



Computer Science Department
Web Application and Technologies (COMP 334)
1st Semester 2022/2023

Lab_SQL lab: SQL

Due Date 28/01/2023

Objectives

What You Will Learn

- How to install and manage a MySQL database
- How to integrate user inputs into SQL queries
- How to manage files inside of a MySQL database

Overview

Exercises descriptions and the starting files are given.

You should submit your task by uploading your complete work to the ITC and CSHost. You must submit the following files:

- You must create databases as given below

The instructions and the working files are given

- This lab will help you to create your database schema import and export them to and from the local server. **. You must do the exercises, but it will not be graded.**

LAB 04

WORKING WITH DATABASES

What You Will Learn

How to install and manage a MySQL database

How to manage files inside of a MySQL database

Fundamentals of Web Development, 2nd Ed

Randy Connolly and Ricardo Hoar

Textbook by Pearson
<http://www.funwebdev.com>

Terminal/Shell Version, Date Last Revised: Mar 2, 2017

INTRODUCING MySQL

EXERCISE 4.1. — PHPMYADMIN

- 1 Start phpAdmin. In CSServer,
Eventually you should see the phpMyAdmin panel as shown in Figure 4.1
- 2 The left side of phpMyAdmin displays the existing databases in MySQL. A default installation of MySQL contains a variety of system tables used by MySQL (which depending on your installation may or may not be visible here).

Check if your installation of MySQL already has the **art**, **books**, and **travels** databases installed (as shown in Figure 4.1). If not, jump to Exercises 4.2 and then return to step 4.
- 4 Click on the **art** database.
This will display the tables in the art database.
- 5 Click the browse link for the **Artist** table.
This will display the first set of records in this table with edit/copy/delete links for each record.
- 6 Click on the Structure tab.
This will display the definitions for the fields in the current table, with links for modifying the structure.

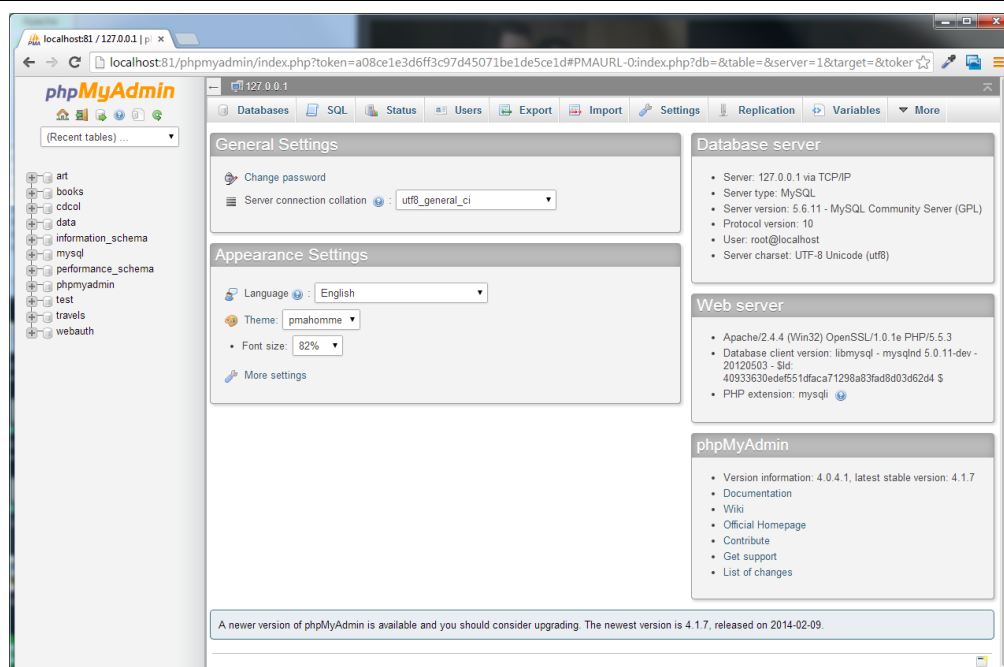


Figure 4.1 – phpMyAdmin

EXERCISE 4.2 — INSTALLING DATABASE IN PHPMYADMIN

- 1 Examine `art-small.sql` in a text editor. When done, close the file.

These import scripts contain the necessary DDL statements to define the database tables as well as the INSERT statements to populate the tables with data.

- 2 In phpMyAdmin, click on the Databases tab.

- 3 Create a database called **art**.

When it is complete, the art database will be visible in left-side of phpAdmin window.

- 4 Click on **art** database on left-side of window.

