# LESSON H2_EN. CREATING AND USING A MYSQL DATABASE FOR eBUSINESS/eCOMMERCE WEBSITE.

PARENT ENTITY: S.C.FOX I.E. s.r.l. , Bucharest, 4 Aurel Botea street, bl.B8, ground floor, ap.4, sector 3
Authors: Constantin Chersin, software engineer, FOX I.E. s.r.l. , Bucharest, 4 Aurel Botea street, bl.B8, ground floor, ap.4, sector 3
Mihaela Chersin, Educational Coordinator, FOX I.E. s.r.l. , Bucharest, 4 Aurel Botea street, bl.B8, ground floor, ap.4, sector 3
Call number: +40-21-3232164, Office hour: Wednesday 10 a.m. – 11 a.m.

***After the learning this lesson you will be richer with the following knowledge:***
- How you create and use a database.

**CONTENT OF THE LESSON**
1. Creating a MySQL database for eBusiness / eCommerce website.
2. How to drop and restore a database
3. How to drop a table and a field

**LEARNING OBJECTIVES:**
**After learning this lesson you will accomplish the ability :**
- to create and delete database
- to create, modify, add and delete tables and fields in a database

## 1. Creating a MySQL database for eBusiness / eCommerce website

### 1.1. Setting up a database

The best way to understand what a database is, is to make an example of setting up and administrate a database.

Your database will be created using a special application provided with Apache-MySQL-PHP (XAMPP) installation. This application is named phpMyAdmin and it is a PHP scripts package. PhpMyAdmin helps you to manage databases. It allows you to create and access databases and to create, modify, add and delete tables and fields in your databases. It will also allow you to backup and restore databases, plus perform simple and complex queries. PhpMyAdmin is the most web-based popular interface for MySQL database administration. It looks like a web site and you can to access it from the URL http://localhost/xampp/, (see fig. 1.1.)

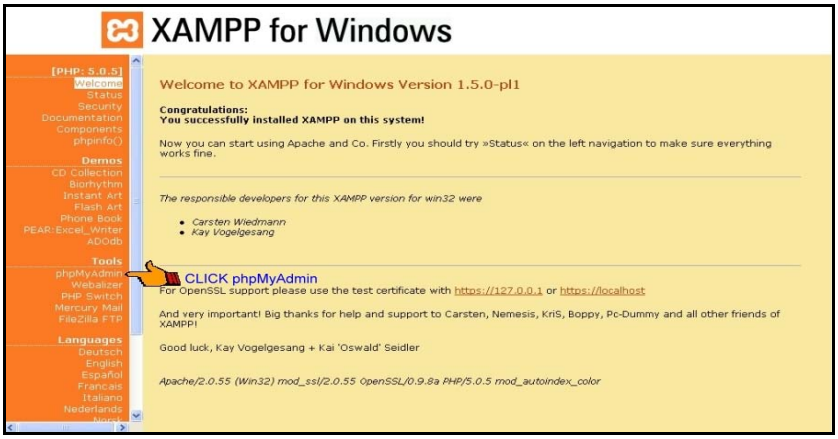

Fig.1.1. Screenshot of XAMPP for Windows

Click on phpMyAdmin (see fig. 1.1.) and a new window will be open (see fig. 1.2.) (otherwise, you can go directly to the URL: http://localhost/phpmyadmin/)

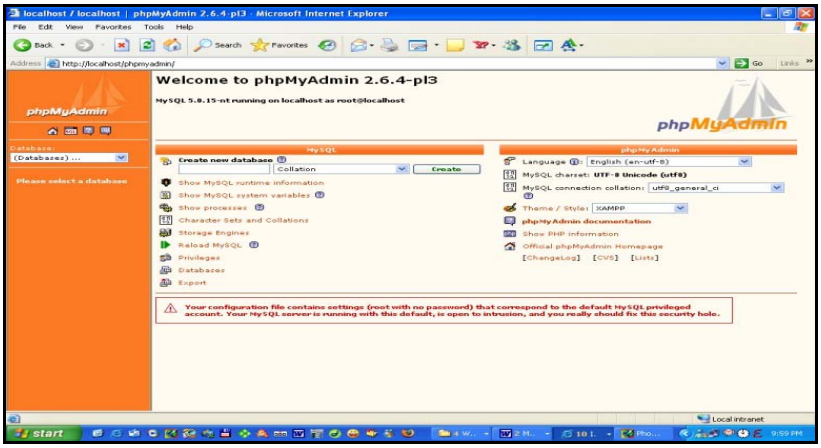


Fig. 1.2. PhpMyAdmin 2.6.4-pl3 interface

*Note:* "PhpMyAdmin 2.6.4-pl3" *can be different at time when you try to download. From time to time, a new version is available. Choose the most recent.*

With this interface you will be able to manage all your future databases.
In working with MySQL you will need a user name and a password. By default our installation has a user named root and no password. Usually working as root without password with MySQL is a security issue, but as we are working locally (not visibility on the internet) this will not be a problem. Later when your web site will be visible on the Internet you will have to setup a password.

