Laboratory Activity
Web Systems and Technology
Midterm

Procedure:

Step 1: Install a local server environment

Since phpMyAdmin requires a web server (Apache), a database server (MariaDB/MySQL), and PHP to run, you will need to install a bundled software package.

Popular options include:

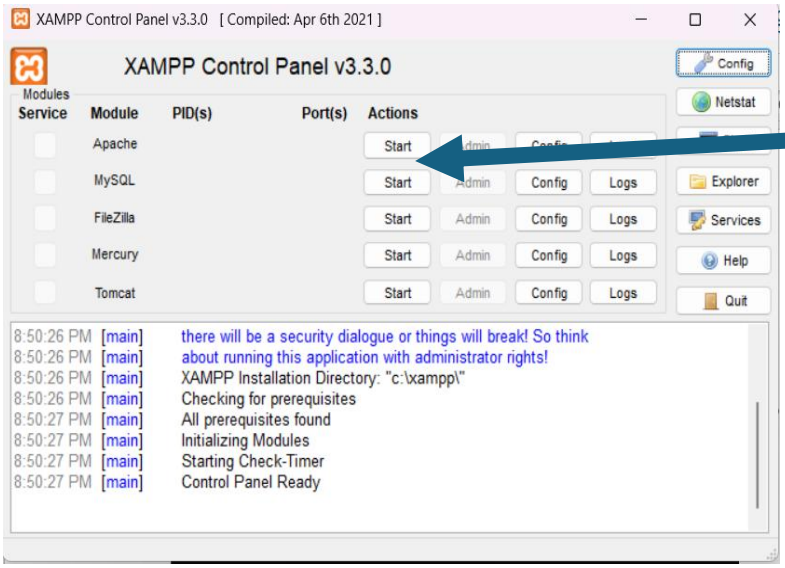
- **XAMPP:** Works on Windows, macOS, and Linux.
- **WAMP:** For Windows users.
- **MAMP:** For macOS users.

Step 2: Start the required services

After installing your server environment, you need to open its control panel and manually start the services.

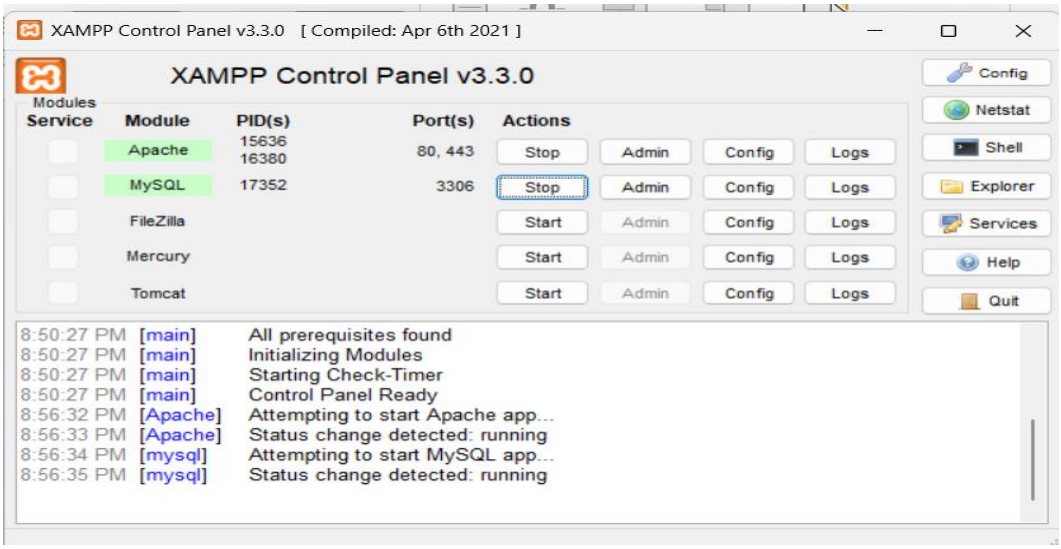
For XAMPP, this means:

1. Open the **XAMPP Control Panel**.
2. Click the **Start** button next to **Apache**.
3. Click the **Start** button next to **MySQL**.
4. Ensure that both modules are running successfully.

