

MySQL Database

Handbook

6/5/2016

Computer Science Middlesex University London
Povilas Urbonas

Contents

1 - Introduction to Databases	4
2 - Getting a MySQL Server.....	4
3 - Creating a Database	4
4 Show and SELECT	6
5 Basic Rules for SQL Statements	7
6 - Getting Multiple Columns.....	8
7 - DISTINCT and LIMIT.....	9
8 - Sorting Results	11
9 - Sort Direction.....	14
10 - Basic Data Filtering and WHERE	17
11 Advanced Filtering Using AND and OR.....	19
12 - Are you IN or are you NOT IN	20
13 - How Search Engines Work	23
14 - More on Wildcards	24
15 - Regular Expressions	24
16 - Creating Custom Columns	26
17 – Functions.....	28
18 - More on Aggregate Functions	29
19 - GROUP BY	31
20 – Subqueries.....	32
21 - Another Subquery Example	34
22 - How to Join Tables	35
23 - Outer Joins	36
24 – UNION	37
25 - Full-Text Searching.....	38
26 - INSERT INTO	39
27 - How to Insert Multiple Rows	41
28 - UPDATE & DELETE.....	42
29 - CREATE TABLE	43
30 - NOT NULL & AUTO INCREMENT	44
31 - ALTER DROP RENAME TABLE	46
32 – View.....	48
33 - Final.....	52

1 - Introduction to Databases

SQL is the language databases use

MySQL is a program that understand SQL

Other software, but MySQL is the best!

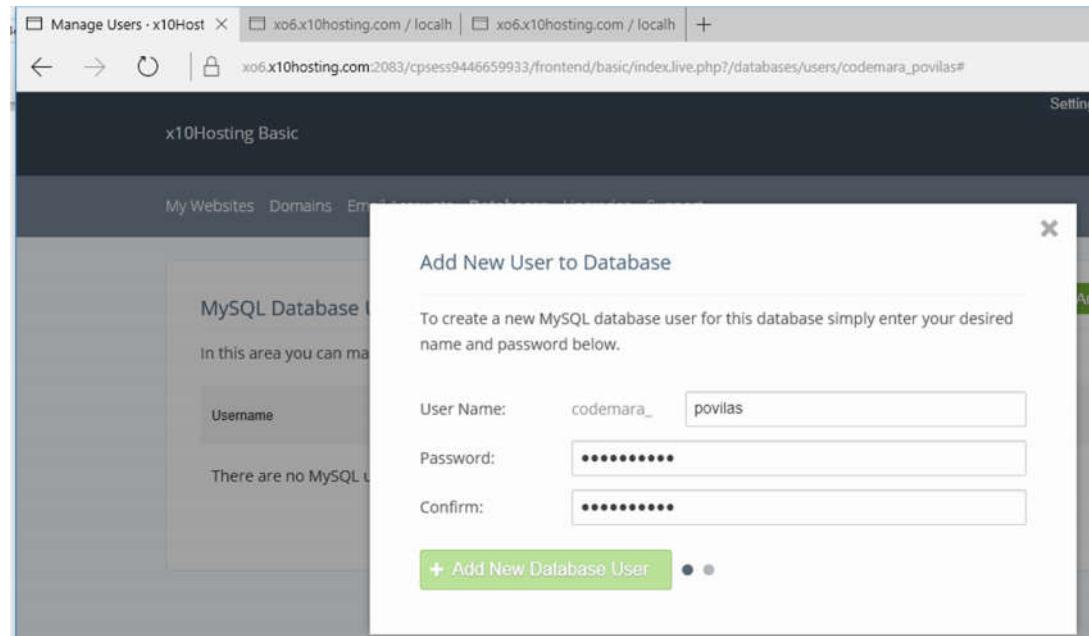
2 - Getting a MySQL Server

<https://x10hosting.com/>

- Username: codemara
- Your Domain: <http://codemarathoner.x10host.com/>
-

povilas.urbonas85@gmail.com

3 - Creating a Database



ting Basic

Visit C

Update Database Privileges

Tick the checkboxes alongside privileges that the user should have against the database.

Database Name: codemara_povilas
User Name: codemara_povilas

Grant All Privileges

<input type="checkbox"/> INSERT	<input type="checkbox"/> UPDATE	<input type="checkbox"/> DELETE
<input type="checkbox"/> SELECT	<input type="checkbox"/> CREATE	<input type="checkbox"/> INDEX
<input checked="" type="checkbox"/> DROP	<input type="checkbox"/> REFERENCES	<input type="checkbox"/> LOCK TABLES
<input type="checkbox"/> EXECUTE	<input type="checkbox"/> CREATE ROUTINE	<input type="checkbox"/> SHOW VIEW
<input type="checkbox"/> CREATE VIEW	<input type="checkbox"/> ALTER	<input type="checkbox"/> TRIGGER
<input type="checkbox"/> CREATE TEMPORARY TABLES		

Update Privileges

x10hosting.com:2083/cpsees944663993/frontend/basic/index.live.php?/databases

Settings Switch Theme Logout ONLINE

Visit Community

MySQL Databases

+ Add MySQL Database

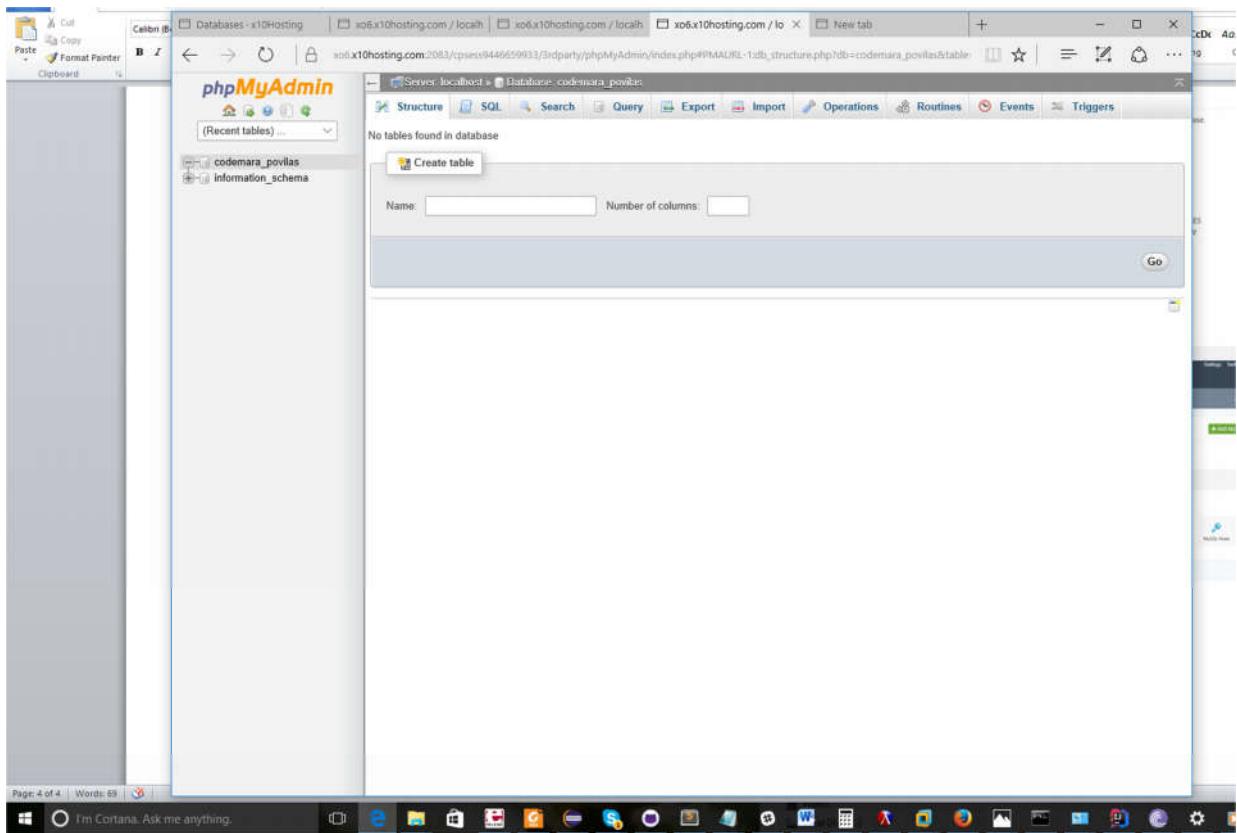
In this area you can manage MySQL databases on your web hosting account.