To create a database you must know that it is made from:

- name of database
- tables with fields (ordered in rows and columns)
- records (ordered in rows and columns)

Our proposal is to create a database. This represents your first step in building an eCommerce website, named eStore.

In real world for a store you need a dossier with sheets having rows with informations about your products, orders and customers. You can consider that are no differences in virtual world. In your database the dossier becomes a database, sheets are tables and rows are fields.

Let's start to create the database. Let's name your database "estore_db". To create it, use the URL: "http://localhost/phpmyadmin/) (fig. 1.2.) and write estore_db along of "Create new database" field. Click on Create button (see fig.1.3.).
The result, that the "estore_db" database has been created (fig. 1.4.)


Fig. 1.3. Screenshot showing how to create "estore_db" database


Fig. 1.4. SQL query to create "estore_db" database

Now you have a database. You can start to organize it into tables.

For your "estore_db" database, you need four tables:

- "**members**" - table with informations about members(customers)
- "**products**" - table with informations about products
- "**orders**" - table with informations about customer's orders
- "**orders items**" - table with informations about ordered items (products)

In addition, for each table you need fields.

On the first table, named "**members**", you will have 5 fields:

- **m_id** - member id (the id field is important because it may be the only field that is unique to each table entry and also keep track of any record).
- **username** - user name of the member

- **password** - password of the member
- **email** - email of the member
- **privilege** - members may be the customers or administrator of the eStore.

On the second table, named "**products**", you will have 3 fields:

- **id** - product id
- **product** - name of the product
- **price** - price of the product

On the third table, named "**orders**", you will have 4 fields:

- **o_id** - order id
- **username** - user name of the member (customer )
- **date** - date when the customer purchased products
- **comments** – comments or instructions of the customer

On the fourth table, named "**orders items**", you will have 4 fields:

- **o_id** - order id
- **product** - name of the product
- **qty** - quantity of products ordered by the customer
- **price -** price of the product

Now after you have organized your database you are invited to fill your already created "estore_db".

- Create the "**members**" table with 5 fields

Fill the "Name" box with the name of your table: **members,** the "Number of fields" box with number 5, and then click "Go", (fig. 1.5.):


Fig. 1.5. Screenshot showing how to create "**members**" table

- Fill out the form fields of "**members**" table (fig.1.6)

**m_id -** is a integer number and because of this you can use an integer type of data, named "INT"**,** which is a normal-size integer in MySQL. Length/Values allowed for an "INTEGER" is 11. The m_id must be "not null" meaning the value of the column cannot be null. We made m_id "Primary key" for this table, used as a unique identifier. This means that 2 different m_ids can have the same value. Because of this m_id has been set to be "auto_increment". Auto_increment makes that each time a new entry is added, the next value will be incremented by 1. This will help you to keep your records in an organized way.
So, you must fill out the **m_id** field, with:

Type: INT Length/Values: 11 Null: not null Extra: auto_increment select Primary

**username -** is a text and because of this you can use a text type of data, named "VARCHAR" meaning variable character ("character" because you can store characters (letters, numbers, and so on) and "variable" because you can store varied amount of characters). Length 25 allowed for a "VARCHAR" is adequate.
So, you must fill out the **username** field, with:

Type: VARCHAR Length/Values: 25 Null: not null

**password -** is a text and because of this you can use "VARCHAR". Length 32 allowed for a "VARCHAR" is adequate.
So, you must fill out the **password** field, with:

Type: VARCHAR Length/Values: 32 Null: not null

**Email -** is a text and because of this you can use "VARCHAR" with length 255.
So, you must fill out the **email** field, with:

Type: VARCHAR Length/Values: 255 Null: not null

**privilege –** is a very small number and because of this you can use "TINYINT" which is a very small integer. The "Default" indicates a default value for a column. The default value must be a constant.

So, you must fill out the **privilege** field, with:

Type: TINYINT  Length/Values: 4  Null: not null  Default: 0

Fill out the fields like in fig. 1.6. and after then, click "Save" (do not click "Go" button).

