**Write python program to understand different types of Exceptions**

i=1

while i<=5:

n=int(input("Please enter numbers between 1 to 5 to see diffrent Exceptions : "))

if n==1:

try:

a=int(input("Please enter number a : "))

b=int(input("Please enter number b (put b=0): "))

c=a/b

except ZeroDivisionError:

print("Oops! Number Divisible by Zero Exception Occurs.")

else:

print "Division is",c

elif n==2:

try:

a=int(input("Please enter number a : "))

b=int(input("Please enter number b (put b='a'): "))

c=a/b

except ValueError:

print("Oops! Value Error Exception Occurs.Please enter a valid number.")

else:

print "Division is",c

elif n==3:

try:

a=int(input("Please enter number a : "))

b=int(input("Please enter number b : "))

c=k/b

except NameError:

print("Oops! Name Error Exception Occurs due to c=k/b (k is not defined ).Please enter a

valid variable number.")

elif n==4:

try:

r='2'+2

except TypeError:

print("Oops! Type Error Exception Occurs (due to '2'+2).Please Provide Valid data type. ")

elif n==5:

try:

n=int(input("Please enter Numbers between 2 to 3: (Check for other nos) "))

assert n>=2 and n<=3

print("The Number Entered is",n)

except AssertionError:

print("Oops! Assertion Error Occurs..Please enter number between 2 to 5.")

else:

print "Existing The Program"

exit()

i+=1

**Write python programs to understand CRUD Operations using Mysql Python**

Database Connectivity

Steps to Perform CRUD Operations Python Mysql Database Connectivity

**Step 1: Install Mysql in Ubuntu**

First you must install a MySQL driver, use the specific installation method below

pythonlab@ubuntu:~$ sudo apt-get install mysql-server

pythonlab@ubuntu:~$sudo apt-get install python-mysqldb

pythonlab@ubuntu:~$ sudo apt-get install python-pip python-dev libmysqlclient-dev

**Step 2: Setup the database**

Make sure you have database access, from the command line type:

pythonlab@ubuntu:~$ mysql –u root -p

MySQL will then ask your password put the password ‘python’.

Execute the following commands to create database TESTDB, Create table EMPLOYEE and

Insert Records using following commands:

mysql> create database TESTDB

-> ;

Query OK, 1 row affected (0.08 sec)

mysql> USE TESTDB

Database changed

**Step 3:Run the Python Program**

---------------------------------------Python Program----------------------------------------------------

#Mysql CRUD Operations using Python

import MySQLdb

# Open database connection

db = MySQLdb.connect("localhost","root","python","TESTDB" )

# prepare a cursor object using cursor() method

cursor = db.cursor()

print "----------Insert record----------------"

# Prepare SQL query to INSERT a record into the database.

a=raw\_input('Enter u r First name: ')

b=raw\_input('Enter u r Last name: ')

c=int(input('Enter u r Age: '))

d=raw\_input('Enter u r Sex: ')

e=int(input('Enter u r Income: '))

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

LAST\_NAME, AGE, SEX, INCOME) \

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

(a, b, c, d, e)

try:

# Execute the SQL command

cursor.execute(sql)

# Commit your changes in the database

db.commit()

except:

# Rollback in case there is any error

db.rollback()

print "----------View records-----------"

sql = "SELECT \* FROM EMPLOYEE"

try:

# Execute the SQL command

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"

print "----------Update records with age+1 For males-----------"

# Prepare SQL query to UPDATE required records

sql = "UPDATE EMPLOYEE SET AGE=AGE+1 WHERE SEX = '%c'" % ('M')

try:

# Execute the SQL command

cursor.execute(sql)

# Commit your changes in the database

db.commit()

except:

# Rollback in case there is any error

db.rollback()

print "----------Delete records with age > 30-----------"

sql = "DELETE FROM EMPLOYEE WHERE AGE > '%d'" % (30)

try:

# Execute the SQL command

cursor.execute(sql)

# Commit your changes in the database

db.commit()

except:

# Rollback in case there is any error

db.rollback()

print "----------View records-----------"

sql = "SELECT \* FROM EMPLOYEE"

try:

# Execute the SQL command

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()

**DB connectivity using SQLite3**

-------------------------------DB conn--------------------------------

import sqlite3

#db=MySQLdb.connect("localhost","root",

conn = sqlite3.connect('test.db')

print "db opened success"

conn.execute('''CREATE TABLE COMPANY

(ID INT PRIMARY KEY NOT NULL,

NAME TEXT NOT NULL,

AGE INT NOT NULL,

ADDRESS CHAR(50),

SALARY REAL);''')

print "Table created successfully";

conn.close()