Note: PostgreSQL database management is available with cPanel X3.

Database Name	Disk Usage	MySQL Users
codemara_povilas	0.00 MB	codemara_povilas

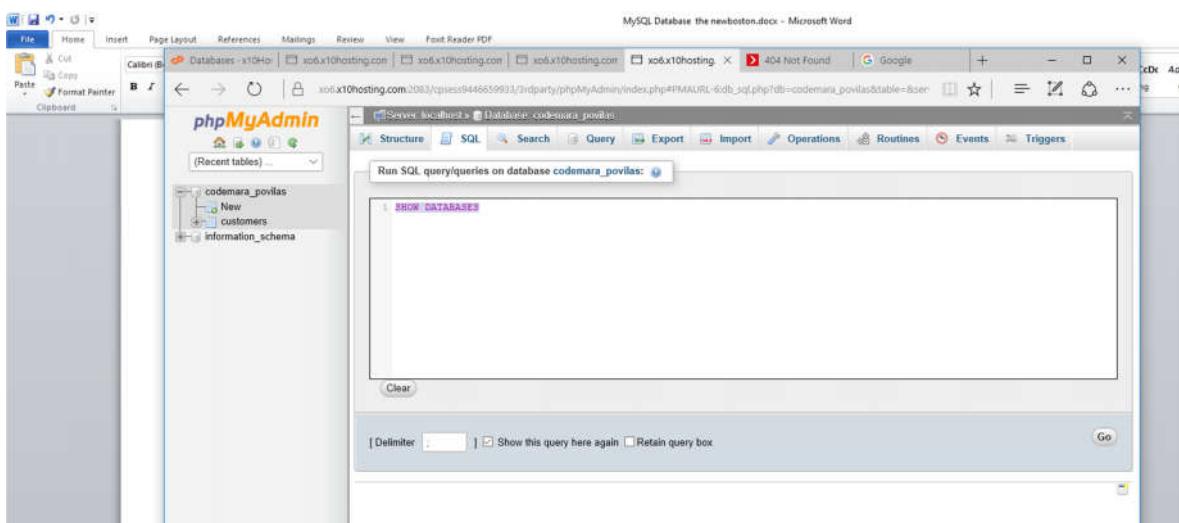
Using 1 SQL databases of your 2 allowed.

MySQL Hosts PHPMyAdmin



4 Show and SELECT

SHOW DATABASES



Databases - x10Hosting | xo6.x10hosting.com / localhost | xo6.x10hosting.com / localhost | xo6.x10hosting.com / localhost

← → ⌂ 🔒 xo6.x10hosting.com:2083/cpsess9446659933/3rdparty/phpMyAdmin/index.php#PMAURL-8:db_sql.php?db=codemara_povilas

phpMyAdmin

Structure SQL Search Query Export Import

Show query box
Your SQL query has been executed successfully

SHOW TABLES

+ Options

Tables_in_codemara_povilas

customers

Query results operations

Print view Print view (with full texts) Create view

SHOW COLUMNS FROM customers

SELECT city FROM customers

5 Basic Rules for SQL Statements

SELECT city FROM customers;

SELECT id FROM customers;

Databases - x10Hosting | xo6.x10hosting.com / localhost | xo6.x10hosting.com / localhost | xo6.x10hosting.com / localhost

← → ⌂ 🔒 xo6.x10hosting.com:2083/cpsess9446659933/3rdparty/phpMyAdmin/index.php#PMAURL-15:tbl_sql

phpMyAdmin

Browse Structure SQL Search Insert Export

Run SQL query/queries on database codemara_povilas:

```
1 SELECT city FROM customers;
2 SELECT id FROM customers;
```

6 - Getting Multiple Columns

SELECT name, zip FROM customers

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The 'SQL' tab is selected. A warning message states: "This table does not contain a unique column". Below it, a green success message says: "Showing rows 0 - 29 (96 total, Query took 0.000 sec)". The SQL query entered is:

```
SELECT name, zip  
FROM customers  
LIMIT 0 , 30
```

Below the query, there are navigation buttons: '1 <' (dropdown), 'Show all', '>', '>>', and 'Show'. A table titled '+ Options' displays the results of the query:

name	zip
Gerald Foster	34479
Kevin Hawkins	30351
Diana Davis	58122
Heather Gonzalez	87105
Terry Hart	91520
Nancy Gomez	31605
Beverly Wood	20557

7 - DISTINCT and LIMIT

The screenshot shows the phpMyAdmin interface. The left sidebar lists databases: codemara_povilas and information_schema. The right panel has tabs for Structure, SQL, Search, Query, Export, Import, and Help. The SQL tab is active, showing the query: `1 SELECT DISTINCT state FROM customers`. The results pane is empty.

Not showing duplicates

`SELECT id, name FROM customers`

`SELECT id, name FROM customers LIMIT 5`

The screenshot shows the phpMyAdmin interface. The left sidebar lists databases: codemara_povilas and information_schema. The right panel has tabs for Structure, SQL, Search, Query, Export, Import, and Help. The SQL tab is active, showing the query: `SELECT id, name FROM customers LIMIT 5`. A message above the results says: `Showing rows 0 - 4 (5 total, Query took 0.0006 sec)`. The results pane displays the following table:

id	name
1	Gerald Foster
2	Kevin Hawkins
3	Diana Davis
4	Heather Gonzalez
5	Terry Hart

```
SELECT id, name FROM customers LIMIT 5, 10
```

The screenshot shows the phpMyAdmin interface for a database named 'xo6'. The left sidebar lists databases: 'codemara_povilas' and 'information_schema'. The main area has tabs for 'Structure' and 'SQL', with 'SQL' selected. A message box says 'This table does not contain a primary key yet.' Below it, a green box says 'Showing rows 5 - 14 (10 total)'. The SQL query entered is:

```
SELECT id, name  
FROM customers  
LIMIT 5 , 10
```

Below the query, a table displays the results:

+ Options	id	name
	6	Nancy Gomez
	7	Beverly Wood
	8	Fred Roberts
	9	Maria Cox
	10	Evelyn Warren
	11	Shirley Vasquez
	12	Maria Fuller
	13	Deborah Parker
	14	Linda Frazier
	15	Randy Ray

8 - Sorting Results

SELECT address FROM customers

SELECT customers.address FROM customers

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The 'SQL' tab is selected. A query has been run:

```
SELECT customers.address  
FROM customers  
LIMIT 0 , 30
```