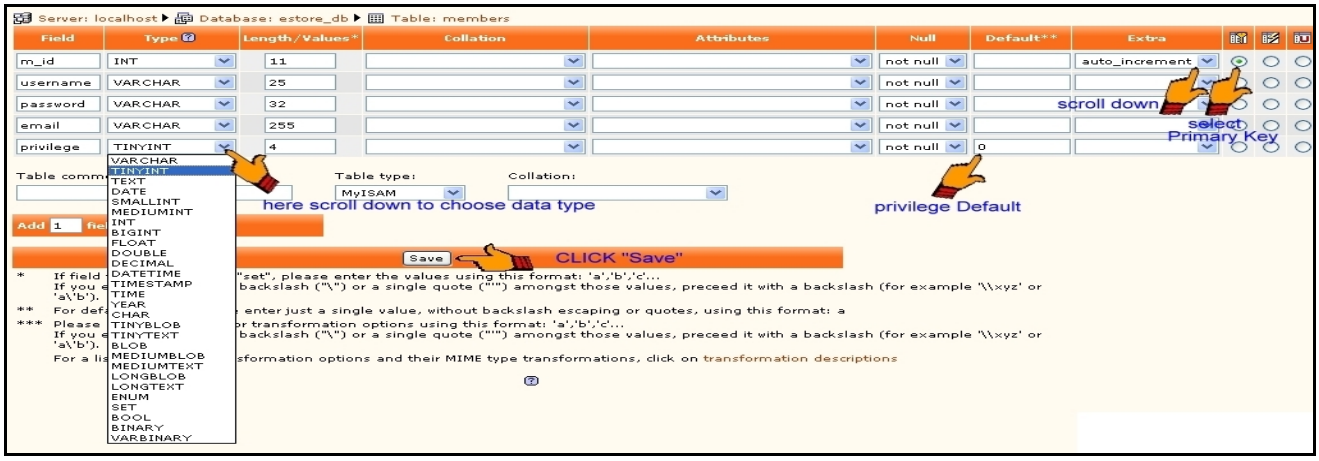


Fig. 1.6. Screenshot showing how to fill out the fields of "**members**" table

The result, that the table "**members**" has been created is shown in (fig. 1.7.)


Fig. 1.7. SQL query to create "**members**" table

- Create the "**products**" table with 3 fields

To create "**products**" table**,** click under "estore_db" link on the left frame of phpMyAdmin: and you will have again the form to create a new table (fig. 1.8.).

Fill the "Name" box with the name of your table: **products,** the "Number of fields" box with number 3, and then click "Go", (fig. 1.8.):

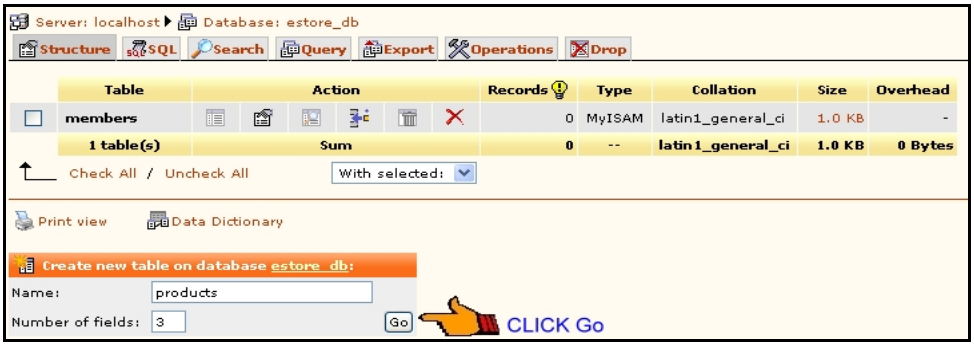


Fig. 1.8. Screen showing you how to create "**products**" table

- Fill out the form fields of "**products**" table (fig.1.9)

**id -** is a integer number.

So, you must fill out the **id** field, with:

Type: INT  Length/Values: 11  Null: not null  Extra: auto_increment  select Primary

**product** - is a text and you can choose "VARCHAR" with length 32.
So, you must fill out the **product** field, with:

Type: VARCHAR Length/Values: 32

**price –** is a decimal number, and you must use "DECIMAL" type of data. Length 5 allowed for decimal digits (the precision) and 2 allowed for numbers of digits after the decimal point (the scale) are adequate.
So, you must fill out the **price** field, with:

Type: DECIMAL Length/Values: 5,2 Null: not null Default: 0.00

Fill out the fields like in fig. 1.9. and after then, click "Save" (do not click "Go" button).


Fig. 1.9. Screenshot showing you how to fill out the fields of "**products**" table

The result, that the table "**products**" has been created is shown in fig. 1.10.


