**KENDRIYA VIDYALAYA GOPALPUR MILITARY STATION**

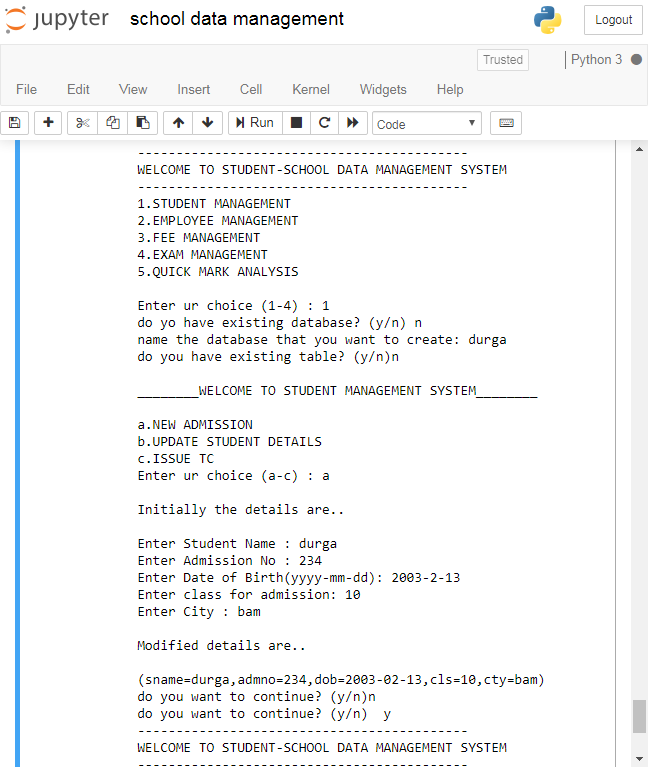
**COMPUTER SCIENCE PROJECT WORK**

**ON THE TOPIC-**

**STUDENT–SCHOOL DATA MANAGEMENT SYSTEM**

**USING-**

1. **PYTHON3 (JUPYTER)**
2. **MYSQL DATABASE 8.0**
3. **MYSQL-PYTHON CONNECTOR**



**MADE BY-**

**DURGA PRATAP BEHERA**

**12TH SCI (2019-20)**

**ROLL NO-25**

**DBMS: MySQL**

**Host : localhost**

**User: kurare**

**Password: student**

Source code:

import mysql.connector

import matplotlib.pyplot as plt

import math

import numpy as np

def selection():

print('-------------------------------------------\nWELCOME TO STUDENT-SCHOOL DATA MANAGEMENT SYSTEM\n-------------------------------------------')

print("1.STUDENT MANAGEMENT")

print("2.EMPLOYEE MANAGEMENT")

print("3.FEE MANAGEMENT")

print("4.EXAM MANAGEMENT")

print('5.QUICK MARK ANALYSIS')

ch=int(input("\nEnter ur choice (1-4) : "))

check=input("do yo have existing database? (y/n) ")

global edu

if check=="n" and ch!=5:

edu=str(input("name the database that you want to create: "))

a='create database '

a=a+edu

b='use '

b=b+edu

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student")

cursor = db.cursor()

cursor.execute(a)

cursor.execute(b)

elif check=="y" and ch!=5:

edu=str(input("name the database that you want to enter: "))

print("going into database........")

b='use '

b=b+edu

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student")

cursor = db.cursor()

cursor.execute(b)

if ch==1:

dat=input("do you have existing table? (y/n)")

if dat=='n':

tbl='create table student(sname varchar(30) not null,admno int primary key,dob date,cls char(2),cty varchar(20))'

cursor.execute(tbl)

db.close()

s="y"

while s=='y':

print('\n\_\_\_\_\_\_\_\_WELCOME TO STUDENT MANAGEMENT SYSTEM\_\_\_\_\_\_\_\_\n')

print('a.NEW ADMISSION')

print('b.UPDATE STUDENT DETAILS')

print('c.ISSUE TC')

c=input("Enter ur choice (a-c) : ")

print('\nInitially the details are..\n')

display1()

if c=='a':

insert1()

print('\nModified details are..\n')

display1()

elif c=='b':

update1()

print('\nModified details are..\n')

display1()

elif c=='c':

delete1()

print('\nModified details are..\n')

display1()

else:

print('Enter correct choice...!!')

