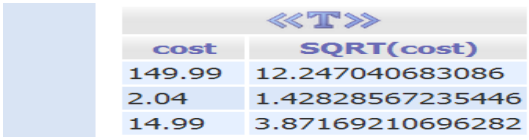
This command change the characters to lowercase…



name	LOWER(name)
Bucky Roberts	bucky roberts
Noah Parker	noah parker
Kelsey Burger	kelsey burger

SELECT cost,SQRT(cost) FROM items

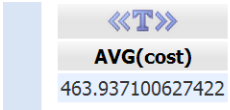
This command calculate the multiple operation of using SQRT



cost	SQRT(cost)
149.99	12.247040683086
2.04	1.42828567235446
14.99	3.87169210696282

SELECT AVG(cost) FROM items

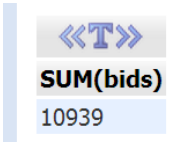
This command calculate the average



AVG(cost)
463.937100627422

SELECT SUM(bids) FROM items

This command calculate the total of the numbers …



SUM(bids)
10939

AGGREGATE FUNCTIONS:

SELECT COUNT(name) AS totalItems FROM items WHERE seller_id=12

This command calculates the number of items (listings) selling by the customer…

totalItems
5

SELECT COUNT(*) AS totalItems,

MAX(cost) AS maximum,

MIN(cost) AS minimum

FROM items WHERE seller_id=12

This command calculates the min and max of the items sold by the customer

totalItems	maximum	minimum
5	5700.5	85.1999969482422

GROUP BY -- HAVING:

SELECT seller_id, COUNT(*) AS countTable FROM items

seller_id	countTable
32	100

SELECT seller_id, COUNT(*) AS countTable FROM items WHERE seller_id=4

seller_id	countTable
4	1

SELECT seller_id, COUNT(*) AS countTable FROM items WHERE seller_id=3

SELECT seller_id, COUNT(*) AS countTable FROM items WHERE seller_id=5

In order to avoid this bunch of codes … we use single command which number of items sold by each customer ….

SELECT seller_id, COUNT(*) AS countTable FROM items GROUP BY seller_id

seller_id	countTable
1	2
2	2
3	1
4	1
6	3

SELECT seller_id, COUNT(*) AS countTable FROM items GROUP BY seller_id

HAVING count(*)>3

This command lists the count table which the seller sells more than 3 items ….

seller_id	countTable
12	5

seller_id	countTable
14	4
15	4
18	4

SELECT seller_id, COUNT(*) AS countTable FROM items GROUP BY seller_id

HAVING count(*)>3 ORDER BY countable

This command sorts the countTable….

seller_id	countTable
18	4
15	4
14	4
12	5

SUB QUERIES:

SELECT name,cost FROM items WHERE cost>(

SELECT AVG(cost) FROM items

) ORDER BY cost

This command displays the cost values more than the average (sub query)

name	cost
iphone	547
camera	550.7
electric oven	645
refrigerator	657.49

SELECT seller_id FROM items WHERE name LIKE '%boxes of frogs'

seller_id
68
6
18

```
SELECT name,MIN(cost) FROM items WHERE name LIKE '%boxes of frogs'
AND seller_id IN(
        SELECT seller_id FROM items WHERE name LIKE '%boxes of frogs'
)
```

This command displays the cheapest cost of the item from the seller ……

«T»

name	MIN(cost)
3 boxes of frogs	30.4899997711182

Joining tables:

This command joins the two tables with some relations. For e.g.: customer id is related to the seller id. In this bucky Roberts sells two items…..

```
select customers.id,customers.name,items.name,items.cost
from customers,items
where customers.id = seller_id
ORDER by customers.id
```

id	name	name	cost
1	Bucky Roberts	used diaper from my sister	2.04
1	Bucky Roberts	bucket	2.5
2	Noah Parker	babyfoot	376.7
2	Noah Parker	baby seat	145.78
3	Kelsey Burger	lipstick	24.75
4	Corey Smith	baby soap	12.7
6	Henry Jackson	48 boxes of frogs	74.29
6	Henry Jackson	microwave	150.29
6	Henry Jackson	shampoing	12.8
7	Cynthia Alvarez	blue dress size 40	88.9
7	Cynthia Alvarez	scarf	11.9

Nickname for tables:

```
Select i.seller_id, i.name, c.state, c.city
From customers as c, items as i
```

This commands helps in generating nicknames or assigning names for tables.

