§Web Programming

## Topic: Python CGI with MySQL

## Introduction

## This worksheet is designed to encourage you to perform MySQL database operation using Python CGI.

Your tutor will help you by demonstrating what is required and also answer any questions you may have. After completing these exercises you should be able to develop web based database applications.

**Pre-requisite:**

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

2. You have downloaded and installed mysql.connector (see last practical)

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.

6. You have practiced MySQL with Python

**Post-requisite:** You should stop apache server and mysql service 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 PythonCGIMySQL.zip in webprog folder. There are many Python CGI and MySQL examples included in the extracted folder. Read and understand all programs, run them in web browser (for testing/debugging use IDLE or terminal window). If anything is not clear then ask!

**Objective here is to learn basic of python mysql to develop basic web based 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 and mysql.

**Exercise 1: All example code is based on the animals and spotted tables you created in previous practical i.e.**

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

Now there are following programs in the folder:

***dbfunc.py***

***animalSpotter\_simple.py and recordAnimals\_simple.py 🡪*** *spots and records animals sightings*

***animalSpotter.py and recordAnimals.py*** *🡪 Spots and records animal sightings (use of functions)*

***animalSpotterMass.py and recordAnimalsMass.py*** *🡪 Spots and records multiple animals sightings*

***animalSpotterUPD.py and recordAnimalsUPD.py*** *🡪 Spots and updates animal in spotted table*

***animalSpotterUPDINS.py and recordAnimalsUPDINS.py*** *🡪 Spots and records animal, updates if*

*animal already exists in spotted table*

***dumpsvar.py*** *🡪 for debugging/testing*

***selectAsTable.py***

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 web browser. Make sure xampp’s apache and mysql services are running

**Note:** Make sure you establish database connection using correct parameter values e.g. password and don’t forget to use correct shebang line.

**Exercise 2: Create a log table to record any actions performed on webprog database**

*In this exercise create a log of all actions which resulted in changing database records for animals and spotted tables. This means:*

1. *Create a log table with 5 attributes using phpmyadmin.*
2. *These attributes are: actiontype (e.g. insert, update, delete), tablename (table name on which action is performed), dateandtime (when was action performed), old value (what was original attribute(s) value(s)?) and new value (what is new attribute(s) value(s)?). For example if you update spotted table for animal KingFisher with new count value 20 (old value was 11) then your log table will look like something:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***actiontype*** | ***tablename*** | ***dateandtime*** | ***oldvalue*** | ***newvalue*** |
| *update* | *spotted* | *2-Nov-15 15:06* | *KingFisher, 11* | *KingFisher, 20* |

1. *Modify recordAnimals.py in record log entries in the log table.*
2. *Create a new webpage using Python CGI and display log table.*
3. If step 2 and 3 are completed then for more practice repeat them for recordAnimalsUPD.py, recordAnimalsUPDINS.py and recordAnimalsMass.py.