Fig. 1.10. SQL query to create "**products**" table

- Create the "**orders**" table with 4 fields

The procedure to create "**orders**" table is the same with the above tables. The "**orders**" table will be filled out with:

**o_id -** is a integer number and need to be auto_increment or Primary Key
So, you must fill out the **o_id** field, with:

Type: INT Length/Values: 11 Null: not null Extra: auto_increment select Primary

**date -** is a date and you must use "DATE" data type.
So, you must fill out the **date** field, with:

Type: DATE Null: not null

**username -** is a text.
So, you must fill out the **username** field, with:

Type: VARCHAR Length/Values: 25 Null: not null

**comments -** is a text which can be extremely long, therefore you can use "BLOB" data type.
So, you must fill out the **comments** field, with:

Type: BLOB Null: not null

The result, that the table "**orders**" has been created is shown in fig. 1.11.

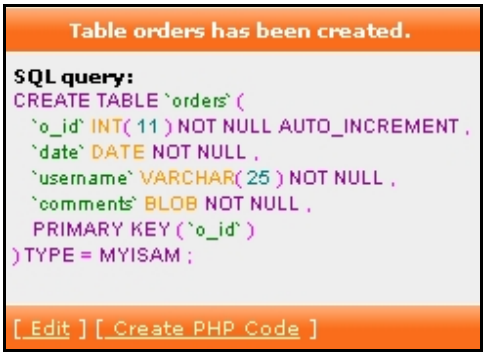

Fig. 1.11. SQL query to create "**orders**" table

- Create the "**orders_items**" table with 4 fields

The procedure to create "**orders_items**" table is the same with the above tables. The "**orders_items**" table will be filled out with:

**o_id -** is a integer number. It does not need to be auto_increment or primary key, because we can have different records with the same o_id value here. The meaning of this is that we can have different products on a same order.
So, you must fill out the **o_id** field, with:

Type: INT Length/Values: 11 Null: not null

**product** - is a text.
So, you must fill out the **product** field, with:

Type: VARCHAR Length/Values: 32 Null: not null

**qty** – is a integer.
So, you must fill out the **qty** field, with:

Type: INT Length/Values: 11 Null: not null

**price -** is a decimal.
So, you must fill out the **price** field, with:

Type: DECIMAL Length/Values: 5,2 Null: not null Default: 0.00

The result, that the table "**orders_items**" has been created is shown in fig. 1.12.

Fig. 1.12. SQL query to create "**orders_items**" table

## 1.2. Exporting (dump) a database as a .sql file in a folder

The need of exporting (dump) a database is the need to have a backup of our database. This way we can restore the structure and data of our database any time.

Let's create a folder named "estore" under: C:\apachefriends\xampp\htdocs\ .

In the "estore" folder you will export (dump) the database as a file with extension .sql. This .sql file is a text type file where you will have the structure of the "estore_db" database.

Open http://localhost/phpmyadmin. In the left frame, scroll down until "estore_db" and click to open it. Click on "Export" tab on the top of the right frame. (fig.1.13.)

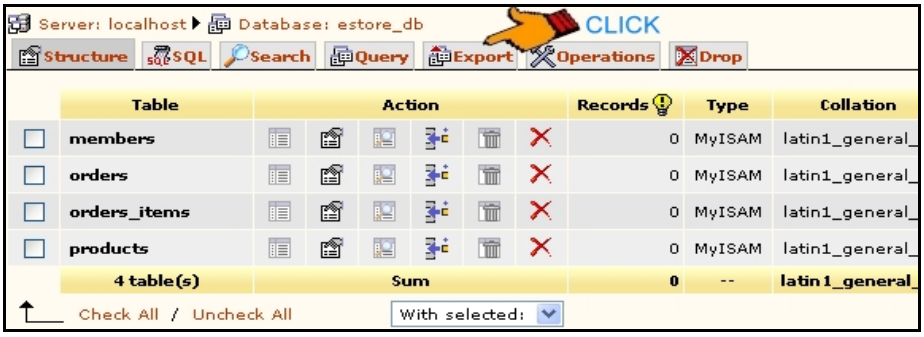

Fig.1.13. Screenshot showing the tab from where to export "estore_db"

A new window will be open (fig. 1.14.).