OUTER joins:

This command generates all the users and their items still who doesn't list items in the database.

```
select customers.name, items.name from customers left outer join
items on customers.id = seller_id
```

RIGHT outer join displays the items still the seller is exited or banned from the website.

select customers.name, items.name from customers right outer join

items on customers.id = seller_id

name	name
Bucky Roberts	used diaper from my sister
Bucky Roberts	bucket
Noah Parker	baby seat
Noah Parker	babyfoot
Kelsey Burger	lipstick
Corey Smith	baby soap
Harry Potter	NULL
Henry Jackson	48 boxes of frogs
Henry Jackson	shampoing
Henry Jackson	microwave
Cynthia Alvarez	pan

LEFT

NULL	cushion
Lani Kulana	refrigerator
Sherry Gibbons	gold necklace
Cynthia Alvarez	pan

RIGHT

UNION

This command displays the union of two queries into single column.

select name, cost, bids from items where cost>1000

UNION

select name, cost, bids from items where bids>190

name	cost	bids
conveyor belt	1120.75	4
used car	5700.5	135
piano	1800.4	147
machintosh	3845	107

select name, cost, bids from items where bids>190

UNION ALL

select name, cost, bids from items where cost>1000

This command displays the duplicates items too.

Full Text Searching:

First you have write this query….

```
alter table items add fulltext(name)
```

Your SQL query has been executed successfully

```
select name,cost from items where match(name) against ('baby')
```

name	cost
baby coat	89.99
baby seat	145.78
baby soap	12.7
baby bottle	27.98

```
select name, cost from items where match(name) against('+baby -coat' in boolean mode)
```

This query searches for baby items and eliminates the coat word with the minus sign…..

name	cost
baby seat	145.78
baby soap	12.7
baby bottle	27.98

INSERT

This command helps in inserting a row or a column.

```
insert into items(id,name,cost,seller_id,bids) values('101','fish nuggets','10.20','1','10')
```

Inserted rows: 1

```
SELECT * FROM items
```

id	name	cost	seller_id	bids
101	fish nuggets	10.2	1	10
100	magazine	3.5	8	152
99	dvd	126.84	14	113

INSERT MULTIPLE ROWS:

This command helps in inserting multiple rows.

```
insert into items(id,name,cost,seller_id,bids) values
('102','beef','7.02','2','84'),
('103','shrimp','20.02','3','30'),
('104','pasta','2.02','4','20')
```

Inserted rows: 3

This command helps in inserting multiple rows from another table.

```
insert into items(id,name,cost,seller_id,bids)
Select id,name,cost,seller_id,bids from faketable
```

UPDATE

This command updates the items in the row.

```
update items set name='beef pasta' where id=102
```

Affected rows: 1

id	name	cost	seller_id	bids
104	pasta	2.02	4	20
103	shrimp	20.02	3	30
102	beef pasta	7.02	2	84

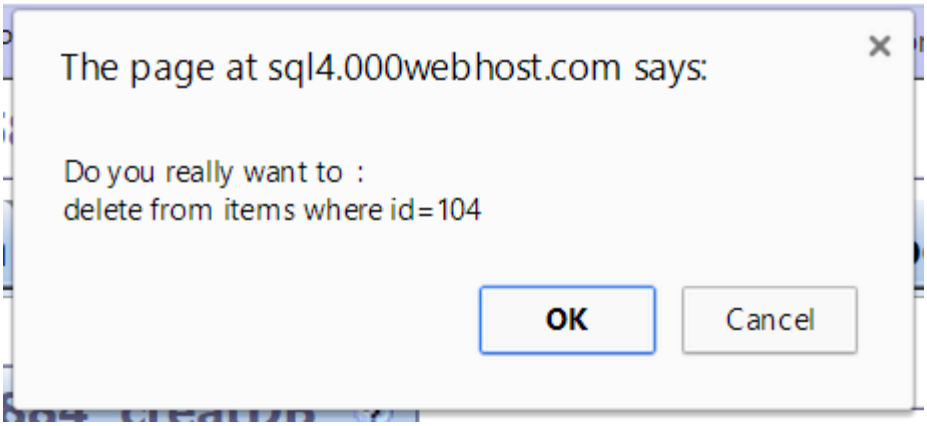
```
update items set name='beef pasta' bids = 100 where id=102
```

This command updates the items in the multiple columns.

