Your repository details have been saved.   ✕

📖 benjamin-sylvester / **CS-148-lab3**

cs148 lab3   https://bsylvest.w3.uvm.edu/cs148/sit...   Edit

Manage topics

ⓘ **2** commits	⑂ **1** branch	🏷 **0** releases	👥 **0** contributors

Branch: master ▾   New pull request   Create new file Upload files Find file Clone or download ▾

Ben Sylvester first commit		Latest commit a2bbc7c 3 hours ago
📁 css	first commit	3 hours ago
📁 images	first commit	3 hours ago
📁 lib	first commit	3 hours ago
📄 .gitignore	my first commit	14 hours ago
📄 README.md	first commit	3 hours ago
📄 commandsGit.txt	first commit	3 hours ago
📄 footer.php	first commit	3 hours ago
📄 gitignore.png	first commit	3 hours ago
📄 header.php	first commit	3 hours ago
📄 index.php	first commit	3 hours ago
📄 nav.php	first commit	3 hours ago
📄 sampleQuery.php	first commit	3 hours ago
📄 sampleQuery2.php	first commit	3 hours ago
📄 sampleQuery3.php	first commit	3 hours ago
📄 sampleQuery4.php	first commit	3 hours ago
📄 tables.php	first commit	3 hours ago
📄 top.php	first commit	3 hours ago
📄 user.ini.png	first commit	3 hours ago

📖 **README.md**   ✎

Database account at UVM

The purpose of this repository to help you get over the hurdle of setting up a database, it is not intended to cover everything. It is not intended to teach you PHP or mySQL. This is an example of using PHP and mySQL, though the process to connect with Java or Python will use the same credentials. You can see the demo on my site. This demo is setup on the silk server www-root (zoo would be similar).

I made a help video as well of this demo. Hopefully the setup goes as smoothly.

Uvm supports mysql databases that you can connect with php, python, java etc. You do not need to download anything to the server as its all setup already.

STEPS:

1. mySQL account: Request a mySQL username and password. Be sure to save the auto generated email.
2. Create new databases here: https://webdb.uvm.edu/account/. To match my sample name your database(case sensitive): Sample
3. Use PHPMyAdmin to create these tables. You copy the sql statements and paste them (watch the video if you are not sure). You need to click on your database name above first to make sure the tables are created in that database.

   Here is the SQL statements to create the table:

   ```
   CREATE TABLE IF NOT EXISTS tblPeople (
   pmkPeopleId int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
   fldFirstName varchar(25) COLLATE utf8mb4_bin NOT NULL,
   fldLastName varchar(50) COLLATE utf8mb4_bin NOT NULL
   ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_bin;
   ```

   The insert statements to populate the table (note i use a different version of the Insert statement in the example.

   ```
   INSERT INTO tblPeople (fldFirstName, fldLastName) VALUES
   ('George', 'Jetson'),
   ('Jane', 'Jetson'),
   ('Judy', 'Jetson'),
   ('Elroy', 'Jetson'),
   ('Astro', 'Jetson'),
   ('Rosie', 'Jetson'),
   ('Cosmo G.', 'Spacely');
   ```

4. Download the files from this repository
5. Edit the file lib/pass.php and put in your passwords
6. Edit the file lib/constants.php and and change the following to your information. It is case sensitive (for the most part it is just changing my username to yours)
   i. define('DATABASE_NAME', 'RERICKSO_Sample');
   ii. define('DATABASE_READER', 'rerickso_reader');
   iii. define('DATABASE_WRITER', 'rerickso_writer');

Problems encountered have been

1. Naming the database different than my Sample (its ok you just have to make sure you update the name in constants.php)

2. Forgetting to update the passwords in pass.php

3. Not making the table in the Sample database

One item to keep in mind is that I have a constants.php:
define("DEBUG", false);

and one in the database.php class:
const DB_DEBUG = false;

If you set these constants to true it can help you see what your variables are set to in case you are having trouble.

UVM Authentication

Place an .htaccess file shown below in your web folder and it will prompt the user to enter their uvm username and password:

```
<Files *>

AuthType WebAuth

require valid-user

satisfy any

order allow,deny

</Files>
```

then use this php code to retrieve the username

```
// get the user who has logged in
$user = htmlentities($_SERVER["REMOTE_USER"], ENT_QUOTES, "UTF-8");
```

To save yourself time looking for php errors add .user.ini file to your web folder. this will display php errors and warnings on your web page directly (just delete it when you go live). Example

```
display_errors =1;
```