The results show 29 rows of address data:

address
994 Transport Plaza
03231 Dwight Hill
35 Fairfield Parkway
1 4th Trail
353 Messerschmidt Pass
121 Southridge Lane
21 Bluestem Trail
9 Main Crossing
1 Crest Line Pass

```
SELECT name FROM customers
```

```
SELECT name FROM customers ORDER BY name
```

The screenshot shows the phpMyAdmin interface with the database 'codemara_povilas' selected. The 'Structure' tab is active, displaying a list of 14 customer names: Annie Montgomery, Anthony Tucker, Ashley Rose, Barbara Reynolds, Beverly Wood, Bobby Hawkins, Bruce Stanley, Deborah Parker, Denise Larson, Dennis Freeman, Dennis Shaw, Diana Davis, Diane Griffin, and Donna Burns. The 'SQL' tab is also visible.

```
SELECT name, address FROM customers ORDER BY id
```

The screenshot shows the phpMyAdmin interface with the database 'codemara_povilas' selected. The 'Structure' tab is active, displaying a list of 14 customer entries, each containing a name and an address. The results are shown in a table format with columns 'name' and 'address'. The 'SQL' tab shows the executed query: `SELECT name, address FROM customers ORDER BY id LIMIT 0 , 30`. A message indicates that the table does not have a unique column.

name	address
Gerald Foster	994 Transport Plaza
Kevin Hawkins	03231 Dwight Hill
Diana Davis	35 Fairfield Parkway
Heather Gonzalez	14th Trail
Terry Hart	353 Messerschmidt Pass
Nancy Gomez	121 Southridge Lane
Beverly Wood	21 Bluestem Trail

SELECT name, address FROM customers ORDER BY id

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The 'Structure' tab is selected. In the SQL box, the following query is entered:

```
SELECT name, address
FROM customers
ORDER BY id
LIMIT 0 , 30;
```

The results are displayed in a grid:

name	address
Gerald Foster	994 Transport Plaza
Kevin Hawkins	03231 Dwight Hill
Diana Davis	35 Fairfield Parkway
Heather Gonzalez	1 4th Trail
Terry Hart	353 Messerschmidt Pass
Nancy Gomez	121 Southridge Lane
Beverly Wood	21 Bluestem Trail
Fred Roberts	9 Main Crossing
Maria Cox	1 Crest Line Pass

The screenshot shows the phpMyAdmin interface for the same database. The 'Structure' tab is selected. The SQL box contains a modified query:

```
SELECT name, address
FROM customers
ORDER BY state, name
LIMIT 0 , 30;
```

The results are displayed in a grid:

name	address
Ernest Hall	85309 Debs Park
Fred Roberts	9 Main Crossing
Juan Graham	27 Green Ridge Center
Paul Simpson	2 Eggendart Drive
Philip Smith	40 Dakota Point
Diane Griffin	6 Ridgeview Place
Jack Mitchell	50354 Eliot Circle
Janet Bailey	5581 Algoma Terrace
Jerry Dean	141 Del Sol Crossing
Ralph Gutierrez	2 Forest Parkway
Robin Chavez	387 Dayton Plaza
Sara Oliver	050 Lawn Terrace
Terry Hart	353 Messerschmidt Pass
Theresa Richardson	66 Grover Pass
Ernest Morris	4119 Pepper Wood Pass
Lois Hanson	614 North Junction
Elizabeth James	1 John Wall Way
Frances Ramirez	914 Golden Leaf Alley
Beverly Wood	21 Bluestem Trail

9 - Sort Direction

SELECT name,zip FROM customers ORDER BY zip

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. In the 'SQL' tab, the following query is entered:

```
SELECT name, zip  
FROM customers  
ORDER BY zip  
LIMIT 0 , 30
```

The results are displayed in a table:

name	zip
Linda Frazier	01905
Elizabeth James	06152
Frances Ramirez	06905
Russell Wells	10990
Dennis Shaw	10184
Gregory Wheeler	19178
Lisa Hanson	19805
Raphael Kellie	19810

SELECT name,zip FROM customers ORDER BY zip DESC

(sorting hi to low)

The screenshot shows the phpMyAdmin interface for the same database. The 'SQL' tab contains the same query as before:

```
SELECT name, zip  
FROM customers  
ORDER BY zip  
LIMIT 0 , 30
```

The results are displayed in a table:

name	zip
Paula George	99210
Jerry Dean	95838
Sara Oliver	95298
Janet Bailey	95133
Jack Mitchell	94544
Diane Griffin	94142
Robin Chavez	92648
Theresa Richardson	92519
Terry Hart	91520
Ralph Gutierrez	91411
Ashley Rose	89436
Bobby Hawkins	88558
Maria Cox	87190

```
SELECT name,id FROM customers ORDER BY id
```

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The 'customers' table is selected. In the SQL tab, the following query is entered:

```
SELECT name, id  
FROM customers  
ORDER BY id  
LIMIT 0 , 30
```

Below the query, there are dropdown menus for 'Show all' and 'Start row: 30', and a 'Number of rows: 30' input field. The results are displayed in a grid:

name	id
Gerald Foster	1
Kevin Hawkins	2
Diana Davis	3
Heather Gonzalez	4
Terry Hart	5
Nancy Gomez	6
Beverly Wood	7
Eduardo	8
Frances Ramirez	96

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

The screenshot shows the phpMyAdmin interface for the same database and table. The 'customers' table is selected. In the SQL tab, the following query is entered:

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

A warning message is displayed: "This table does not contain a unique column. Grid edit, checkbox, E". Below the query, a green success message says: "Showing rows 0 - 0 (1 total, Query took 0.0004 sec) [id: 96 - 96]". The results are displayed in a grid:

name	id
Frances Ramirez	96

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

The screenshot shows the phpMyAdmin interface on a web browser. The URL is `xo6.x10hosting.com:2083/cpsess9446659933/`. The left sidebar lists databases: `codemara_povilas` and `information_schema`. The main area shows a query result for the `customers` table. The query was:

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

The result table has one row:

name
Frances Ramirez

A green message bar at the top right says: `Showing rows 0 - 0 (1 total, Query took 0.0002 sec)`.

10 - Basic Data Filtering and WHERE

SELECT id, name FROM customers WHERE id=54

+ Options	
id	name
54	Russell Wells

SELECT id, name FROM customers WHERE id !=54

```
SELECT id, name  
FROM customers  
WHERE id !=54  
LIMIT 0 , 30
```

1 ▾ Show all > >> | Show : Star

+ Options	
id	name
1	Gerald Foster
2	Kevin Hawkins
3	Diana Davis
4	Heather Gonzalez
5	Terry Hart
6	Nancy Gomez
7	Beverly Wood
8	Fred Roberts
9	Maria Cox
10	Evelyn Warren
11	Shirley Vasquez
12	Maria Fuller
13	Deborah Parker
14	Linda Frazier
15	Randy Ray

SELECT id, name FROM customers WHERE id <8

+ Options	
id	name
1	Gerald Foster
2	Kevin Hawkins
3	Diana Davis
4	Heather Gonzalez
5	Terry Hart
6	Nancy Gomez
7	Beverly Wood

SELECT id, name FROM customers WHERE id BETWEEN 25 AND 30

```
SELECT id, name
FROM customers
WHERE id
BETWEEN 25
AND 30
LIMIT 0 , 30
```