DELETE

This command deletes the row from the table.

```
delete from items where id=104
```



Deleted rows: 1

CREATE A NEW TABLE

This command helps in creating a new table.

```
create table newtable(
id int,
username varchar(30),
password varchar(20),
primary key(id)
)
```

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
	id	int(11)			No	0		
	username	varchar(30)	latin1_general_ci		Yes	*NULL*		
	password	varchar(20)	latin1_general_ci		Yes	*NULL*		

NOT NULL AUTO_INCREMENT

This command which has NOT NULL this will not allow to leave the username and password blank……

This command which has AUTO_INCREMENT this will increment automatically the id…..

```
create table freshtable(
id int NOT NULL AUTO_INCREMENT,
username varchar(20) NOT NULL,
password varchar(30) NOT NULL,
primary key(id)
)
```

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
	id	int(11)			No		auto_increment	
	username	varchar(20)	latin1_general_ci		No			
	password	varchar(30)	latin1_general_ci		No			

<<T>>	id	username	password
	1	yunus	irshad
	2	zakira	banu

ADDING COLUMN:

This command helps in adding a new column in the table.

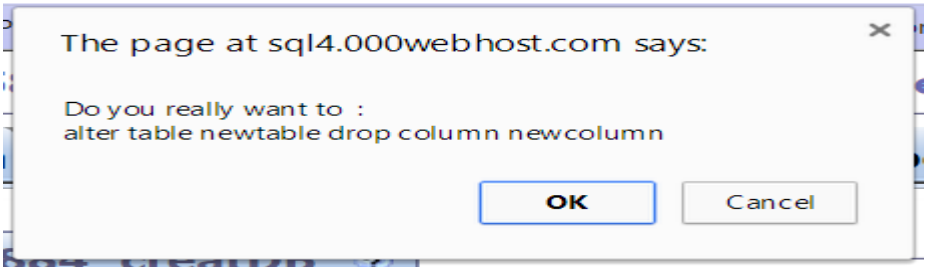
```
alter table newtable add newcolumn varchar(30)
```

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
	id	int(11)			No	0		
	username	varchar(30)	latin1_general_ci		Yes	*NULL*		
	password	varchar(20)	latin1_general_ci		Yes	*NULL*		
	newcolumn	varchar(30)	latin1_general_ci		Yes	*NULL*		

ALTER

```
alter table newtable drop column newcolumn
```

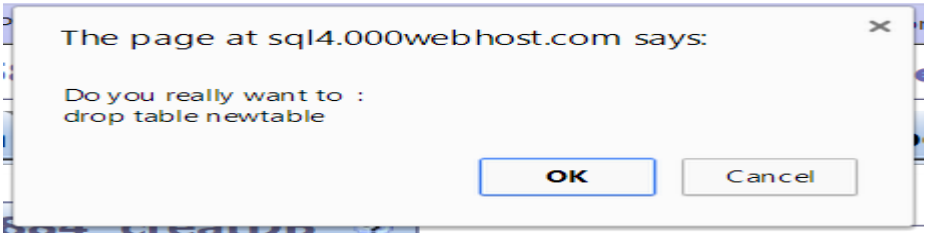
This command helps in dropping or deleting the new column from the table.



DROP

drop table newtable

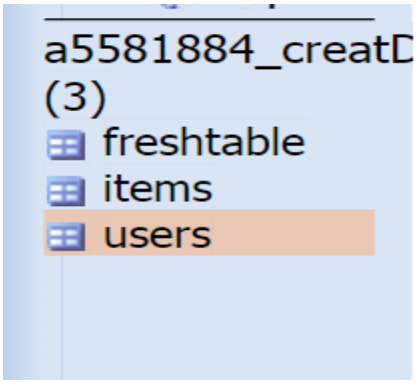
This command helps in dropping the table.



RENAME

This command helps in renaming the table.

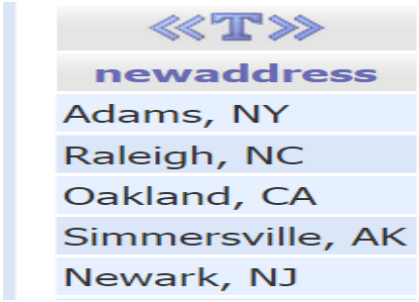
rename table customers to users

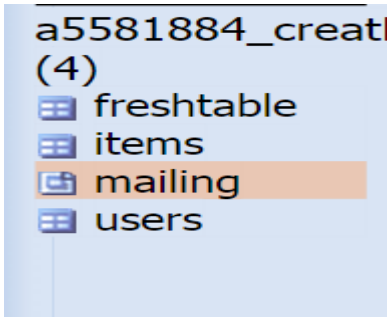


VIEWS

This command is nothing but to temporary view of the table of the command.

select concat(city,', ',state) as newaddress from users





create view mailing AS

```
select concat(city,', ',state) as newaddress from users
```