**Responsive PHP Shopping Cart with Bootstrap Tutorial (Step2)**

**PHP & MySQL 102**

**(Skill: Data Scientist)**

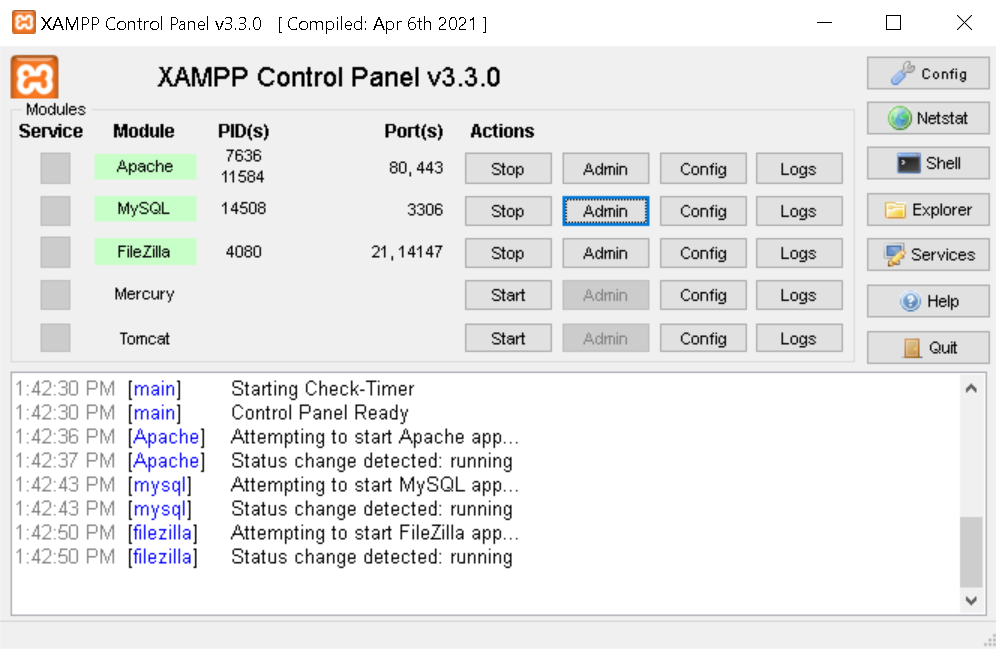
**YouTube: Responsive PHP Shopping Cart with Bootstrap Tutorial, by: Clever Techie**