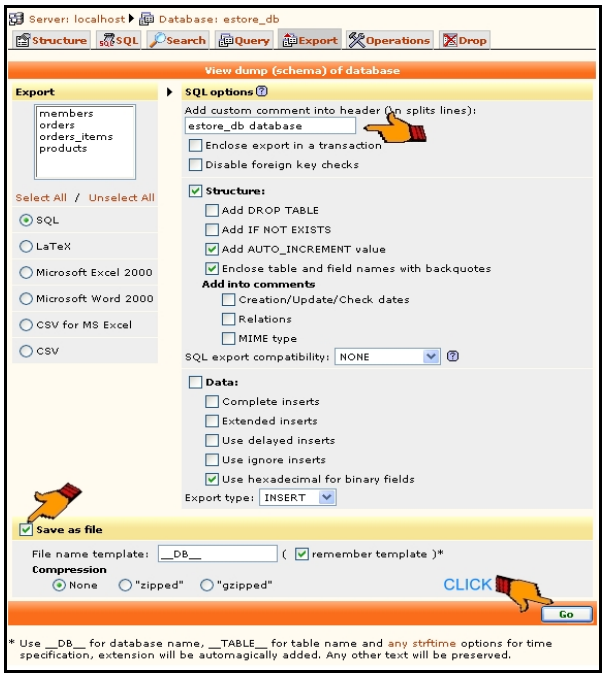

Fig.1.14. Dump (schema) of database

After clicking "Go" a new box will be open, in which you must click "Save". You will be asked where you want to save. You will select your "estore" folder from C:\apachefriends\xampp\htdocs\.
So, if you go now to C:\apachefriends\xampp\htdocs\estore, you will find your file estore_db. sql.

# 2. How to drop and restore a database

## 2.1. How to drop a database

Go to [http://localhost/phpmyadmin)](http://localhost/phpmyadmin) and select your database "estore_db (4)"

In the right frame, you are seeing your database structure and if you want to drop it, click on "Drop" tab from top (fig. 2.1.)

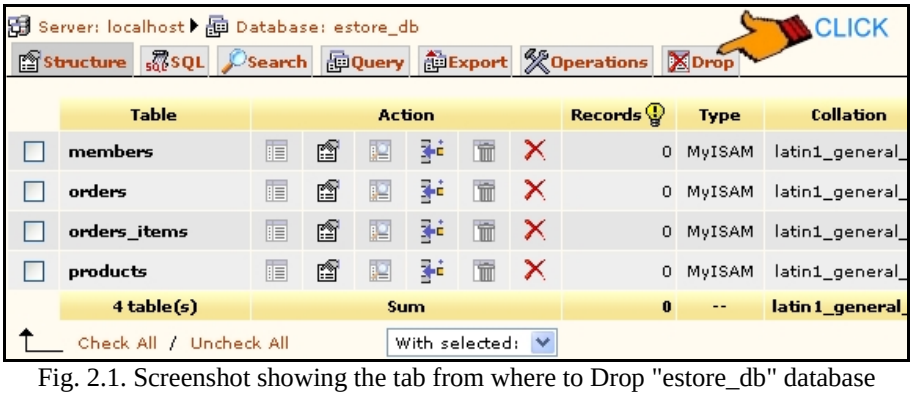

Fig. 2.1. Screenshot showing the tab from where to Drop "estore_db" database

It will appear a small box in which you are asked if: you really want to DROP DATABASE `estore_db`. You will click OK. (if you will choose "Cancel" your database will be not deleted). Your database "estore_db" is now dropped! (fig.2.2.)


Fig.2.2. Screenshot showing that "estore_db" database has been dropped

## 2.2. How to restore a database

To restore "estore_db" database go to http://localhost/phpmyadmin/ and create a database with the name estore_db (fig.1.3.). After the database has been created, click on "SQL" tab from top of the right frame.
On the next screen you will see Browse button. Click Browse button and locate estore_db.sql on your computer to C:\apachefriends\xampp\htdocs\estore, select and open estore_db. sql file. Then click "Go"(see fig.2.3.)


Fig.2.3. Restore "estore_db" database

After this you will have a surprise that all the tables of your database have been successfully created. That means you have the same “estore_db” database that you have been deleted earlier. (fig.2.4.) Congratulations!


Fig.2.4. SQL query has been executed

# 3. How to drop a table and a field

## 3.1. How to drop a table from a database

Let’s say you want to drop "**products**" table. Go to http://localhost/phpmyadmin and select your database "estore_db ( 4 ) ". Click on "Drop" sign of "**products**" table (see fig.3.1.).

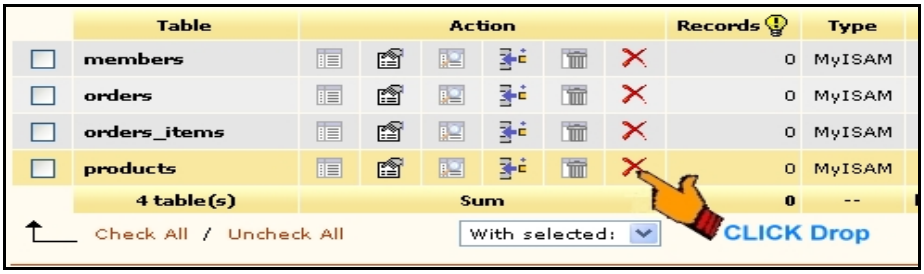


Fig 3.1. Screenshot showing from where to drop "**products**" table from the database

A small box will be appearing (see fig. 3.2).