s=input("do you want to continue? (y/n)")

elif ch==2:

dat=input("do you have existing table? (y/n)")

if dat=='n':

tbl='create table emp(empno integer primary key,ename varchar(20),job varchar(20),hiredate date)'

cursor.execute(tbl)

db.close()

s="y"

while s=='y':

print('\n\_\_\_\_\_\_\_\_WELCOME TO EMPLOYEE MANAGEMENT SYSTEM\_\_\_\_\_\_\_\_\n')

print('a.NEW EMPLOYEE')

print('b.UPDATE STAFF DETAILS')

print('c.DELETE EMPLOYEE')

c=input("Enter ur choice : ")

if c=='a':

insert2()

print('\nModified details are..\n')

display2()

elif c=='b':

update2()

print('\nModified details are..\n')

display2()

elif c=='c':

delete2()

print('\nModified details are..\n')

display2()

else:

print('Enter correct choice...!!')

s=input("do you want to continue? (y/n)")

elif ch==3:

dat=input("do you have existing table? (y/n)")

if dat=='n':

tbl='create table fee(admno integer,fee integer,month varchar(15))'

cursor.execute(tbl)

db.close()

s="y"

while s=='y':

print('\n\_\_\_\_\_\_\_\_WELCOME TO FEE MANAGEMENT SYSTEM\_\_\_\_\_\_\_\_\n')

print('a.NEW FEE')

print('b.UPDATE FEE')

print('c.EXEMPT FEE')

print('d.ENQUIRY')

c=input("Enter ur choice : ")

if c=='a':

insert3()

elif c=='b':

update3()

elif c=='c':

delete3()

elif c=='d':

enquiry3()

else:

print('Enter correct choice...!!')

s=input("do you want to continue? (y/n)")

elif ch==4:

dat=input("do you have existing table? (y/n)")

if dat=='n':

tbl='create table exam(sname varchar(20),admno integer,per decimal(4,2),class integer)'

cursor.execute(tbl)

db.close()

s="y"

while s=='y':

print('\n\_\_\_\_\_\_\_\_WELCOME TO EXAM MANAGEMENT SYSTEM\_\_\_\_\_\_\_\_\n')

print('a.INSERT EXAM DETAILS')

print('b.UPDATE DETAILS ')

print('c.DELETE DETAILS')

print('d.FULL MARK ANALYSIS')

c=input("Enter ur choice : ")

if c=='a':

insert4()

elif c=='b':

update4()

elif c=='c':

delete4()

elif c=='d':

full()

else:

print('Enter correct choice...!!')

s=input("do you want to continue? (y/n)")

elif ch==5:

s="y"

while s=='y':

print("\n\_\_\_\_\_\_\_\_WELCOME TO QUICK MARK ANALYSIS\_\_\_\_\_\_\_\_\n")

mkanalysis()

s=input("do you want to continue? (y/n)")

else:

print('Enter correct choice..!!')

def insert1():

sname=input("Enter Student Name : ")

admno=int(input("Enter Admission No : "))

dob=input("Enter Date of Birth(yyyy-mm-dd): ")

cls=input("Enter class for admission: ")

cty=input("Enter City : ")

