

## Python with DATABASE COMMUNICATION

- Programming languages are good at processing the data. But they can not store the data in permanent manner.
- Because programming languages memory will be allocated in the RAM.
- RAM is volatile, i.e , once the program execution is over , the data which is present in side the RAM will be deleted.
- In order to store the data in permanent manner, we can use the files or databases.
- Databases are nothing but softwares which stores the data in hard disk in permanent manner and provides the security for the data.
- Different data bases are,
  1. Oracle
  2. MySQL
  3. MySQLserver,DB2 ...
  4. MongodbEtc....
- We need to download any one database and install it by following below links.

### **MySQL Software Download Link:**

<https://dev.mysql.com/downloads/installer/>

### **MySQL Software Installation Guidelines Steps:**

<https://www.sqlshack.com/how-to-install-mysql-database-server-8-0-19-on-windows-10/>

- After downloading and installing then we will get **MySQL Command Line Client** prompt and **MySQL Workbench** GUI tool in our machine.
- Open any one and executes SQL Queries according to our requirement.  
mysql> SQL Queries ;

## What is pymysql?

"pymysql" is an interface/module for connecting to a MySQL database server from Python.

## How do I Install pymysql ?

Before proceeding, you make sure you have MySQLdb installed on your machine.

Just type the following in your Python script and execute it

```
>>> import pymysql
>>>
```

### Establishing the connection with pymysql :

- *Open command prompt and install our required module by using PIP command.*

```
cmd> pip install pymysql
```

- **Create python script.py file**

```
import pymysql
db = pymysql.connect( host='localhost', user='root', password='root',
db='pythondb')
```

### Connect Python to MySQLdb :

1. import required db module  
import pymysql

2. open db connection  
connection\_obj = pymysql.connect(  
host='localhost',  
user= 'root',  
password ='root',  
db='mixins\_db'  
)

3. prepare a cursor object using cursor() method by using connection object.  
cursor\_obj = connection\_obj.cursor()

4. create required sql queries.  
sql = "select \* from emp;"  
sql = "select VERSION()"

5. execute sql query using execute() method  
cursor\_obj = cursor\_obj.execute(sql)

6. Fetch a single row using fetchone() method

```
data = cursor_obj.fetchone()
print(data)
print("db version : %s " % data)
```

7. Disconnect from cursor object

```
cursor_obj.close()
```

8. close the db connection.

```
connection_db.close()
```

## STEPS TO DEVELOP :

STEP 1. In order to execute the sql queries, first we have to get the cursor object

---> we can get the cursor object by calling cursor() method of connection object.

```
cursor_obj = con_obj.cursor()
```

---> After getting the cursor object, we can execute the sql queries by calling execute method of cursor object

```
cursor_obj.execute('select*from dept')
```

----> we can display the records from the cursor object by using for loop.

```
for row in cursor_obj :
```

```
    print(row)-->(---,----,-----)
```

----> cursor\_obj contains tuple of tuple data

----> Finally we need to close all the opened objects.

```
cursor_obj.close()
```

```
conn_obj.close()
```

---->> To accessing the data from cursor object then we have some special methods like below,

**fetchone()**

---->> This function is used to fetch the first record from the table.

It returns direct tuple only.

**output:** (201, 'Vivo', 20000.0, 'black')

**fetchmany(n)**

---> This function is used to fetch the "n" number of records from the table.

--->> Here "N" optional , if not provided then it returns only first line  
**fetchall()**

-----> This function is used to fetch the total number of records from the table.  
It returns in the form tuple of tuple objects.

**output :** ( (10 , 'Vivo', 1000.0, 'black' ), (), () )

## MySQL Database Commands

TO show list of databases :

mysql > **show databases;**

3 databases come by defaultly

1. information\_schema
2. mysql
3. test

--> create user defined database.

**mysql> create database mydb;**

---> select required database name as our current database name by using the  
"use" command.

**mysql > use mydb;** ==> db changed

---->> To see available tables in "mydb" database then use below command

**mysql> show tables;**

---> creating a table in "mydb"

**mysql> create table emp(eid int, ename varchar(10), loc varchar(10), sal int);**

---> To see table description then use "desc" command.

**mysql> desc emp;**

---> To Inserting data into the table then use "insert into table\_name" command.

**mysql> insert into emp(eid,ename,loc,sal) values(1,'Ram','hyd',1000)**

----> If you want to read the data from existing table then use "select" command.

**mysql> select \* from employee where eid = 1;**

----> If you want to Update data into existing table then use "update" command.

**mysql> update emp set ename = 'ravi' where eid = 1;**

----> If you want to Delete data from existing table then use "delete" command.

**mysql> delete from emp where eid = 1;**

----> If you want to delete the table if existing then use "drop" command.

mysql> drop table if exists emp;

----> If you want to read the all data from existing table then use "select" command.

mysql> select \* from emp;

---> If you want to see|show the default db

mysql> select database();

---> If you want to displays all available tables from given database

mysql> show tables();

---> If you want to displays only particular fields of all the records

mysql> select eid,ename from emp;

---->> to display only specific number of records,

mysql> select \* from emp limit 4;

---->> to display only specific number of records from secific index onwords,

mysql> select \* from emp limit 4 offset 2;

**Q. How do I clear the screen in MySQL in windows?**

--->> Use the command "system cls" and this will clear the MySQL command line window in Windows.

mysql> system cls;

--->> There is also shorthand for the system command which is simply "\!" and would look like

mysql> \! cls

**Explanation:**

--->> \! is used to execute system shell commands

--->> cls is a command to clear the Windows command prompt screen

--->> you can easily clear the screen using just press ( ctrl + down arrow ) until you reach to top... happy coddng...

**Note:** If you want to close MySQL command line screen directly then press

mysql> exit --->> then press "Enter" button

--->> press Ctrl + Z --->> then press "Enter" button

mysql> Ctrl + Z

**CURD(Create – Update - Read - Delete ):**

Examples:

**Q. How to create the MySQL connection object using Python script?**

```
import pymysql
connection_obj = pymysql.connect(host='localhost',user='root',password='root')
print(connection_obj)
```

**Q. How to create the MySQL cursor object using Python script?**

```
import pymysql
connection_obj = pymysql.connect(host='localhost',user='root',password='root')
cursor_obj = connection_obj.cursor()
print(cursor_obj)
```

**Q. How to show all available databases from MySQL DB using Python script?**

```
import pymysql
connection_obj = pymysql.connect(host='localhost',user='root',password='root')
cursor_obj = connection_obj.cursor()
count = cursor_obj.execute('show databases')
print(count) # returns count of total dbs
# displays all dbs from cursor_obj
for db in cursor_obj:
    print(db)
```

**Q. How to create a new database using Python script?**

```
import pymysql
connection_obj = pymysql.connect(host='localhost',user='root',password='root')
cursor_obj = connection_obj.cursor()
sql = cursor_obj.execute('create database python_db2')
print(sql) # 1 returns
print('database created successfully.')
```

**Note :** If database already created then we will get error like bellow.

```
pymysql.err.ProgrammingError: (1007, "Can't create database 'python_db2';
database exists")
```

Note: sql = cursor\_obj.execute('create database if not exists python\_db2')

