import MySQLdb

# Open database connection

db = MySQLdb.connect("localhost","testuser","test123","TESTDB" )

# prepare a cursor object using *cursor()* method

cursor = db.cursor()

cursor.execute("SELECT VERSION()")

data = cursor.fetchone()

print "Database version : %s " % data

# disconnect from server

db.close()

CREATE TABLE

import MySQLdb

# Open database connection

db = MySQLdb.connect("localhost","testuser","test123","TESTDB" )

# prepare a cursor object using *cursor()* method

cursor = db.cursor()

cursor.execute("DROP TABLE IF EXISTS EMPLOYEE")

sql = "CREATE TABLE EMPLOYEE (

FIRST\_NAME CHAR(20) NOT NULL,

LAST\_NAME CHAR(20),

AGE INT,

SEX CHAR(1),

INCOME FLOAT )"

cursor.execute(sql)

# disconnect from server

db.close()

INSERT

import MySQLdb

db = MySQLdb.connect("localhost","testuser","test123","TESTDB" )

cursor = db.cursor()

sql = "INSERT INTO EMPLOYEE(FIRST\_NAME,

LAST\_NAME, AGE, SEX, INCOME)

VALUES ('Mac', 'Mohan', 20, 'M', 2000)"

try:

cursor.execute(sql)

db.commit()

except:

db.rollback()

db.close()

ANOTHER WAY

sql = "INSERT INTO EMPLOYEE(FIRST\_NAME, \

LAST\_NAME, AGE, SEX, INCOME) \

VALUES ('%s', '%s', '%d', '%c', '%d' )" % \

('Mac', 'Mohan', 20, 'M', 2000)

UPDATE

sql = "UPDATE EMPLOYEE SET AGE = AGE + 1

WHERE SEX = '%c'" % ('M')

DELETE

sql = "DELETE FROM EMPLOYEE WHERE AGE > '%d'" % (20)

READ Operation

* **fetchone()** − It fetches the next row of a query result set. A result set is an object that is returned when a cursor object is used to query a table.
* **fetchall()** − It fetches all the rows in a result set. If some rows have already been extracted from the result set, then it retrieves the remaining rows from the result set.
* **rowcount** − This is a read-only attribute and returns the number of rows that were affected by an execute() method.

import MySQLdb

db = MySQLdb.connect("localhost","testuser","test123","TESTDB" )

cursor = db.cursor()

sql = "SELECT \* FROM EMPLOYEE \

WHERE INCOME > '%d'" % (1000)

try:

cursor.execute(sql)

# Fetch all the rows in a list of lists.

results = cursor.fetchall()

for row in results:

fname = row[0]

lname = row[1]

age = row[2]

sex = row[3]

income = row[4]

# Now print fetched result

print "fname=%s,lname=%s,age=%d,sex=%s,income=%d" % \

(fname, lname, age, sex, income )

except:

print "Error: unable to fecth data"

# disconnect from server

db.close()

Output

fname=Mac, lname=Mohan, age=20, sex=M, income=2000

import sqlite3

connection = sqlite3.connect("survey.db")

cursor = connection.cursor()

cursor.execute("SELECT Site.lat, Site.long FROM Site;")

results = cursor.fetchall()

for r in results:

print(r)

cursor.close()

connection.close()

Output

(-49.85, -128.57)

(-47.15, -126.72)

(-48.87, -123.4)

records[0][0] es -49.85

**ADICIONAL**

print("Rows: ", cursor.rowcount)

|  |  |
| --- | --- |
| for row in records:  id = row[0]  name = row[1]  price = row[2] | for row in records:  id = row["Id"]  name = row["Name"]  price = row["Price"] |

Si precio viene en string y lo desea en float

price = float(record[2])