Show : Start row: 0

+ Options

id	name
25	Jack Mitchell
26	Terry Hicks
27	Ashley Rose
28	Elizabeth James
29	Todd Edwards
30	Rachel Kelley

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

```
SELECT name, state
FROM customers
WHERE state = 'CA'
LIMIT 0 , 30
```

Show : Start row: 0

+ Options

name	state
Terry Hart	CA
Jack Mitchell	CA
Janet Bailey	CA
Theresa Richardson	CA
Sara Oliver	CA
Robin Chavez	CA
Diane Griffin	CA
Ralph Gutierrez	CA
Jerry Dean	CA

11 Advanced Filtering Using AND and OR

SELECT name, state, city FROM customers WHERE state='MA' AND city='Lynn'

```
SELECT name, state, city  
FROM customers  
WHERE state = 'MA'  
AND city = 'Lynn'  
LIMIT 0 , 30
```

Show : Start row:

+ Options

name	state	city
Linda Frazier	MA	Lynn

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

```
SELECT name, state, city  
FROM customers  
WHERE state = 'FL'  
OR city = 'Tulsa'  
LIMIT 0 , 30
```

Show : Start row: Number of rows

+ Options

name	state	city
Gerald Foster	FL	Ocala
Randy Ray	OK	Tulsa
Julie Dixon	OK	Tulsa
Annie Montgomery	FL	Saint Petersburg
Kevin Hernandez	FL	Miami
Robin Duncan	FL	Hialeah
Steve Turner	FL	Sarasota
Earl Payne	FL	Tallahassee

```
SELECT name, id, city FROM customers WHERE (id=1 OR id=2) AND city='Atlanta'
```

SELECT name, id, city
FROM customers
WHERE (
 id =1
 OR id =2
)
AND city = 'Atlanta'
LIMIT 0 , 30

Show : Start row: 0

+ Options

name	id	city
Kevin Hawkins	2	Atlanta

12 - Are you IN or are you NOT IN

```
SELECT name,state FROM customers WHERE state='FL' OR state=" " OR state='NM' OR state='ND'
```

SELECT name, state
FROM customers
WHERE state = 'FL'
 OR state = "
 OR state = 'NM'
 OR state = 'ND'
LIMIT 0 , 30

Show : Start row: 0 Number

+ Options

name	state
Gerald Foster	FL
Diana Davis	ND
Heather Gonzalez	NM
Maria Cox	NM
Annie Montgomery	FL
Kevin Hernandez	FL
Robin Duncan	FL
Steve Turner	FL
Earl Payne	FL

Show : Start row: 0 Number

Query results operations

Print view Print view (with full !)

```
SELECT name,state FROM customers WHERE state IN('CA','NM','ND')
```

Show : Start row:

+ Options

name	state
Diana Davis	ND
Heather Gonzalez	NM
Terry Hart	CA
Maria Cox	NM
Jack Mitchell	CA
Janet Bailey	CA
Theresa Richardson	CA
Sara Oliver	CA
Robin Chavez	CA

```
SELECT name,state FROM customers WHERE state IN('CA','NM','ND') ORDER BY state
```

```
SELECT name, state
FROM customers
WHERE state
IN (
    'CA', 'NM', 'ND'
)
ORDER BY state
LIMIT 0 , 30
```

Show : Start row:

+ Options

name	state
Terry Hart	CA
Jack Mitchell	CA
Janet Bailey	CA
Theresa Richardson	CA
Sara Oliver	CA
Robin Chavez	CA
Diane Griffin	CA
Ralph Gutierrez	CA
Jerry Dean	CA
Diana Davis	ND
Heather Gonzalez	NM
Maria Cox	NM

```
SELECT name,state FROM customers WHERE state NOT IN('CA','NM','ND') ORDER BY state
```

name	state
Ernest Hall	AL
Fred Roberts	AZ
Juan Graham	AZ
Paul Simpson	AZ
Philip Smith	AZ
Lois Hanson	CO
Ernest Morris	CO
Elizabeth James	CT
Frances Ramirez	CT
Beverly Wood	DC
Dorothy Wright	DC
Russell Powell	DC
Russell Bowman	DC
Lisa Hanson	DE
Rachel Kelley	DE
Joseph Cole	DE
Gerald Foster	FL
Annie Montgomery	FL
Kevin Hernandez	FL
Robin Duncan	FL
Steve Turner	FL
Earl Payne	FL
Kevin Hawkins	GA
Nancy Gomez	GA
Jeremy Nelson	IA
Sarah Sims	IL
Jesse Riley	IL
Denise Larson	IL
Mark Brooks	IL
Donna Burns	IN

13 - How Search Engines Work

SELECT name FROM items WHERE name LIKE 'new%'

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

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

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

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

Start with h ends with d.

+ Options
city
Hayward
Hartford

14 - More on Wildcards

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

name
3 boxes of frogs
7 boxes of frogs

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

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

15 - Regular Expressions

SELECT name FROM items WHERE name REGEXP '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'

Sort by key:

Options
ame
boxes of frogs
boxes of frogs
3 boxes of frogs

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

name
traditional carpet
gold necklace
used car
gold earring
scarf

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

Same thing as

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

name
3 boxes of frogs

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

Search for anything besides 12345

name
7 boxes of frogs
48 boxes of frogs

16 - Creating Custom Columns

```
SELECT CONCAT(city, ', ', state) FROM customers
```

Concat joins things together

A screenshot of a MySQL query results table. At the top, there is a navigation bar with a dropdown menu set to '1', and buttons for 'Show all', '>', and '>>'. Below this is a section labeled '+ Options' with a 'CONCAT(city, ', ', state)' button. The main area displays a list of concatenated addresses:

Ocala, FL
Atlanta, GA
Fargo, ND
Albuquerque, NM
Burbank, CA
Valdosta, GA
Washington, DC
Gilbert, AZ

```
SELECT CONCAT(city, ', ', state) AS new_address FROM customers
```

A screenshot of a MySQL query results table. At the top, there is a navigation bar with a dropdown menu set to '1', and buttons for 'Show all', '>', and '>>'. Below this is a section labeled '+ Options' with a 'new_address' button. The main area displays a list of concatenated addresses:

Ocala, FL
Atlanta, GA
Fargo, ND
Albuquerque, NM
Burbank, CA
Valdosta, GA
Washington, DC
Gilbert, AZ
Albuquerque, NM
Flint, MI
Kansas City, MO
New Orleans, LA
Toledo, OH
Lynn, MA
Tulsa, OK
Springfield, MO

SELECT name, cost FROM items

name	cost
Brand New iMac Computer	885.06
used diaper from my sister	801.84
Fresh apple pie	988.33
New gym socks	940.37
Weedwacker only slightly used	682.33
New ipad stolen from best buy	546.67
Book about having babies	701.46
Woman jeans	130.95
traditional carpet	227.91
shows	691.11
beer	79.34
box of frogs	77.21

SELECT name, cost, cost -1 FROM items

+ Options

name	cost	cost -1
Brand New iMac Computer	885.06	884.06
used diaper from my sister	801.84	800.84
Fresh apple pie	988.33	987.33
New gym socks	940.37	939.37
Weedwacker only slightly used	682.33	681.33
New ipad stolen from best buy	546.67	545.67
Book about having babies	701.46	700.46
Woman jeans	130.95	129.95
traditional carpet	227.91	226.91
shows	691.11	690.11