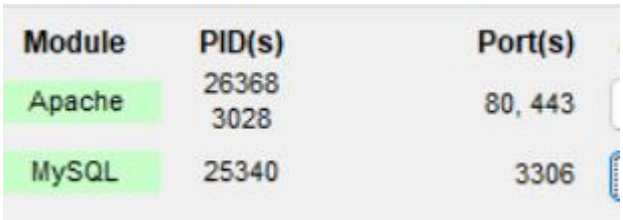


Make sure to click the Start button to open the Apache and MySQL

- Once it green your server is now running



- Make sure to check the ports of Apache and MySQL as follows

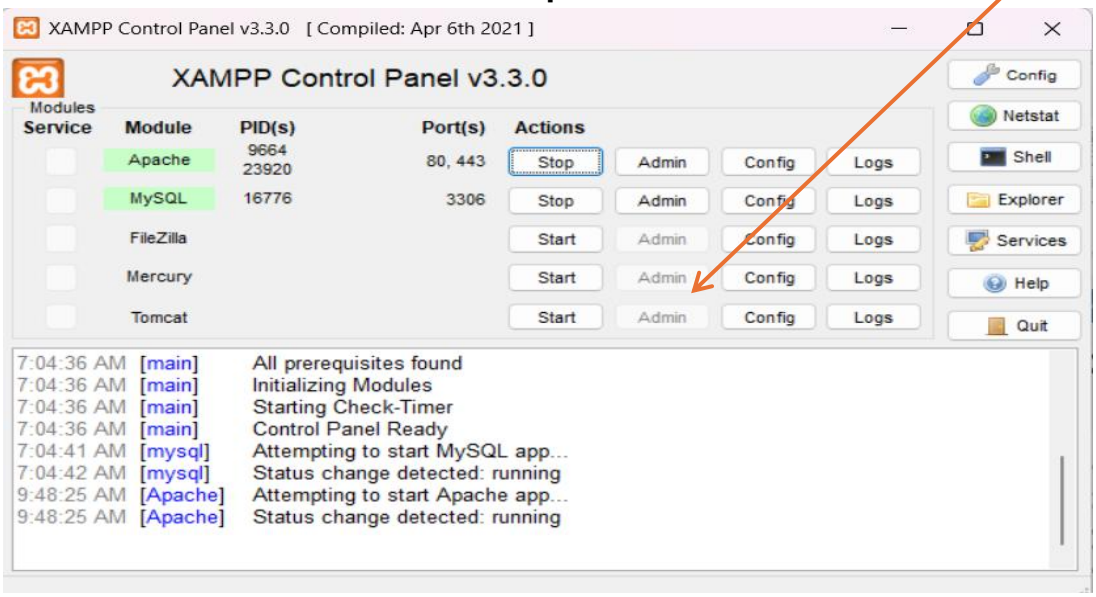


- Your now ready to test your server

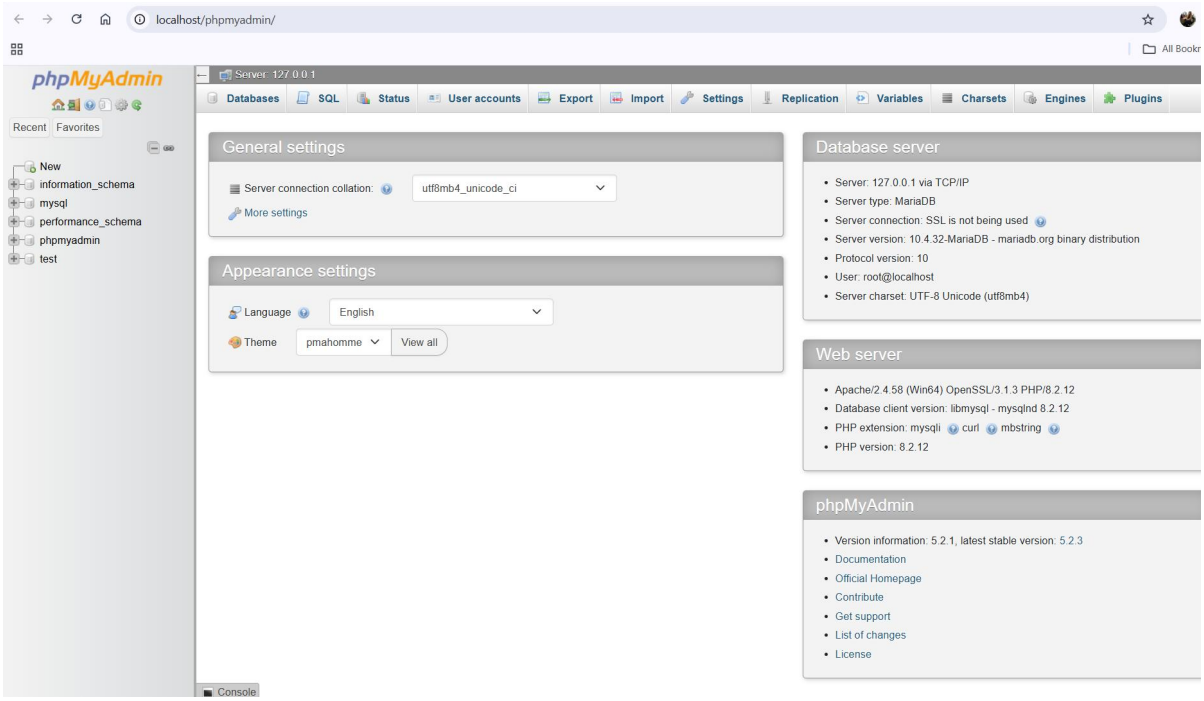
Step 3: Access phpMyAdmin

Once your servers are running, you can access the phpMyAdmin interface through your web browser.

1. Open your web browser.
2. Type http://localhost/phpmyadmin in the address bar and press Enter.
3. Log in with your credentials. If you are using a new installation, the default username is often **root** with **no password**.



Then it will redirect you to the admin panel



Step 4: Plan your database schema