**SQL** (Structured Query Language) *(pronounced S-QL)*

**SEQUEL** (Structured English Query Language). Originally developed by IBM in the 70s. *(pronounced See Quel).*

**Tutorial 1: Download phpMyAdmin.net**

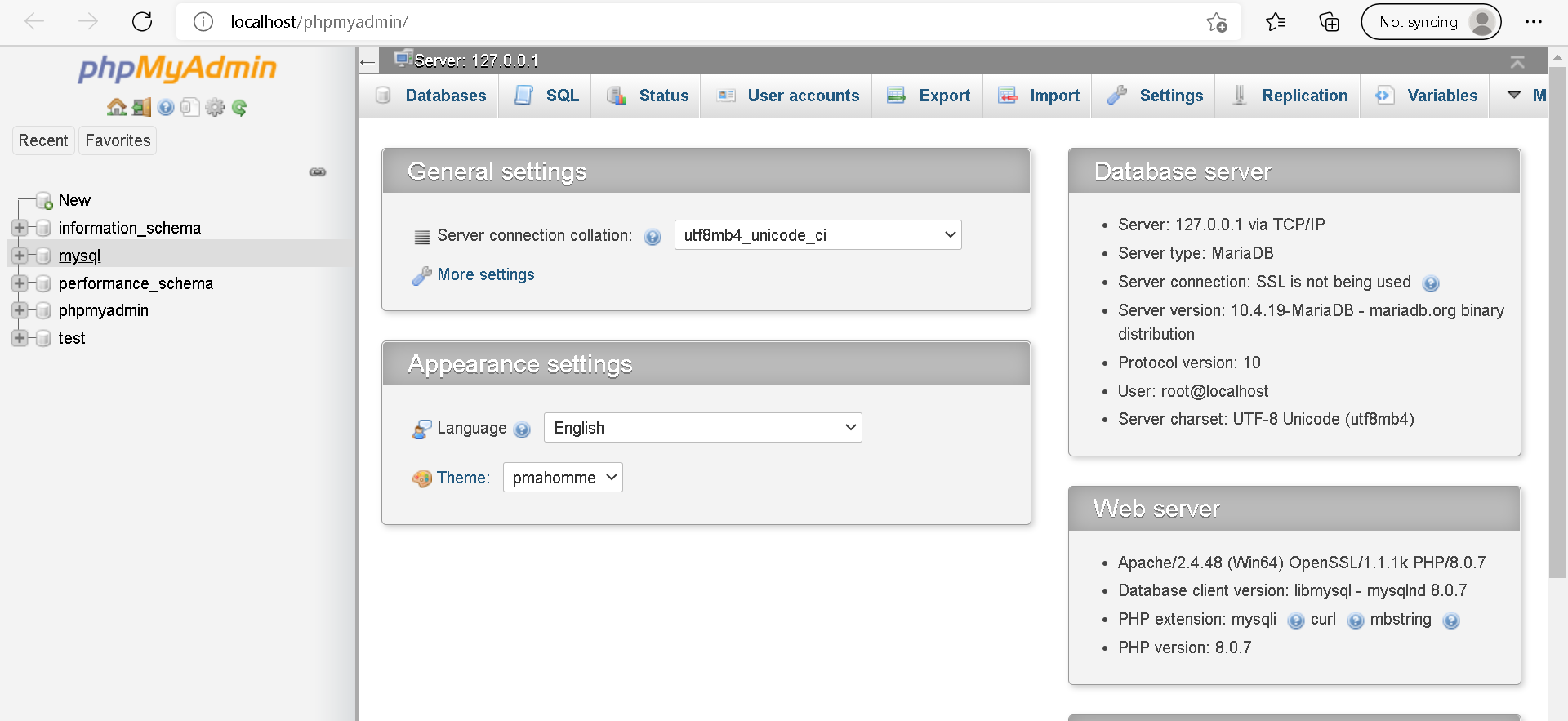
1. Launch the Internet.
2. In the URL type **phpMyAdmin.net**
3. Click on Support
4. Scroll down to “Use the phyMyAdmin source..Click on **quick install guide.**
5. Under **Installing on Windows** click on **XAMPP.**
6. Go to the green arrow that says **Download click here for other versions.**
7. Click on **XAMPP for Windows.** (the download begins).
8. After downloaded go to your Local Disk: C; **right click on xampp windows x64….**
9. Install the program.
10. Or if this doesn’t work and you receive an error when trying to strat MySQL go to Step 10
11. In the URL type https://www.filecroco.com/download-xampp/download/. Or Google xampp. If MySQL throws out an error, uninstall the XAMPP program then reinstall it. See Figure 1.



**Figure 1**

**Objective:** Tables; Relationships; Joins; Subqueries: Regular Expressions.

**MySQL Workbench** is used to connect to your database server.

**PHP** is a fast and feature-rich open source scripting language used to develop Web Applications or Internet / Intranet Applications. **MySQL** is a powerful open source database server built based on a relational database management system (RDBMS) and is capable of handling a large concurrent database connection. **MySQL**  is the most popular database management system used in the world. See Figure 2.

**Figure 2**

The above window opens after typing in localhost/phymyadmin/

The Server is: 127.0.0.1

**Overview: What is a Database?** A database is a collection of data stored in a format that can easily be accessed. In order to manage our databases, we use a software called **“phymyadmin**” or another Database Management System (DBMS). We collect data from the DBMS and give it instructions for querying or modifying data. The DBMS will execute our instructions and send results back.

**Types of Database Management Systems:** There are several DBMS in the cloud and they are classified into two categories: Relational (SQL—Structured Query Language) and NoSQL (Non Relational). However there are many different DBMS out there: **RDBMS**; **MySQL**; **SQL** Server(Microsoft) and **Oracle**. Each DBMS has a unique form of SQL however, all of the implementations are similar in nature and are based on the standard of SQL specification. Most of the SQL code you will learn will work with any DBMS.

**Relational Databases:** We store data in tables that are linked to each other: Customers; Products; Orders forming relationships. This is why we call it Relational Databases. Each table stores data about each specific object like Customer, Product, Order. Therefore SQL is the language we use to work with the relational database management systems. It looks like this…….

**SQL (Structured Query Language)**

SELECT \*

FROM products

WHERE category = ‘food’

ORDER BY price

We use SQL to query or modify our data.

**NoSQL (Non-Relational) Databases:** We don’t have tables that form relationships. Non-relational database management systems don’t understand SQL. They have their own query language.