Fig.3.2. OK to drop the table


Fig. 3.3. Screenshot showing that the table `products` has been dropped

You learned to delete a table from a database.
*(Note: to drop much more tables at once, you just must select all of them and then go to "With selected" box, scroll down to "Drop" and click on it).*

## 3.1. How to drop a field from a table

Let's say you want to drop `comments` from the "**orders**" table. Go to http://localhost/phpmyadmin and select your database "estore_db (4) ". Click on "Structure" sign of "**orders**" table (see fig.3.4.).

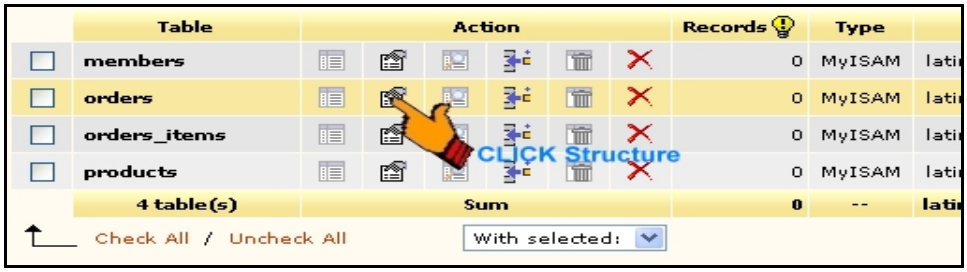


Fig.3.4. Screenshot showing from where to open the structure of "**orders**" table

In the next window you will see the structure of "**orders**" table and click on "Drop" sign (see fig.3.5.)

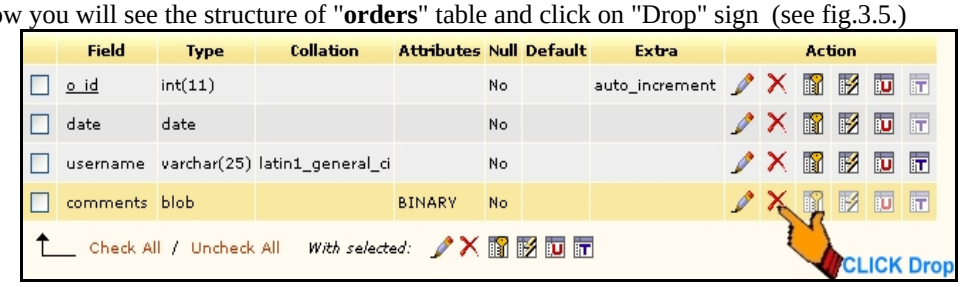


Fig 3.5. Screenshot showing from where to drop `comments` field from "**orders**" table

A small box will be appearing (see fig. 3.6).


Fig 3.6. OK to drop the field


Fig. 3.7. Screenshot showing that the field `comments` has been dropped

You learned to delete a field from a table (fig.3.7.).

*(Note: to drop much more fields at once, you just must select all fields you want to drop and then to go down of table structure to the "Drop" sign and click on it).*

With these general knowledge's about what is and what you can do in a MySQL database you can go on with PHP.

## Key Point Summary Conclusions and Recommendations

PhpMyAdmin interface helps you to save your time in managing MySQL databases.
We don't recommend inserting records directly in your database such as: products. It is possible to have `products` fields in more tables and if you omit to insert in any of them than you will have a MySQL error.
We recommend you to download MySQL manual, it would help you very much in learning MySQL.

## Study Guide

**ESSENTIAL QUESTIONS FOR THE VERIFICATION OF THE ACCOMPLISHED KNOWLEDGE**

1. How can you add a new table in your database?
2. How can you add a new field named "my_field", VARCHAR type with length 20, in "**products**" table?

3. How can you drop the "my_field" field from "**products**" table?
4. How can you change the Length/Values allowed to VARCHAR from 25 to 30 of username field the. ?
5. How can you create a new database named: "toys_database". Using SQL query of the lesson, create t the same "**members**" table as in the lesson.
6. How can you add a product "pencil" with the price 5.50, in your database?

