

ACIT 3910 Database Administration and Management Assignment #1

ACIT 3910 Database Administration and Management
Assignment #1

Introduction:

For this assignment you will be using Python to connect to a MySQL database to add to and modify the existing word dictionary from your lab. This assignment expands upon the *Connect Python* and the *Import non SQL Data* Labs.

Objectives:

Your short Python program will do the following:

1. Prompt the user on the command line for the following connection parameters (in order):
 - a. Hostname:
Hostname, if you wish to connect to MySQL running on your VM, will be `localhost`.
 - b. Port:
The default port for MySQL is `3306`.
 - c. Username:
You can use the `root` user to connect.
 - d. Password:
Your password for the `root` account.

IMPORTANT: Your program must prompt for these values *in the correct order*:

Host, Port, User, Password.

Do not mix up the order! (ex: Host, User, Password, Port is **not** allowed!)

2. With the connection parameters, attempt to connect to MySQL:
If the connection was unsuccessful print "Could not login to host with user/password provided."
and exit the program.
If the connection was successful, print "Connected Successfully." and continue to next step.
Hint: You may want to consider using exceptions (try: except:) and the exit() function.
3. Once connected to the host and port provided, print to the screen the version of MySQL.
4. Prompt the user for a word.
5. Run a query to check to see if the word provided by the user exists in the `dictionary.word` table.
The `dictionary.word` table should already be created and populated with all the words from the English dictionary as described in the *Import non SQL Data* instructions.
 - a. If the word exists in the `dictionary.word` table, prompt the user to change the word to a new word. Update the database to change the old word to the new word.
 - b. If the word doesn't exist in the `dictionary.word` table, add the word to the table.

For all queries to the database ensure you are properly escaping your inputs to prevent SQL Injection Attacks. For examples in python: <https://realpython.com/prevent-python-sql-injection/>

Save your Python program in a text file called `add_a_word.py`.

THOROUGHLY TEST your program to ensure it does what you expect it to do.

Refer to the Helpful Links section for documentation on Python and MySQL Connection Tips.

ACIT 3910 Database Administration and Management
Assignment #1

Examples of the output when the program is run:

Your code will be run on the terminal from within Ubuntu Linux using Python.

Ensure your code runs! If there are too many errors for the code to run properly, you will get a grade of 0!

1. Invalid Host/Port, User/Password combination:

```
user@ubuntu-VirtualBox:~/Documents$ python3 add_a_word.py
What host to connect to (localhost): localhost
What port to connect to (3306): 3306
What user to connect with (root): root
What password to connect with: ThisIsNotTheCorrectPassword
Could not login to host with user/password provided.
Exiting.
user@ubuntu-VirtualBox:~/Documents$
```

2. Correct Host/Port, User/Password
Word `selfie` doesn't exist in the database:

```
user@ubuntu-VirtualBox:~/Documents$ python3 add_a_word.py
What host to connect to (localhost): localhost
What port to connect to (3306): 3306
What user to connect with (root): root
What password to connect with: 
Connected successfully.
('version', '8.0.12')
What word do you want to add/change: selfie
The word 'selfie' was not found... adding
Added 'selfie' to the database
user@ubuntu-VirtualBox:~/Documents$
```

3. Correct Host/Port, User/Password
Word `hangry` doesn't exist in the database, either:

```
user@ubuntu-VirtualBox:~/Documents$ python3 add_a_word.py
What host to connect to (localhost): localhost
What port to connect to (3306): 3306
What user to connect with (root): root
What password to connect with: 
Connected successfully.
('version', '8.0.12')
What word do you want to add/change: hangry
The word 'hangry' was not found... adding
Added 'hangry' to the database
user@ubuntu-VirtualBox:~/Documents$
```

ACIT 3910 Database Administration and Management
Assignment #1

4. Correct Host/Port, User/Password

Word apple exists in the database – prompt to change to apples:

```
user@ubuntu-VirtualBox:~/Documents$ python3 add_a_word.py
What host to connect to (localhost): localhost
What port to connect to (3306): 3306
What user to connect with (root): root
What password to connect with: 
Connected successfully.
('version', '8.0.12')
What word do you want to add/change: apple
Found 'apple' in the database.
('apple')
Change 'apple' to: apples
Changed 'apple' to 'apples' in the database
user@ubuntu-VirtualBox:~/Documents$
```

5. Correct Host/Port, User/Password

Change apples back to apple:

```
user@ubuntu-VirtualBox:~/Documents$ python3 add_a_word.py
What host to connect to (localhost): localhost
What port to connect to (3306): 3306
What user to connect with (root): root
What password to connect with: 
Connected successfully.
('version', '8.0.12')
What word do you want to add/change: apples
Found 'apples' in the database.
('apples')
Change 'apples' to: apple
Changed 'apples' to 'apple' in the database
user@ubuntu-VirtualBox:~/Documents$
```

ACIT 3910 Database Administration and Management
Assignment #1

Marking Guide:

Criteria	Marks
Prompts for and uses Host, Port, User, Password to connect to database. IMPORTANT: You must prompt them in this order.	2 marks
Prints "Connected Successfully." on success and Prints "Could not login to host with user/password provided." and exits the program on invalid host and/or credentials.	2 marks
Prints the MySQL Version of the host.	2 marks
Properly adds the word prompted for on the console to the database (if it didn't exist).	2 marks
Properly changes the word to a new word prompted for on the console to the database (if the word did exist).	2 marks
Queries are written to prevent SQL Injection Attacks	2 marks
Note: You will receive a mark of 0 if your code doesn't run (i.e. too many errors to run properly)!	
Total:	12 marks

ACIT 3910 Database Administration and Management
Assignment #1

Helpful Links:

Console Python (print and input):

<https://www.dotnetperls.com/console-python>

Connect to MySQL from Python:

https://www.w3schools.com/python/python_mysql_getstarted.asp

<https://dev.mysql.com/doc/connector-python/en/connector-python-api-mysqldcursor.html>

Python MySQL Connection Exceptions:

<https://dev.mysql.com/doc/connector-python/en/connector-python-api-errors-error.html>

Python MySQL Selects (**Prevent SQL Injection Attacks**):

https://www.w3schools.com/python/python_mysql_where.asp

Python MySQL Inserts:

https://www.w3schools.com/python/python_mysql_insert.asp

Python MySQL Updates:

https://www.w3schools.com/python/python_mysql_update.asp

Get number of rows returned from a query in Python:

<https://dev.mysql.com/doc/connector-python/en/connector-python-api-mysqldcursor-rowcount.html>

Preventing SQL Injection Attacks in Python:

<https://realpython.com/prevent-python-sql-injection/>

<https://blog.sqreen.com/preventing-sql-injections-in-python/>