Currently there are no tables in this database. You can manually create a new table here, or using the art.sql script to populate the database.

- 5 Click on the Import tab.

- 6 Use the Choose File button to select the [art-small.sql](#) file examined in step 1. Then click the Go button.

If import works successfully, then you should be able to examine the tables in the database.

Probably one of the most common problems encountered by students is that a timeout error occurs or that a file exceeds the maximum upload size. Fixing this may require modifying your php.ini configuration file or using the MySQL Monitor instead to import the data. Ask your instructor (or look online) for guidance.

- 7 Create a database named **travel** and then import [travels-small.sql](#). Verify that it creates and populates a variety of tables.

- 8 Create a database named **book** and then import [books-small.sql](#). Verify that it creates and populates a variety of tables.

SQL

MySQL, like other relational databases, uses Structured Query Language or, as it is more commonly called, SQL (pronounced sequel) as the mechanism for storing and manipulating data.

The following exercises assume that your databases have been created and populated. They also use phpMyAdmin to run the SQL commands.

EXERCISE 4.3. — QUERYING A DATABASE

- 1 In phpMyAdmin, click on the art database, then click on the **SQL** tab.

You should see the message “Run SQL query/queries on database art:”

- 2 In the blank SQL window, enter the following.

```
select * from Artists
```

In MySQL, database names correspond to operating system directories while tables correspond to one or more operating system files. Because of this correspondence, table and database names ARE case sensitive on non-Windows operating systems.

- 3 Now press the Go button.

This will run the query. Notice that only the first 30 records are retrieved. This limit is appended to each query for performance reasons (you likely will not want all million records in a given table for instance). If you wish to see all the records retrieved from a query, there is a Show All link at the bottom of the retrieved records.

- 4 Return to the SQL window, enter the following new query, and then press Go.

```
select paintingid, title, yearofwork from Paintings
where yearofwork < 1600
```

This will display just the paintings prior to 1600. Notice that in MySQL, a query can be spread across multiple lines. SQL in general is not case sensitive, which means you do not have to worry about the case of field names. However, remember the comment in the above step 2: in MySQL, table names are case sensitive on non-Windows environments.

- 5 Modify the query (you can click the **Show query box** link) as follows and test.

```
select paintingid, title, yearofwork from Paintings
where yearofwork < 1600 order by yearofwork
```

- 6 Modify the query as follows and test.

```
SELECT Artists.ArtistID, Title, YearOfWork, LastName FROM Artists
INNER JOIN Paintings ON Artists.ArtistID = Paintings.ArtistID
```

This query contains a join since it queries information from two tables. Notice that you must preface ArtistId with the table name since both joined tables contain a field called ArtistId.

- 7 Modify the query as follows and test.

```
SELECT Nationality, Count(ArtistID) AS NumArtists
FROM Artists
GROUP BY Nationality
```

This query contains an aggregate function as well as a grouping command.