SELECT name, cost, cost -1 AS sale_proce FROM items

name	cost	sale_proce
Brand New iMac Computer	885.06	884.06
used diaper from my sister	801.84	800.84
Fresh apple pie	988.33	987.33
New gym socks	940.37	939.37
Weedwacker only slightly used	682.33	681.33
New ipad stolen from best buy	546.67	545.67

17 – Functions

SELECT name, UPPER(name) FROM customers

+ Options

name	UPPER(name)
Gerald Foster	GERALD FOSTER
Kevin Hawkins	KEVIN HAWKINS
Diana Davis	DIANA DAVIS
Heather Gonzalez	HEATHER GONZALEZ
Terry Hart	TERRY HART
Nancy Gomez	NANCY GOMEZ
Beverly Wood	BEVERLY WOOD
Fred Roberts	FRED ROBERTS
Maria Cox	MARIA COX
Evelyn Warren	EVELYN WARREN
Shirley Vasquez	SHIRLEY VASQUEZ
Maria Fuller	MARIA FULLER

SELECT cost, SQRT(cost) FROM items

cost	SQRT(cost)
885.06	29.749957983163604
801.84	28.316779477899672
988.33	31.43771620203987
940.37	30.66545287452967
682.33	26.1214471268343
546.67	23.38097517213514
701.46	26.485090145211892
130.95	11.44333867365639

SELECT AVG(cost) FROM items

+ Options

AVG(cost)

517.1749000000001

```
SELECT SUM(bids) FROM items
```

+ Options

SUM(bids)

31765

18 - More on Aggregate Functions

```
SELECT COUNT(name) FROM items WHERE seller_id=6
```

COUNT(name)

4

```
SELECT AVG(cost) FROM items WHERE seller_id=6
```

+ Options

AVG(cost)

339.0025

```
SELECT COUNT(*) AS item_count,  
MAX(cost) AS max,  
AVG(cost) AS avg  
FROM items WHERE seller_id=12;
```

```
SELECT COUNT( * ) AS item_count, MAX( co  
FROM items  
WHERE seller_id =12
```

+ Options

item_count	max	avg
5	5700.5	1307.126

19 - GROUP BY

```
SELECT seller_id, COUNT(*) AS item_count FROM items WHERE seller_id=1
```

seller_id	item_count
1	2

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

seller_id	item_count
1	2
2	3
3	1
4	1
5	2
6	4
7	1
10	2
11	1
12	5
13	1
14	2
15	2
16	1
17	1

```
SELECT seller_id, COUNT(*) AS item_count FROM items GROUP BY seller_id Having COUNT(*)>=3
```

seller_id	item_count
2	3
6	4
12	5
52	4
86	3

```

SELECT seller_id, COUNT(*) AS item_count FROM items GROUP BY seller_id Having COUNT(*)>=3
ORDER BY
item_count DESC

```

seller_id	item_count	▼
12	5	
6	4	
52	4	
2	3	
86	3	

20 – Subqueries

`SELECT AVG(cost) FROM items`

AVG(cost)

517.1749000000001

`SELECT name, cost FROM items WHERE cost>(517) ORDER BY cost DESC`

· Options	
name	cost
Wikizz	993.06
Fresh apple pie	988.33
Rooxo	976.30
New gym socks	940.37
Twitterbridge	928.97
Twitternation	916.97
Einti	915.96
Yabox	913.18
Eamia	906.61
Brand New iMac Computer	885.06
Oyope	882.47
awesome alien computer	872.39
refrigerator	860.90
Jaxworks	850.49

```
SELECT name, cost FROM items WHERE cost>(
```

```
SELECT AVG(cost) FROM items
```

```
)ORDER BY cost DESC
```

Sort by key: ▾

+ Options

name	cost
Wikizz	993.06
Fresh apple pie	988.33
Rooxo	976.30
New gym socks	940.37
Twitterbridge	928.97
Twittermentation	916.97
Einti	915.96
Yabox	913.18
Eamia	906.61
Brand New iMac Computer	885.06
Oyope	882.47
awesome alien computer	872.39
refrigerator	860.90
Jaxworks	850.49
Gabtype	837.82
Minyx	834.23
newspaper	813.85
Jayo	807.12
used diaper from my sister	801.84
Skynoodle	799.71
Quatz	799.10
Thoughtbridge	759.87
Reallinks	756.04
Yabox	748.64
Zoozzy	746.41
Yambee	733.56
Youopia	725.61
Wikizz	714.29

21 - Another Subquery Example

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

The screenshot shows the phpMyAdmin interface. In the left sidebar, there's a tree view with 'codemara_povilas' expanded, showing 'New' and 'customers'. The main area has a title bar 'Server: localhost > Database: codemara_povilas > Table: items'. Below it are tabs: Browse, Structure, SQL, Search, Insert, Export, Import. A search bar says 'Run SQL query/queries on database codemara_povilas:'. The SQL query is: '1 SELECT seller_id FROM items WHERE name LIKE "%boxes of frogs%"'. The results pane shows a table with one column 'seller_id' containing three rows: 6, 6, and 68.

+ Options

seller_id

6

6

68

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

AND seller_id IN(68,6 ,6)

```
SELECT name, MIN( cost )
FROM items
WHERE name LIKE '%boxes of frogs'
AND seller_id
IN ( 68, 6, 6 )
```

Show : Start row: 0 Number of rows

+ Options

name	MIN(cost)
3 boxes of frogs	30.48

```

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'
)

```

**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'**

Show : Start row: 0 Number of rows:

+ Options

name	MIN(cost)
3 boxes of frogs	30.48

22 - How to Join Tables

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

```
SELECT customers.id, customers.name, items.name, items.cost
```

```
FROM customers,items
```

```
WHERE customers.id=seller_id
```

```
ORDER by customers.id
```

23 - Outer Joins

```
SELECT i.seller_id, i.name, c.id  
FROM customers AS c, items AS i  
WHERE i.seller_id=c.id
```

seller_id	name	id
28	Brand New iMac Computer	28
28	used diaper from my sister	28
34	Fresh apple pie	34
41	New gym socks	41
65	Weedwacker only slightly used	65
78	New ipad stolen from best buy	78
78	Book about having babies	78
23	Woman jeans	23
52	traditional carpet	52

```
SELECT customers.name, items.name FROM customers, items WHERE  
customers.id=seller_id
```

```
SELECT customers.name, items.name  
FROM customers, items  
WHERE customers.id = seller_id  
LIMIT 0 , 30
```

1 ▾ Show all > >> | Show : Start row:

Options

name	name
Zabeth James	Brand New iMac Computer
Zabeth James	used diaper from my sister
Maria Burns	Fresh apple pie
ura Cooper	New gym socks
neth Stewart	Weedwacker only slightly used
elyn Morrison	New ipad stolen from best buy
elyn Morrison	Book about having babies
thy Gonzales	Woman jeans
holas Patterson	traditional carpet
ssell Powell	shows
erald Foster	beer
an Gilbert	box of frogs
vin Hawkins	laptop
ula George	bed
test Hall	thumbelina
resa Morgan	stove
ncy Gomez	shampoing

```
SELECT customers.name, items.name FROM customers LEFT OUTER JOIN  
items ON customers.id=seller_id
```

name	name
Elizabeth James	Brand New iMac Computer
Elizabeth James	used diaper from my sister
Donna Burns	Fresh apple pie
Laura Cooper	New gym socks
Kenneth Stewart	Weedwacker only slightly used
Evelyn Morrison	New ipad stolen from best buy
Evelyn Morrison	Book about having babies
Kathy Gonzales	Woman jeans
Nicholas Patterson	traditional carpet
Russell Powell	shows
Gerald Foster	beer
Ryan Gilbert	box of frogs
Kevin Hawkins	laptop
Paula George	bed
Ernest Hall	thumbelina
Teresa Morgan	stove