Before creating your database, consider what information you need to store and how it should be organized. Proper planning helps avoid structural problems later on.

A simple planning process involves:

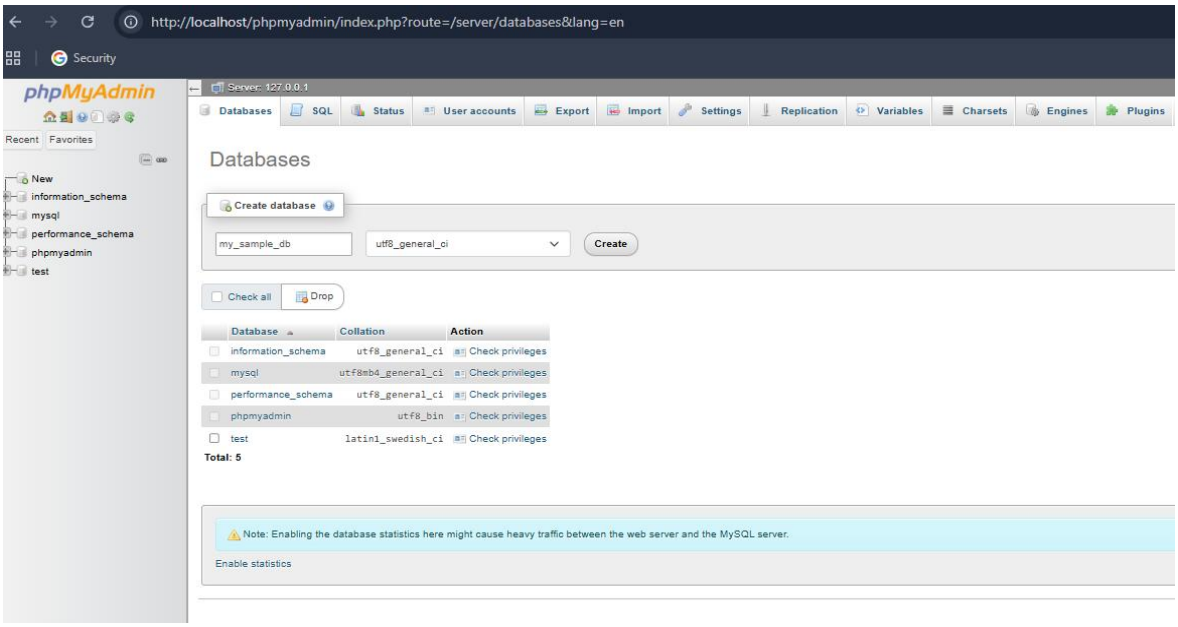
- **Identifying entities:** Figure out the main "things" your database needs to track, like Users, Products, or Orders.
- **Defining attributes:** Decide on the specific data points for each entity. For a Users table, this might include first_name, last_name, and email.
- **Specifying primary keys:** Choose a unique identifier for each table, such as a user ID. This is typically an auto-incrementing integer.
- **Establishing relationships:** Determine how your tables will connect to each other. For example, the Orders table might link to a Users table.