## BIBLIOGRAPHY. REFERENCES.


[1.] Luke Welling, Laura Thomson, *MySQL Tutorial MySQL Press*, November 2003 ( publisher ) ISBN: 0-672-32584-5
[2.] Paul DuBois MySQL, *Third Edition Sams Developer's Library*, March, 2005 ISBN: 0-672-32673-6


## SUPPLEMENTARY IMPORTANT BIBLIOGRAPHY. REFERENCES.


[SUP.1] http://www.mysql.com/ - everything what you want to know about MySQL
[SUP.2] http://dev.mysql.com/ - here you find all kind of interesting links about MySQL
[SUP.3] http://dev.mysql.com/doc/ - from here you can download MySQL manual
[SUP.4] http://dev.mysql.com/doc/refman/5.0/en/ - here you find online MySQL manual


## SUPPLEMENTARY INDICATIONS ABOUT THE CONTENT OF THE LESSON

It will be better that you will learn moreover than in this lesson how can you work with phpMyAdmin interface. You can learn to make more operations with your database

## ANSWERS TO THE QUESTIONS

1. To add a new table in your database you must follow the steps:
   - to think what name should have your table
   - to design the structure of the table (how many fields should contains, their names and properties)
   - then you can go to http://localhost/phpmyadmin/ to start in adding your table
   - select your " estore_db" database
   - and in the "Create a new table on database estore_db" form (fig.ex_1.1.), fill out the "Name" and "Number of fields"

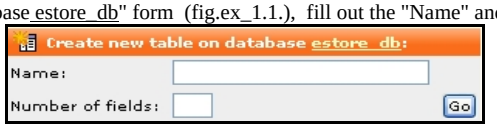


Fig. ex_1.1. Adding a new table in your database

   - then click "Go"
   - in next screen you must to fill out the fields
   - then click "Save"

And your table has been added.

2. To add a new field named "my_field" in "**products**" tables you must follow the steps:
   - go to http://localhost/phpmyadmin/
   - select your " estore_db" database
   - open the structure of "**products**" table (fig. ex_2.1.)

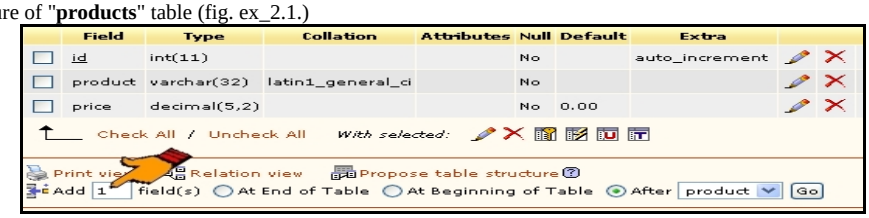


Fig.ex_2.1 Structure of "**products**" table, showing where you can add field.

   - fill out the "Add field(s)" box with 1 (because you want to add only one field). Also you can choose where your field to be placed (we choused to be placed "After" product, see fig. ex_2.1.)
   - click "Go"
   - in the next screen fill out the field with: name: "my_field", type: VARCHAR, Length/Values: 20.
   - then click "Save"

Your field was added.

3. To drop the "my_field" field from "**products**" tables, you must follow the steps:
   - go to http://localhost/phpmyadmin/
   - select your " estore_db" database
   - click "Structure" sign of "**products**" table
   - click "Drop" sign see fig. ex_3.1.)

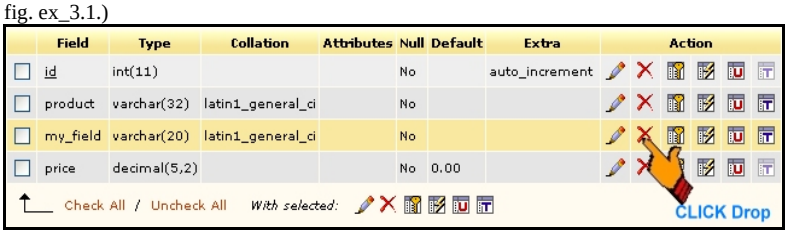


Fig.ex_3.1. Structure of "**products**" table, showing from where you can drop `my_field` field.

   - a small box will be appear


Fig.ex_3.2. OK to drop the field

- click OK , and `my_field` field was dropped.

4. First of all we must notice that username field is present in in "**members**" and in "**orders**" tables. We will change the property as required in both tables.
a) To change the Length/Values allowed to VARCHAR from 25 to 30, of username field, in "**members**" table, you must follow the steps:

- go to http://localhost/phpmyadmin/
- select your " estore_db" database
- open the structure of "**members**" table (fig. ex_4.1.)
- click "Change" sign fig.ex_4.1.

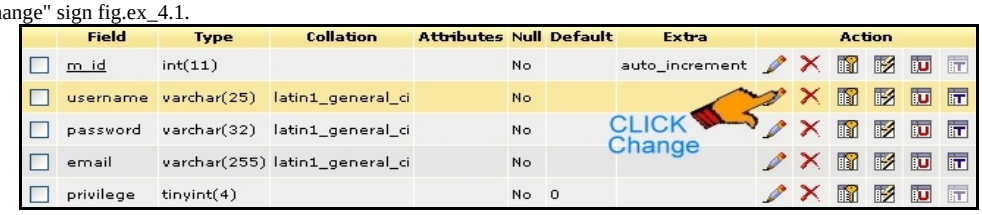


Fig.ex_4.1 Structure of "**members**" table, showing from where change the VARCHAR length of username field.