24 – UNION

```
SELECT name, cost, bids FROM items WHERE bids > 190
```

UNION

```
SELECT name, cost, bids FROM items WHERE bids > 1000
```

name	cost	bids
used diaper from my sister	801.84	471
Fresh apple pie	988.33	487
New gym socks	940.37	495
Weedwacker only slightly used	682.33	403
New ipad stolen from best buy	546.67	277
Book about having babies	701.46	556
Woman jeans	130.95	279
traditional carpet	227.91	394
shows	691.11	295
beer	79.34	278

25 - Full-Text Searching

```
SELECT name, cost FROM items WHERE Match(name) Against('baby')
```

The screenshot shows the phpMyAdmin interface with the following details:

- SQL Query:**

```
SELECT name, cost
FROM items
WHERE MATCH (
    name
)
AGAINST (
    'baby'
)
```
- Show Options:** Start row: 0
- Sort by key:** None
- + Options:** A table showing the results of the search:

name	cost
baby coat	620.63
baby seat	206.88
baby soap	1.49
baby bottle	125.36

```
SELECT name, cost FROM items WHERE Match(name) Against('+baby -coat'
```

IN BOOLEAN MODE)

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** localhost | **Database:** codemara_povilas
- Browse** | **Structure** | **SQL**
- Show query box:**
 - This table does not contain a unique column
 - Showing rows 0 - 2 (3 total, Query took 0.)
- SQL Query:**

```
SELECT name, cost
FROM items
WHERE MATCH (
    name
)
AGAINST (
    '+baby -coat'
)
IN BOOLEAN
```
- Show Options:** Start row: 0 | Number of rows: 3
- Sort by key:** None
- + Options:** A table showing the results of the search:

name	cost
baby seat	206.88
baby soap	1.49
baby bottle	125.36

26 - INSERT INTO

```
INSERT INTO items VALUES('101','bacon strips','9.95','1','0')
```

The screenshot shows the phpMyAdmin interface. On the left, the database structure for 'codemara_povilas' is visible, including tables 'New', 'customers', and 'items'. The main area is titled 'Server: localhost > Database: codemara_povilas > Table: items'. The 'SQL' tab is selected. A query window contains the SQL command:

```
1 INSERT INTO items VALUES('101','bacon strips','9.95','1','0')
```

```
INSERT INTO items (id, name, cost, seller_id, bids) VALUES('102',
'fish and chips', '7.99', '1', '0')
```

The screenshot shows the phpMyAdmin interface after the insertion. The message '1 row inserted. (Query took 0.0012 sec)' is displayed in a green box. Below it, the SQL query is shown again:

```
1 INSERT INTO items( id, name, cost, seller_id, bids )
VALUES (
'102', 'fish and chips', '7.99', '1', '0'
)
```

The screenshot shows the results of a 'SELECT * FROM items' query. The results are displayed in a table format:

	id	name	cost	seller_id	bids
<input type="checkbox"/>	101	bacon strips	9.95	1	0
<input type="checkbox"/>	102	fish and chips	7.99	1	0
<input type="checkbox"/>	11	beer	79.34	1	278
<input type="checkbox"/>	20	computer	67.70	1	77
<input type="checkbox"/>	76	JabberType	147.49	2	366
<input type="checkbox"/>	13	laptop	493.39	2	126
<input type="checkbox"/>	94	Oyope	882.47	2	194
<input type="checkbox"/>	84	Topicware	549.39	3	114

```
INSERT INTO items (id, cost, name) VALUES
('104', 8.88, 'baby potatos')
```

SELECT *						
FROM `items`						
LIMIT 0 , 30						
1	Show all	>	>>	Show : Start row:	30	Number of rows: 30
Sort by key:	None					
+ Options						
← T →	▼ id	name	cost	seller_id	▲	bids
<input type="checkbox"/>	Edit Copy Delete	103	12.99	beef stake	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	20	computer	67.70	1	77
<input type="checkbox"/>	Edit Copy Delete	101	bacon strips	9.95	1	0
<input type="checkbox"/>	Edit Copy Delete	102	fisn and chips	7.99	1	0
<input type="checkbox"/>	Edit Copy Delete	11	beer	79.34	1	278
<input type="checkbox"/>	Edit Copy Delete	13	laptop	493.39	2	126
<input type="checkbox"/>	Edit Copy Delete	76	Jabbertype	147.49	2	366
<input type="checkbox"/>	Edit Copy Delete	94	Oyope	882.47	2	194

```
INSERT INTO items (id, cost, name) VALUES
('104', 8.88, 'baby potatos')
```

SELECT *						
FROM `items`						
LIMIT 0 , 30						
1	Show all	>	>>	Show : Start row:	30	Number of rows: 30
Sort by key:	None					
+ Options						
← T →	▼ id	name	cost	seller_id	▲	bids
<input type="checkbox"/>	Edit Copy Delete	104	baby potatos	8.88	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	103	beef stake	12.99	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	20	computer	67.70	1	77

27 - How to Insert Multiple Rows

```
INSERT INTO items(id,name,cost,seller_id,bids)VALUES
```

```
('105','beef chops','7.69','1','0'),
```

```
('106','jelly pockets','4.50','1','0'),
```

```
('107','sack of ham','9.95','1','0')
```

The screenshot shows a MySQL Workbench interface. At the top, there is a query editor window containing the following SQL code:

```
SELECT *
FROM `items`
ORDER BY `items`.`id` DESC
LIMIT 0 , 30
```

Below the query editor is a toolbar with buttons for "Show all", "Show : Start row:", and "Number of rows:". The "Start row:" field is set to 30. To the right of the toolbar is a "Sort by key:" dropdown set to "None".

Below the toolbar is a table grid displaying the results of the query. The table has columns: id, name, cost, seller_id, and bids. The data is as follows:

	id	name	cost	seller_id	bids
<input type="checkbox"/>	107	sack of ham	9.95	1	0
<input type="checkbox"/>	106	jelly pockets	4.50	1	0
<input type="checkbox"/>	105	beef chops	7.69	1	0
<input type="checkbox"/>	104	baby potatos	8.88	NULL	NULL
<input type="checkbox"/>	103	12.99	beef stake	NULL	NULL
<input type="checkbox"/>	102	fish and chips	7.99	1	0
<input type="checkbox"/>	101	bacon strips	9.95	1	0
<input type="checkbox"/>	100	Jaxworks	850.49	96	183
<input type="checkbox"/>	99	Topicblab	499.75	50	247
<input type="checkbox"/>	98	Ainyx	297.66	52	283
<input type="checkbox"/>	97	Flipbug	320.47	89	491
<input type="checkbox"/>	96	Muxo	580.62	4	183

```
INSERT INTO items(id,name,cost,seller_id,bids) SELECT id,name,cost,seller_id,bids FROM faketable
```