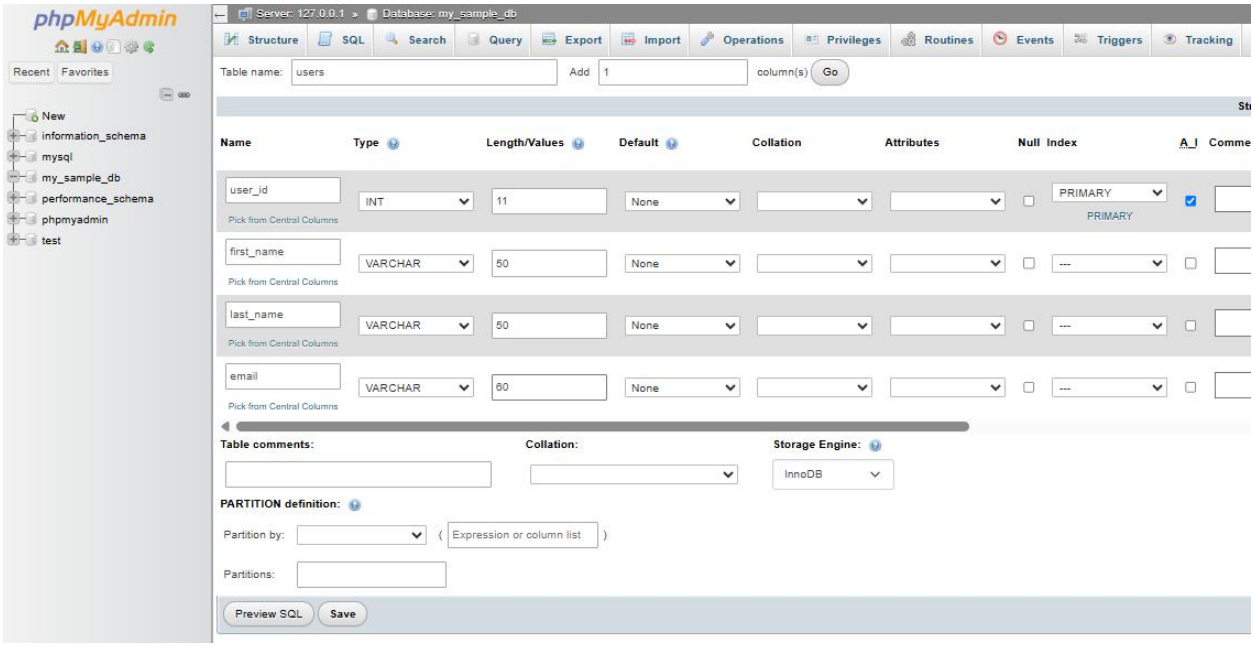
  This time:

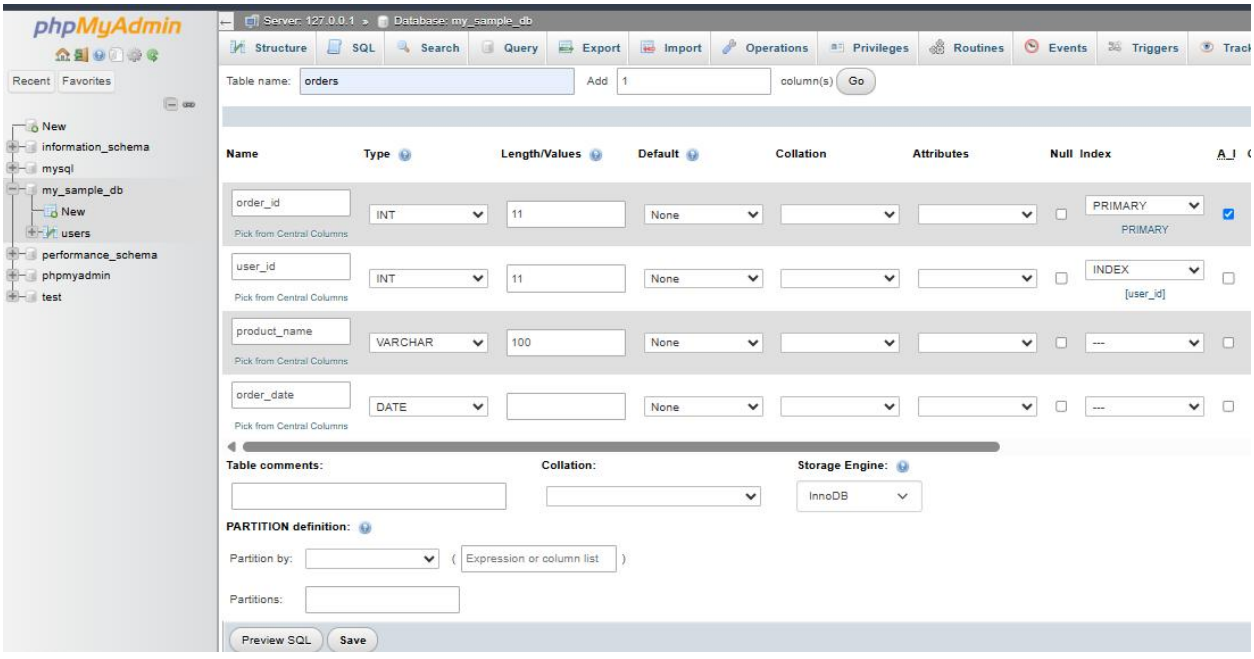
  - I want you to explore or try to create a sample database as your activity for today without connecting it to php (or to your website)
  - Name it on the desired table name.
  - Screenshot all your works and output in this document and sent it through your github (filename: Activity 3 -10/30) this is a different filename from your Activity 2.