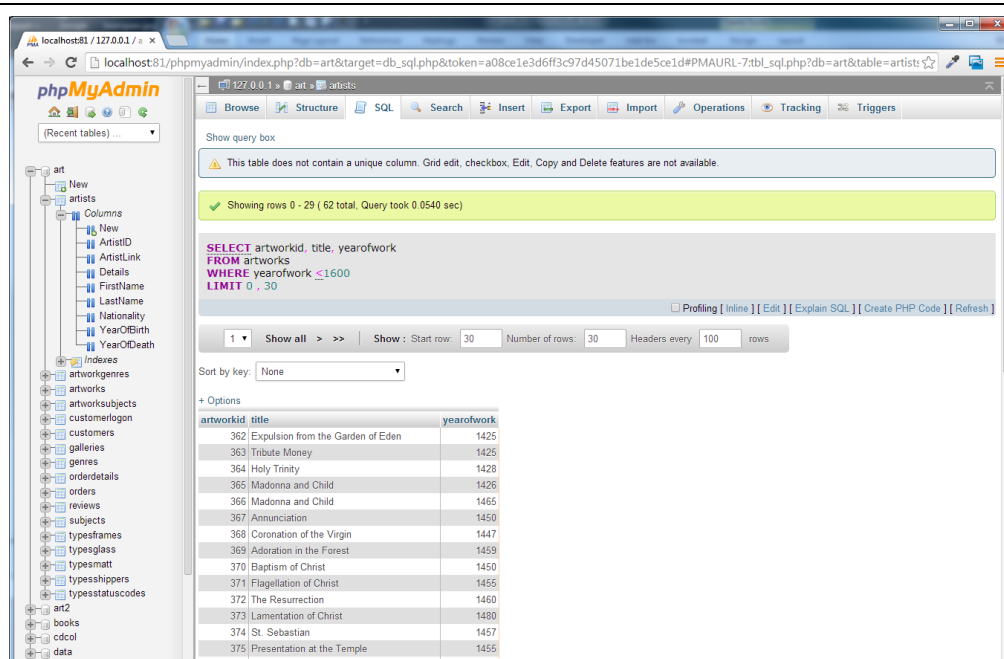


Figure 4.2 – EXERCISE 4.3

EXERCISE 4.4. — MODIFYING RECORDS

- 1 In phpMyAdmin, click on the **art** database, then click on the SQL tab.
You should see the message “Run SQL query/queries on database art:”
- 2 In the blank SQL window, enter the following.

```
insert into Artists (firstname, lastname, nationality, yearofbirth,
yearofdeath) values ('Palma','Vecchio','Italy',1480,1528)
```
- 3 Press the Go button.
You should see message about one row being inserted.
- 4 Examine the just-inserted record by running the following query.

```
select * from Artists where lastname = 'Vecchio'
```


Notice that ArtistId value has been auto-generated by MySQL. This has happened because this key field has the auto-increment property set to true.
- 5 Run the following new query:

```
update Artists
set details='Palmo Vecchio was an Italian painter of the Venetian school'
```
- 6 Verify the record was updated (i.e, by running the query from step 4).
- 7 Run the following new query:

```
delete from Artists
where lastname = 'Vecchio'
```
- 7 Verify the delete worked by running the following query:

```
SELECT * FROM Artists WHERE nationality = 'Italy'
```

One of the key benefits of databases is that the data they store can be accessed by queries. This allows us to search a database for a particular pattern and have a resulting set of matching elements returned quickly. In large sets of data, searching for a particular record can take a long time. To speed retrieval times, a special data structure called an index is used. A database table can contain one or more indexes.

EXERCISE 4.5. — BUILD AN INDEX

- 1 In phpMyAdmin, click on the art database, click on the artworks table, and then click on the **Structure** tab.
- 2 In the Structure page, click on the **Indexes** link.
You will see a list of already-existing indexes. The import script for this database already has created indexes for the primary key, the foreign keys, and a two other fields.
- 3 Click the Go button in the section that begins Create an index ...
- 4 In the pop-up window, name the index YearOfWork, the index type INDEX, and the column YearOfWork as shown in Figure 4.3. Click Go.
There will be a short delay as MySQL creates the index.