**FYI:**  Make sure the **PHP** files(index.php; cart.php) and **Image** files are in the same folder.

**Tutorial 2: Create a Responsive PHP Shopping Cart with Bootstrap Tutorial**

**Step 1:**

1. Create shopping cart database and products table
2. Insert data into products table
3. Display products

These 2 steps are completed with pure PHP

* Connect to the database and run MySQL query to select all products from the products table
* Loop through all the product results
* Display product data inside “add to cart form” with image, name, price, quantity text field and add to cart button

1. Display order details

* Check if the add to cart form has been submitted
* Check if the $\_SESSION shopping cart exists
* Create counter to keep track of how many products are in the shopping cart so we know the array key for next product

If shopping cart exists:

* Create numeric $product\_ids array which will track array keys and match them to product ids
* Add next product to existing $\_SESSION shopping cart array using $count as next array key

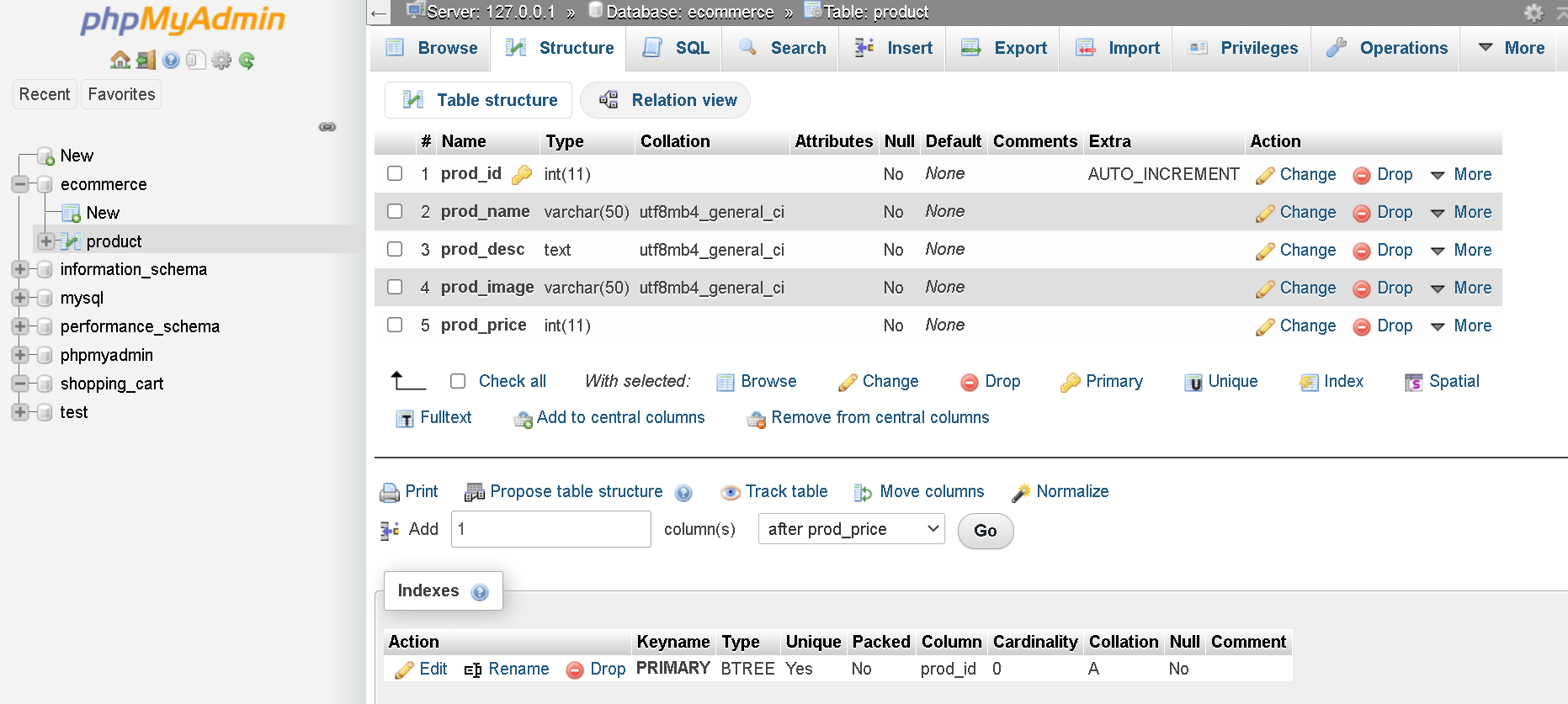
If shopping cart doesn’t exist:

* Create array from data submitted by the add to cart form
* Create $\_SESSION shopping cart array with submitted product data using first array key 0
  + Calculate individual product total
  + Calculate grand total for all products
  + Display order details on page including product name, quantity, price, total and grand total

1. Add “remove” button next to each product

* Loop through the $\_SESSION shopping cart variable and match it with GET id of the product being deleted
* Unset array key of the product in the $\_SESSION variable
* Reset array keys in the $\_SESSION of the product cart so they match with $product\_ids array

1. Display checkout button if the shopping cart is not empty
2. Click the **Save** button after entering the data. Below is the result. See Figure 5.



**Figure 5**

**8.** Now you are ready to work on inserting new data into the database table.