28 - UPDATE & DELETE

UPDATE items SET name='puddinghammock' WHERE id=103

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```
SELECT *  
FROM `items`  
LIMIT 0 , 30
```

Below the query editor, there are navigation buttons: '1 <' (previous page), 'Show all', '>', '>>', 'Show : Start row: 30', and 'Number of rows: 30'. A dropdown menu 'Sort by key: None' is also present.

The main area displays a table with the following data:

	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	id	name	cost	seller_id	bids
<input type="checkbox"/>	Edit	Copy	Delete	107	sack of ham	9.95	1	0
<input type="checkbox"/>	Edit	Copy	Delete	106	jelly pockets	4.50	1	0
<input type="checkbox"/>	Edit	Copy	Delete	105	beef chops	7.69	1	0
<input type="checkbox"/>	Edit	Copy	Delete	104	baby potatos	8.88	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	103	puddinghammock	beef stake	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	102	fish and chips	7.99	1	0
<input type="checkbox"/>	Edit	Copy	Delete	101	bacon strips	9.95	1	0
<input type="checkbox"/>	Edit	Copy	Delete	100	Jaxworks	850.49	96	183

UPDATE items SET name='chicken paste',bids=66 WHERE id=104

The screenshot shows the MySQL Workbench interface with a table containing the following data:

	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	id	name	cost	seller_id	bids
<input type="checkbox"/>	Edit	Copy	Delete	107	sack of ham	9.95	1	0
<input type="checkbox"/>	Edit	Copy	Delete	106	jelly pockets	4.50	1	0
<input type="checkbox"/>	Edit	Copy	Delete	105	beef chops	7.69	1	0
<input type="checkbox"/>	Edit	Copy	Delete	104	chicken paste	8.88	NULL	66

DELETE FROM items WHERE id=106

The screenshot shows the MySQL Workbench interface with a table containing the following data:

	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	id	name	cost	seller_id	bids
<input type="checkbox"/>	Edit	Copy	Delete	107	sack of ham	9.95	1	0
<input type="checkbox"/>	Edit	Copy	Delete	105	beef chops	7.69	1	0
<input type="checkbox"/>	Edit	Copy	Delete	104	chicken paste	8.88	NULL	66
<input type="checkbox"/>	Edit	Copy	Delete	103	frog paste	beef stake	NULL	66
<input type="checkbox"/>	Edit	Copy	Delete	102	fish and chips	7.99	1	0
<input type="checkbox"/>	Edit	Copy	Delete	101	bacon strips	9.95	1	0
<input type="checkbox"/>	Edit	Copy	Delete	100	Jaxworks	850.49	96	183

29 - CREATE TABLE

```
CREATE TABLE users(
```

```
    id int,
```

```
    username varchar(30),
```

```
    password varchar(20),
```

```
    PRIMARY KEY(id)
```

```
)
```

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The 'Structure' tab is selected for the 'users' table. The table has three columns: 'id', 'username', and 'password'. The 'id' column is defined as int(11) with a primary key constraint. The 'username' and 'password' columns are defined as varchar(30) with collation latin1_swedish_ci.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	0		Change Drop Primary U
2	username	varchar(30)	latin1_swedish_ci		Yes	NULL		Change Drop Primary U
3	password	varchar(20)	latin1_swedish_ci		Yes	NULL		Change Drop Primary U

30 - NOT NULL & AUTO INCREMENT

username varchar(30) NOT NULL

id int NOT NULL AUTO_INCREMENT

automatically add 1 to the last id

```
CREATE TABLE bacon(
```

```
    id int NOT NULL AUTO_INCREMENT,
```

```
    username varchar(30) NOT NULL,
```

```
    password varchar(20) NOT NULL,
```

```
    PRIMARY KEY(id)
```

```
)
```

```
)
```

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. On the left, the database structure is visible with tables 'acon', 'bacon', 'customers', 'items', and 'users' under the 'codemara_povilas' schema. The 'bacon' table is selected. The main panel displays the 'Structure' tab for the 'bacon' table. The table has three columns: 'id' (int(11), primary key, auto-increment), 'username' (varchar(30)), and 'password' (varchar(20)). Below the table structure, there is an 'Information' section providing details about space usage and row statistics. The 'Space usage' section shows Data at 0 B, Index at 1 KiB, and Total at 1 KiB. The 'Row statistics' section shows Format as dynamic, Collation as latin1_swedish_ci, Rows as 0, and Next autoindex as 1. It also includes Creation and Last update dates.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)	latin1_swedish_ci		No	None	AUTO_INCREMENT	Change Drop Primary
2	username	varchar(30)	latin1_swedish_ci		No	None		Change Drop Primary
3	password	varchar(20)	latin1_swedish_ci		No	None		Change Drop Primary

Space usage		Row statistics	
Data	0 B	Format	dynamic
Index	1 KiB	Collation	latin1_swedish_ci
Total	1 KiB	Rows	0
		Next autoindex	1
		Creation	Jun 06, 2016 at 05:22 AM
		Last update	Jun 06, 2016 at 05:22 AM

phpMyAdmin

Databases - x10Hosting | xo6.x10hosting.com / local | xo6.x10hosting.com / lo | +

Server: localhost | Database: codemara_povilas | Table: bacon

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#)

Column Type Function Null Value

id	int(11)		
username	varchar(30)		povilas
password	varchar(20)		password

[Go](#)

Ignore

Column Type Function Null Value

id	int(11)		
username	varchar(30)		saly
password	varchar(20)		chicken

[Go](#)

[Insert as new row](#) and then [Go back to previous page](#)

[Go](#) [Reset](#)

Continue insertion with [2](#) rows

phpMyAdmin

Databases - x10Hosting | xo6.x10hosting.com / local | xo6.x10hosting.com / lo | +

Server: localhost | Database: codemara_povilas | Table: bacon

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Operations](#) [Triggers](#)

2 rows inserted.
Inserted row id: 2

```
INSERT INTO `codemara_povilas`.`bacon` (
  `id`,
  `username`,
  `password`
)
VALUES (
  NULL,
  'povilas',
  'password'
),
(
  NULL,
  'saly',
  'chicken'
);
```

Run SQL query/queries on database codemara_povilas: [?](#)

```
1 INSERT INTO `codemara_povilas`.`bacon` ('id', `username`, `password`) VALUES (NULL, 'povilas', 'password'), (NULL, 'saly', 'chicken');
```

[SELECT](#) [SELECT](#) [INSERT](#) [UPDATE](#) [DELETE](#) [Clear](#)

Structure

```

SELECT *
FROM `bacon`
LIMIT 0 , 30

```

	id	username	password
<input type="checkbox"/>	1	povilas	password
<input type="checkbox"/>	2	saly	chicken

Query results operations

31 - ALTER DROP RENAME TABLE

ALTER TABLE bacon ADD samplecolumn varchar(10)

Structure