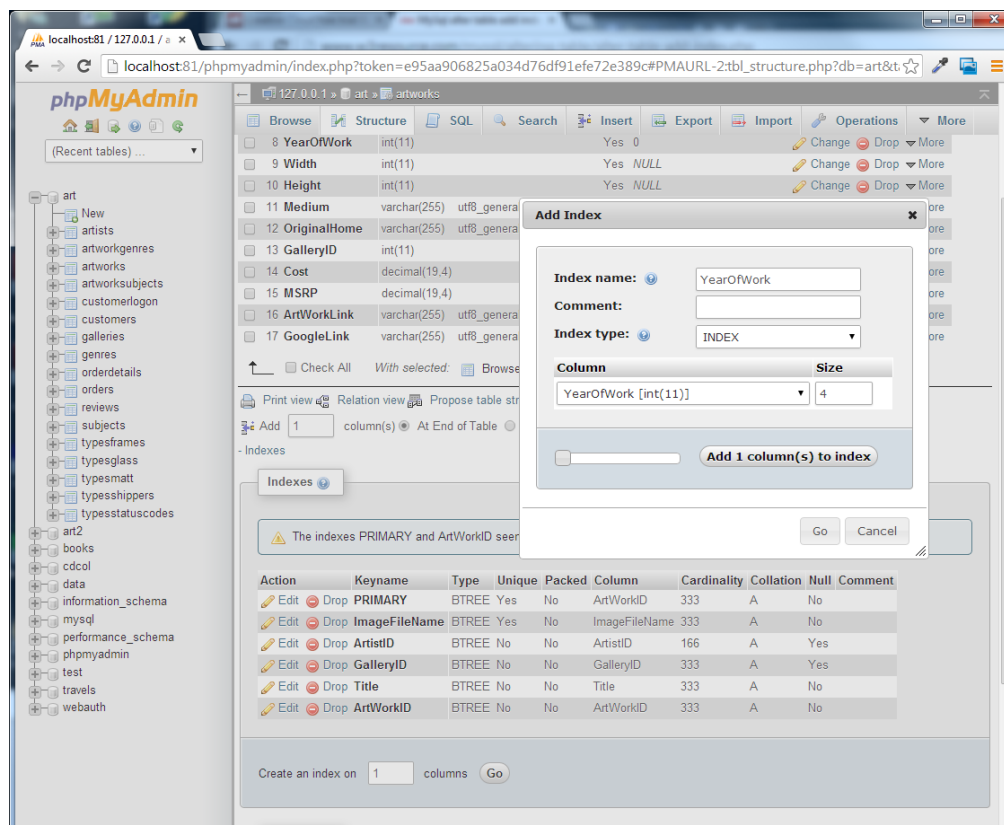


Figure 4.3 – Creating an index in phpMyAdmin

EXERCISE 4.6. — CREATING USERS IN PHPADMIN

- 1 In phpMyAdmin, click on the **art** database, and then click on the **Privileges** tab.
This will display the users who currently have access to this database. Notice the root user. This root user has special privileges within MySQL: indeed, you very well may have logged into phpMyAdmin using the root account. For development-only environments, using the root user will likely be okay. Nonetheless, we are going to create a new user which you will use for subsequent examples in this lab.
- 2 Click the **Add user** link.
This will display the Add user page (see Figure 4.4).
- 3 In the Add user page, enter the following into the Login information section:
 User name (use text filed): **testuser**
 Host (Local):
 Password (use text filed): **mypassword**
 Re-Type: **mypassword**
You are of course welcome to enter a different user name and password. If you do, you will need to substitute future references to testuser and password. Also, depending on the environment you are using, you may need to enter something different in the Host field (perhaps 'localhost' or '127.0.0.1')
- 4 In the **Database for user** section, check the **Grant all privileges on database “art”** checkbox.

- 5 In the **Global privileges** section, check the five Data privileges (select, insert, update, delete, and file).
- 6 Click the Go button.

The screenshot shows the 'Add user' form in phpMyAdmin. The 'Login Information' section has 'User name' set to 'testuser', 'Host' set to 'localhost', and 'Password' and 'Re-type' fields filled with asterisks. The 'Database for user' section has 'Grant all privileges on database "art"' checked. The 'Global privileges' section has 'Data' privileges (SELECT, INSERT, UPDATE, DELETE, FILE) checked. The 'Administration' and 'Resource limits' sections are empty. A 'Go' button is at the bottom right.

localhost:81 / 127.0.0.1

phpMyAdmin

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers More

(Recent tables) ...

art

- New
- artists
- artworkgenres
- artworks
- artworksubjects
- customerlogin
- customers
- galleries
- genres
- orderdetails
- orders
- reviews
- subjects
- typesframes
- typesglass
- typesmatt
- typesshippers
- typesstatuscodes

art2

- books
- cdcol
- data
- information_schema
- mysql
- performance_schema
- phpmyadmin
- test
- travels
- webauth

Add user

Login Information

User name: Use text field: testuser

Host: Local localhost

Password: Use text field:

Re-type:

Generate password: Generate

Database for user

☐ Create database with same name and grant all privileges

☐ Grant all privileges on wildcard name (username, %)

☒ Grant all privileges on database "art"

Global privileges (Check All / Uncheck All)

Note: MySQL privilege names are expressed in English

Data

- ☒ SELECT
- ☒ INSERT
- ☒ UPDATE
- ☒ DELETE
- ☒ FILE

Structure

- ☐ CREATE
- ☐ ALTER
- ☐ INDEX
- ☐ DROP
- ☐ CREATE TEMPORARY TABLES
- ☐ SHOW VIEW
- ☐ CREATE ROUTINE
- ☐ ALTER ROUTINE
- ☐ EXECUTE
- ☐ CREATE VIEW
- ☐ EVENT
- ☐ TRIGGER

Administration

- ☐ GRANT
- ☐ SUPER
- ☐ PROCESS
- ☐ RELOAD
- ☐ SHUTDOWN
- ☐ SHOW DATABASES
- ☐ LOCK TABLES
- ☐ REFERENCES
- ☐ REPLICATION CLIENT
- ☐ REPLICATION SLAVE
- ☐ CREATE USER

Resource limits

Note: Setting these options to 0 (zero) removes the limit

MAX QUERIES PER HOUR 0

MAX UPDATES PER HOUR 0

MAX CONNECTIONS PER HOUR 0

MAX USER_CONNECTIONS 0

Go

Figure 4.4 – Creating a user

GOOD LUCK and ENJOY THE TASK