------------------------------------------------DBDelete-----------------------------

import sqlite3

conn = sqlite3.connect('test.db')

print "Opened database successfully";

conn.execute("DELETE from COMPANY where ID = 2;")

conn.commit()

print "Total number of rows deleted :", conn.total\_changes

cursor = conn.execute("SELECT id, name, address, salary from COMPANY")

for row in cursor:

print "ID = ", row[0]

print "NAME = ", row[1]

print "ADDRESS = ", row[2]

print "SALARY = ", row[3], "\n"

print "Operation done successfully";

conn.close()

------------------------------------------------------------DBinsert-----------------------------------

import sqlite3

conn = sqlite3.connect('test.db')

print "Opened database successfully";

conn.execute("INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY) \

VALUES (1, 'Paul', 32, 'California', 20000.00 )");

conn.execute("INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY) \

VALUES (2, 'Allen', 25, 'Texas', 15000.00 )");

conn.execute("INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY) \

VALUES (3, 'Teddy', 23, 'Norway', 20000.00 )");

conn.execute("INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY) \

VALUES (4, 'Mark', 25, 'Rich-Mond ', 65000.00 )");

conn.commit()

print "Records created successfully";

conn.close()

-------------------------------------------------------DBselect----------------------------------

import sqlite3

conn = sqlite3.connect('test.db')

print "Opened database successfully";

cursor = conn.execute("SELECT id, name, address, salary from COMPANY")

for row in cursor:

print "ID = ", row[0]

print "NAME = ", row[1]

print "ADDRESS = ", row[2]

print "SALARY = ", row[3], "\n"

print "Operation done successfully";

conn.close()

---------------------------------------------------DBupdate-------------------------------------------

import sqlite3

conn = sqlite3.connect('test.db')

print "Opened database successfully";

conn.execute("UPDATE COMPANY set SALARY = 25000.00 where ID = 1")

conn.commit

print "Total number of rows updated :", conn.total\_changes

cursor = conn.execute("SELECT id, name, address, salary from COMPANY")

for row in cursor:

print "ID = ", row[0]

print "NAME = ", row[1]

print "ADDRESS = ", row[2]

print "SALARY = ", row[3], "\n"

print "Operation done successfully";

conn.close()

**Write python programs to understand Client-Server programming using TCP**

**Socket**

---------------------------------------------tcpserver.py----------------------------------------------------------

import socket

host='127.0.0.1'

port=8000

#Create server side socket

s=socket.socket()

s.bind((host,port))

print "Server is Waiting........."

#Allow max clients connections=1

s.listen(1)

# wait till client connects

c,addr=s.accept()

print "A Client is connected"

#Server runs continiously

while True:

#receive 1024 byte data from client

data=c.recv(1024)

# if client send empty string then come out

if not data:

break

print "From Client: "+str(data.decode())

# Enter response from server

data1=raw\_input("From Server : ")

# Send data to the client

c.send(data1.encode())

#Close Connection

c.close()

--------------------------------------------------tcpclient.py----------------------------------------------------------

import socket

host='127.0.0.1'

port=8000

#Create client side socket

s=socket.socket()

s.connect((host,port))

# Enter message at Client side

str1=raw\_input("Enter your message : ")

# Continue till client exits

while str1!='exit':

# Send data from client to server

s.send(str1.encode())

#receive 1024 byte data from client

data=s.recv(1024)

data1=data.decode()

print "From server :",data1

#Enter data

str1=raw\_input("Enter data : ")

#Close Connection

s.close()

**Write python programs to understand Client-Server programming using UDP**

**Socket**

---------------------------------------udpserver.py--------------------------------------------------------

import socket

import time

host='localhost'

port=5000

#create Socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

print "Server is Waiting ........."

time.sleep(5)

#Send message to client

s.sendto(b"Hello Client,..How r u ??",(host,port))

msg="Bye"

s.sendto(msg.encode(),(host,port))

s.close()

---------------------------------------udpclient.py---------------------------------------------------------

import socket

host='localhost'

port=5000

#create client Socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

#Connect to server using hostname and port

s.bind((host,port))

#receive a message

msg, addr=s.recvfrom(1024)

try:

#block socket for 5 seconds for sync

s.settimeout(5)

#repeat till msg get empty

while msg:

print'Received from Server:',msg.decode()

msg, addr=s.recvfrom(1024)

except socket.timeout:

print"Time is over hence Terminating...."

s.close()