```

SELECT *
FROM `bacon`
LIMIT 0 , 30

```

	id	username	password	samplecolumn
<input type="checkbox"/>	1	povilas	password	NULL
<input type="checkbox"/>	2	saly	chicken	NULL

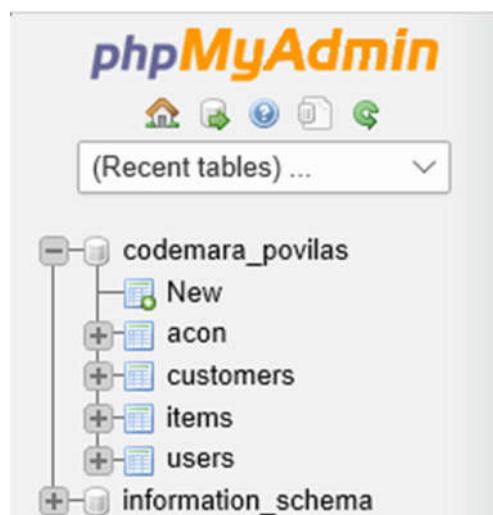
ALTER TABLE bacon DROP COLUMN samplecolumn

Delete column

Structure

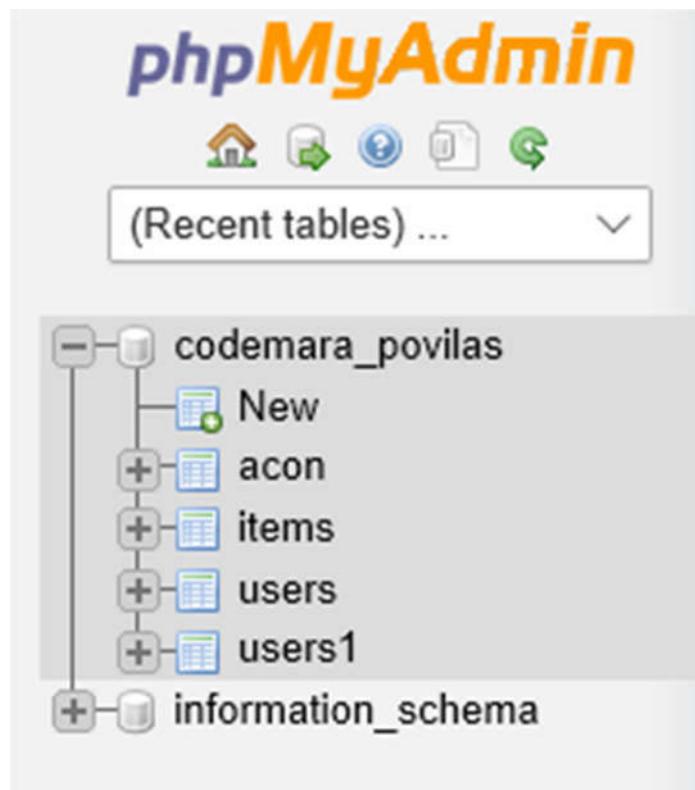
```
DROP TABLE bacon
```

Delete table



RENAME TABLE

```
RENAME TABLE customers TO users1
```



32 - View

They like temporary table

```
SELECT id, name, bids FROM items ORDER BY bids DESC LIMIT 10
```

The screenshot shows the phpMyAdmin interface for a database named 'codemara_povilas'. The left sidebar lists tables: 'New', 'acon', 'items', 'users', 'users1', and 'information_schema'. The 'items' table is selected. The top navigation bar includes tabs for Structure, SQL, Search, Query, Export, and Import.

The 'Query' tab is active, displaying the executed SQL query:

```
SELECT id, name, bids
FROM items
ORDER BY bids DESC
LIMIT 10
```

The results are shown in a table below:

	id	name	bids
<input type="checkbox"/>	39	air conditioner	686
<input type="checkbox"/>	86	Gabtype	660
<input type="checkbox"/>	32	48 boxes of frogs	655
<input type="checkbox"/>	30	gold necklace	651
<input type="checkbox"/>	35	scarf	643
<input type="checkbox"/>	61	Zoombeat	640
<input type="checkbox"/>	70	Shufflester	630
<input type="checkbox"/>	26	baby coat	619
<input type="checkbox"/>	16	stove	596
<input type="checkbox"/>	44	Jayo	582

Below the table are buttons for 'Check All', 'With selected:', 'Check', 'Delete', and 'Export'.

```
CREATE VIEW mostbids AS  
SELECT id,name,bids FROM items ORDER BY bids DESC LIMIT 10
```

The screenshot shows the MySQL Workbench interface. On the left, the database schema tree displays the 'codemara_povilas' database, which contains the 'Tables' and 'Views' nodes. A new view named 'mostbids' is currently selected. On the right, the main workspace shows the creation of the view with the following SQL query:

```
SELECT *  
FROM `mostbids`  
LIMIT 0 , 30
```

Below the query, there is a 'Show' button and a 'Start row:' input field set to 0. The results pane displays the data from the 'mostbids' view:

id	name	bids
39	air conditioner	686
86	Gabtype	660
32	48 boxes of frogs	655
30	gold necklace	651
35	scarf	643
61	Zoombeat	640
70	Shufflester	630
26	baby coat	619
16	stove	596
44	Jayo	582

```
SELECT Concat(city,',',state) AS address from users1
```

The screenshot shows the phpMyAdmin interface for the 'codemara_povilas' database. The left sidebar lists tables like 'acon', 'items', 'users', and 'users1'. The 'SQL' tab contains the query:

```
SELECT CONCAT( city, ',', state ) AS address
FROM users1
LIMIT 0 , 30
```

The results pane shows the output:

address
Ocala,FL
Atlanta,GA
Fargo,ND
Albuquerque,NM
Burbank,CA
Valdosta,GA

```
CREATE VIEW mailing AS
```

```
SELECT Concat(city,',',state) AS address from users1
```

The screenshot shows the phpMyAdmin interface for the 'codemara_povilas' database. The left sidebar lists tables like 'acon', 'items', 'users', and 'users1', and a view named 'mailing'. The 'SQL' tab contains the query:

```
SELECT *
FROM `mailing`
LIMIT 0 , 30
```

The results pane shows the output:

address
Ocala,FL
Atlanta,GA
Fargo,ND
Albuquerque,NM
Burbank,CA
Valdosta,GA
Washington,DC
Gilbert,AZ
Albuquerque,NM
Flint,MI
Kansas City,MO
New Orleans,LA
Toledo,OH
Lynn,MA
Tulsa,OK
Springfield,MO

```
SELECT name ,bids FROM mostbids
```

The screenshot shows the MySQL Workbench interface. On the left, the database schema is visible with the 'Tables' and 'Views' sections expanded. The 'mostbids' view is selected. The main pane displays the results of the executed SQL query:

```
Showing rows 0 - ... (30 total)
+-----+-----+
| name | bids |
+-----+-----+
| air conditioner | 686 |
| Gabtype | 660 |
| 48 boxes of frogs | 655 |
| gold necklace | 651 |
| scarf | 643 |
| Zoombeat | 640 |
| Shufflester | 630 |
| baby coat | 619 |
| stove | 596 |
| Jayo | 582 |
+-----+-----+
```

The results are presented in a table with two columns: 'name' and 'bids'. The 'name' column lists various items, and the 'bids' column shows their respective bid counts.

33 - Final

Save database

The screenshot shows the phpMyAdmin interface for the database 'codemara_povilas'. On the left, there is a tree view of the database structure under the 'Tables' and 'Views' sections. The 'Tables' section contains 'New', 'acon', 'items', 'users', and 'users1'. The 'Views' section contains 'New', 'mailing', and 'mostbids'. At the top right, there are tabs for Structure, SQL, Search, Query, Export, Import, and Insert. Below the tabs, the 'Export' tab is selected. The main area displays the 'Export Method' section with two radio buttons: 'Quick - display only the minimal options' (selected) and 'Custom - display all possible options'. Below this is the 'Format' section with a dropdown menu set to 'SQL'.

Never use SELECT ALL

Never get more data when u actually need

FULLTEXT > LIKE