- in Length/Values column change 25 with 30 (fig.ex_4.2.)


Fig.ex_4.2. Screenshot showing how to change VARCHAR length of username field

- then click Save.

b) To change the Length/Values allowed to VARCHAR from 25 to 30, of username field, in "**orders**" table, you must follow the same steps as above.

5. To create a new database named toys_database you must follow the steps:

- go to http://localhost/phpmyadmin/
- write toys_database along of "Create new database" field (fig.ex_5.1.)

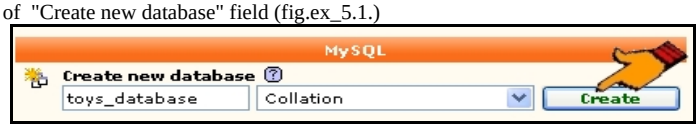


Fig.ex_5.1. Screenshot showing where create "toys_database" database

- click on Create button

The result will be that the "toys_database" has been created.

Now, to create the same table "**members**" as from the lesson, helped by SQL query, you must follow the steps;

- click SQL tab from the top (fig.ex_5.2.)


Fig.ex_5.2. Screenshot showing where to click on SQL tab

- a new screen will be open (fig.ex_5.3.)

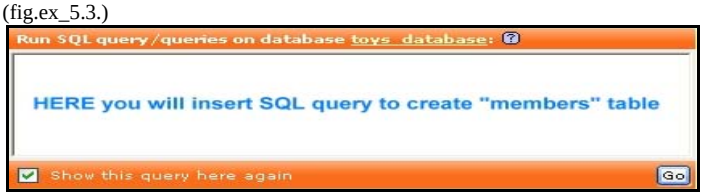


Fig.ex_5.3. Screenshot showing where you will insert SQL query to create "**members**" table

- insert the SQL query to create "**members**" table (fig.ex_5.4.)

*(TIP: you will find the SQL query of "**members**" table in the* estore_db. sql file (where you exported your database) to C:\apachefriends\xampp\htdocs\estore*)*

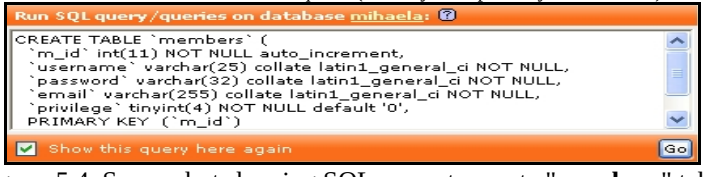


Fig.ex_5.4. Screenshot showing SQL query to create "**members**" table

- click Go (see fig.ex_5.4.)

The "**members**" table was created in "toys_database".

6. To add the product "pencil" in your database, you must follow the steps;

- go to http://localhost/phpmyadmin/

- select your " estore_db" database
- click "Insert" sign fig.ex_6.1.

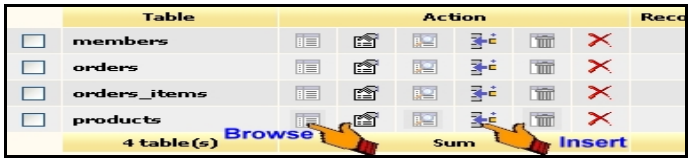


Fig.ex_6.1. Screenshot showing where to click on "Insert" sign

- a new screen will be open (fig.ex_6.2.). Here you fill out the fields with the name of the product "pencil" and its price5.50.

Is not necessary to fill out the "id" field, as you know it is auto_increment. And then click "Go"

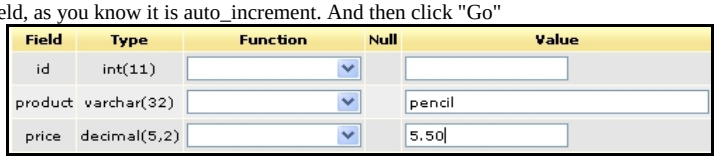


Fig.ex_6.2. Screenshot showing where insert a product "pencil" and its price5.50 in "estore_db" database.

If you want see the inserted product click "Browse" sign, from the same row with table "products" fig.ex_6.1.and you see your product in database. Fig.6.3.

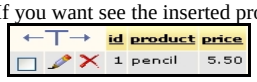


Fig.6.3. Product "pencil" and its price inserted database. From here you can select, edit and delete it.

WORDS TO THE LEARNER:
*In doing we learn*