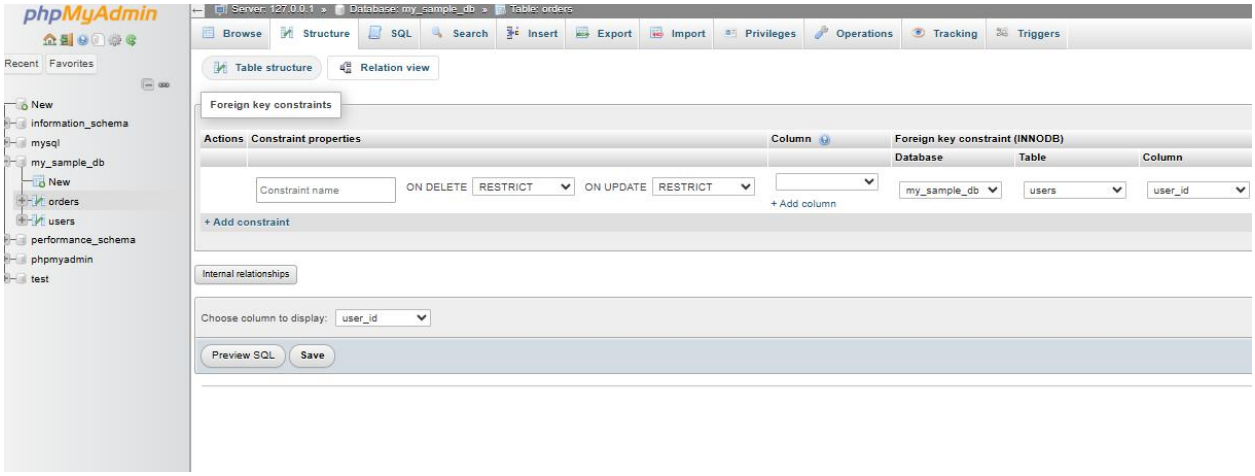
PASTRANA, MARY ROSE V. **C334_IT**

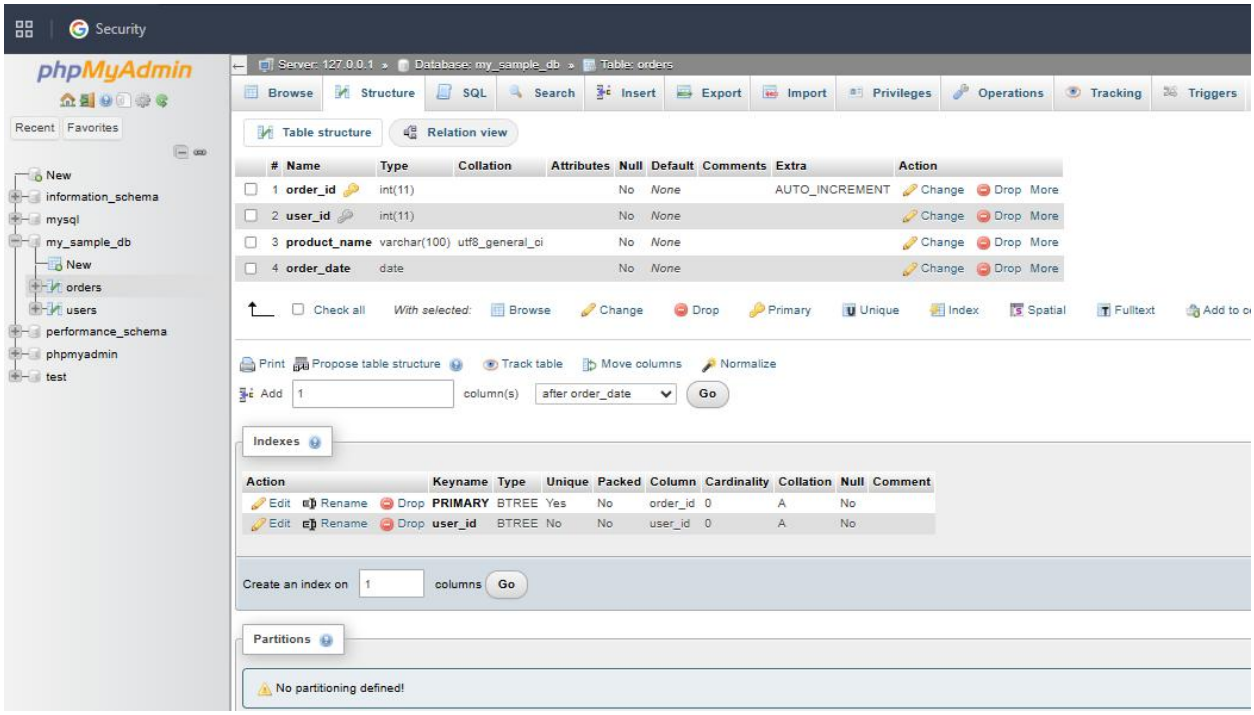
OUTPUT




phpMyAdmin
Recent Favorites
New
information_schema
mysql
my_sample_db
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: my_sample_db
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking
Table name: users
Add 1 column(s) Go
Name Type Length/Values Default Collation Attributes Null Index A_I Comme
user_id INT 11 None PRIMARY PRIMARY
first_name VARCHAR 50 None ---
last_name VARCHAR 50 None ---
email VARCHAR 60 None ---
Pick from Central Columns
Table comments: Collation: Storage Engine: InnoDB
PARTITION definition:
Partition by: (Expression or column list)
Partitions:
Preview SQL Save


phpMyAdmin
Recent Favorites
New
information_schema
mysql
my_sample_db
New
users
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: my_sample_db
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Track
Table name: orders
Add 1 column(s) Go
Name Type Length/Values Default Collation Attributes Null Index A_I
order_id INT 11 None PRIMARY PRIMARY
user_id INT 11 None INDEX [user_id]
product_name VARCHAR 100 None ---
order_date DATE None ---
Pick from Central Columns
Table comments: Collation: Storage Engine: InnoDB
PARTITION definition:
Partition by: (Expression or column list)
Partitions:
Preview SQL Save


phpMyAdmin
Recent Favorites
New
information_schema
mysql
my_sample_db
New
orders
users
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: my_sample_db » Table: orders
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Foreign key constraints
Actions Constraint properties Column Foreign key constraint (INNODB)
Database Table Column
Constraint name ON DELETE RESTRICT ON UPDATE RESTRICT
+ Add column
my_sample_db users user_id
+ Add constraint
Internal relationships
Choose column to display: user_id
Preview SQL Save


Security
phpMyAdmin
Recent Favorites
New
information_schema
mysql
my_sample_db
New
orders
users
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: my_sample_db » Table: orders
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 order_id int(11) No None AUTO_INCREMENT Change Drop More
2 user_id int(11) No None Change Drop More
3 product_name varchar(100) utf8_general_ci No None Change Drop More
4 order_date date No None Change Drop More
Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext
Print Propose table structure Track table Move columns Normalize
Add 1 column(s) after order_date Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Rename Drop PRIMARY BTREE Yes No order_id 0 A No
Edit Rename Drop user_id BTREE No No user_id 0 A No
Create an index on 1 columns Go
Partitions
No partitioning defined!