§Web Programming

## Topic: Python with MySQL

## Introduction

## This worksheet is designed to encourage you to perform MySQL database operation using Python which will later be used in Python CGI.

Your tutor will help you by demonstrating what is required and also answer any questions you may have.

**Pre-requisite:**

1. Your memory stick is ready with xampp and Python34.

2. You have downloaded and installed **mysql.connector** (see attached document ‘Instructions mysqlconnector for python.docx’)

3. Your mysql service is running in xampp

4. You need a web server (e.g. apache) to load your python CGI script. This means you should save your python scripts in htdocs/webprog folder in xampp and run it in web browser by typing URL e.g. <http://loaclhost/webprog/yourprogram.py>. Please make sure apache server is running.

5. You have practiced HTML exercises and are familiar HTML elements especially <Form> element.

**Post-requisite:** You should stop apache server and exit from xampp control panel. Also, you should properly unplug your memory stick. It is always a good idea to zip your webprog folder and email it to yourself or make a copy e.g. in Dropbox.

If you need help - **ask**. If you are curious - **ask**. If you want to know - **ask**.

You are encouraged to discuss the workshop with other students in the class. Share and Learn!

**IMPORTANT:**

1. **Always** try to run your python script in IDLE or on terminal (command prompt) to check if there are any errors. Remove errors before loading python scripts in a web browser.

2. For CGI scripts you must check if shebang line is referring to correct version of Python34 folder on your memory stick. Shebang line should be something like: #! /Python34/python

## What you must do

Again you should work in pairs (two or three students). Download mySQLexamples.zip in webprog folder. There are many python MySQL examples included in the extracted folder. Read and understand all programs, run them on terminal window (or IDLE). If anything is not clear then ask!

**Objective here is to learn basics of python mysql to develop basic database applications.**

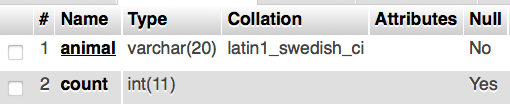
* Make sure you have access to the relevant sites/lectures
* Make sure you have bookmarked links to web help pages and references
* Quickly read python cgi examples and/or any lectures on html (this week).

**Exercise 1: Getting familiar with phpmyadmin**

* **Read phpadminGuide to understand how to use phpmyadmin web interface.**
* **Create a database ‘webprog’**
* **Load ‘webprog’ and create two tables as below:**
* **animals**

****

* **spotted**

****

* **fill above tables with following data (you may use phpmyadmin)**

|  |  |
| --- | --- |
| **animals**   * **Screen Shot 2015-09-20 at 17.15.43.png** | **Spotted**  **Screen Shot 2015-10-05 at 09.11.19.png** |

You can use phpmyadmin user interface to create tables and fields. OR you can use following SQL statements:

*CREATE TABLE animals (id int(11), name varchar(20));*

*CREATE TABLE spotted (animal varchar(20) NOT NULL, count (11), PRIMARY KEY(animal));*

**Hint:** See **webprog.sql** which gives you structure and data of **webprog** database on my machine.

**Exercise 2: Getting familiar with program examples**

Following programs are included:

***mysql\_example\_simple.py 🡪*** *shows if a connection with MySQL database is successful*

***createTable.py 🡪*** *shows how to connect with MySQL database and create table*

***mysql\_example\_delete.py 🡪*** *shows how to connect with MySQL database and delete a record from a table*

***mysql\_example\_insertmany.py 🡪*** *shows how to connect with MySQL database and insert many records into a table using one SQL statement*

***mysql\_example\_insertone.py 🡪*** *shows how to connect with MySQL database and insert one record*

***mysql\_example\_update.py 🡪*** *shows how to connect with MySQL database and update an existing record*

***mysql\_example\_query\_id.py 🡪*** *shows how to connect with MySQL database and select records from a table using user defined parameter (id) – uses fetchone function*

***mysql\_example\_query.py 🡪*** *shows how to connect with MySQL database and select and display record(s) from a table (by using fetchone function) – this example should also run in web browser*

***mysql\_example\_queryfall.py 🡪*** *shows how to connect with MySQL database and select and display record(s) from a table (by using fetchall function)*

***mysql\_example\_queryfmany.py 🡪*** *shows how to connect with MySQL database and select and display record(s) from a table (by using fetchmany function)*

***mysql\_example\_stprocedure.py 🡪*** *shows how to connect with MySQL database and calls a stored procedure from MySQL*

***mysql\_example\_stprocedure\_param.py 🡪*** *shows how to connect with MySQL database and calls a stored procedure where a parameter is passed to stored procedure*

***clientform.html and serverformp4.py 🡪*** *These programs should run in web browsers. This is a working example where clientform.html gets id and sends to serverformp4.py. serverformp4 stores form data into a dictionary named form and gets the value of ‘id’ and after establishing database connection calls stored procedure and passes value of ‘id’ field. After receiving results from stored procedure it creates a web page and displays results.*

***webprog.sql 🡪*** *shows sql statements for animal and spotted tables.*

***dbfunc.py 🡪*** *this is a generic script that establishes MySQL database connection and returns a handle to database that can be used with any other script*

1. In this exercise go through all program examples, understand the code/statements and their purpose. If not clear then ask your tutor.
2. Run all programs in IDLE or terminal window. clientform.html and serverformp4.py should be run using web browser.
3. Try to make appropriate changes in code (e.g. id value can be different when trying to delete a record) and run each example code and see the output. You may like to refer to lecture slides.

**Note:** Make sure you establish database connection using correct parameter values e.g. password

**Exercise 3: Moving towards Python CGI and MySQL**

*Once you have run all programs successfully then try to think what steps you would need to take to convert above python-mysql applications into web based applications. See clientform.html and serverformp4.py and try to convert* ***mysql\_example\_stprocedure\_param.py*** *example into web based application.*

***Hint:*** *adding shebang, creating header, html tags using print statement etc.*

**Exercise 4: Creating a Log file**

*Modify* ***mysql\_example\_update.py*** *in such a way that all updates are logged into a file name log.txt (hint: open log file in append mode) in csv (comma separated values) format. Each log entry should include: operationtype (e.g. update), tablename (e.g. spotted), eventtime (date and time when this action was performed), olddata (i.e. old record row before update operation), newdata (i.e. new record row after update operation).*

*Read data form log.txt and display on screen.*