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

sql="INSERT INTO student(sname,admno,dob,cls,cty) VALUES ( '%s' ,'%d','%s','%s','%s')"%(sname,admno,dob,cls,cty)

try:

cursor.execute(sql)

db.commit()

except:

db.rollback()

db.close()

def display1():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM student"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

dob=c[2]

cls=c[3]

cty=c[4]

print ("(sname=%s,admno=%d,dob=%s,cls=%s,cty=%s)" % (sname,admno,dob,cls,cty))

except:

print ("Error: unable to fetch data")

db.close()

def update1():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM student"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

dob=c[2]

cls=c[3]

cty=c[4]

print ("(sname=%s,admno=%d,dob=%s,cls=%s,cty=%s)" % (sname,admno,dob,cls,cty))

except:

print ("Error: unable to fetch data")

print()

tempst=input("Enter Admission No : ")

inp=input("enter column name to be updated: ")

temp=str(input("Enter new value : "))

try:

sql = "Update student set "

sql=sql+inp+"="+"'"+temp+"'"+" where admno="+tempst

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def delete1():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM student"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

dob=c[2]

cls=c[3]

cty=c[4]

print ("(sname=%s,admno=%d,dob=%s,cls=%s,cty=%s)" % (sname,admno,dob,cls,cty))

except:

print ("Error: unable to fetch data")

temp=int(input("\nEnter adm no to be deleted : "))

try:

sql = "delete from student where admno='%d'" % (temp)

ans=input("Are you sure you want to delete the record(y/n) : ")

if ans=='y' or ans=='Y':

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def insert2():

ename=input("Enter Employee Name : ")

empno=int(input("Enter Employee No : "))

job=input("Enter Designation: ")

hiredate=input("Enter date of joining: ")

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

sql="INSERT INTO emp(ename,empno,job,hiredate) VALUES ( '%s' ,'%d','%s','%s')"%(ename,empno,job,hiredate)

try:

cursor.execute(sql)

db.commit()

except:

db.rollback()

db.close()

def display2():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM emp"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

empno = c[0]

ename= c[1]

job=c[2]

hiredate=c[3]

print ("(empno=%d,ename=%s,job=%s,hiredate=%s)" % (empno,ename,job,hiredate))

except:

print ("Error: unable to fetch data")

db.close()

def update2():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM emp"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

empno = c[0]

ename= c[1]

job=c[2]

hiredate=c[3]

print ("(empno=%d,ename=%s,job=%s,hiredate=%s)" % (empno,ename,job,hiredate))

except:

print ("Error: unable to fetch data")

print()

tempst=input("Enter Employee No : ")

inp=input("enter column name to be updated: ")

temp=str(input("Enter new value : "))

try:

sql = "Update emp set "

sql=sql+inp+"="+"'"+temp+"'"+" where empno="+tempst

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def delete2():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM emp"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

empno = c[0]

ename= c[1]

job=c[2]

hiredate=c[3]

print ("(empno=%d,ename=%s,job=%s,hiredate=%s)" % (empno,ename,job,hiredate))

except:

print ("Error: unable to fetch data")

temp=int(input("\nEnter emp no to be deleted : "))

try:

sql = "delete from emp where empno='%d'" % (temp)

ans=input("Are you sure you want to delete the record(y/n) : ")

if ans=='y' or ans=='Y':

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def insert3():

admno=int(input("Enter adm no: "))

fee=float(input("Enter fee amount : "))

month=input("Enter Month: ")

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

sql="INSERT INTO fee(admno,fee,month) VALUES ( '%d','%d','%s')"%(admno,fee,month)

try:

cursor.execute(sql)

db.commit()

except:

db.rollback()

db.close()

def display3():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM fee"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

admno= c[0]

fee=c[1]

month=c[2]

print ("(admno=%d,fee=%s,month=%s)" % (admno,fee,month))

except:

print ("Error: unable to fetch data")

db.close()

def update3():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM fee"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

admno= c[0]

fee=c[1]

month=c[2]

print ("(admno=%d,fee=%d,month=%s)"%(admno,fee,month))

except:

print ("Error: unable to fetch data")

print()

tempst=input("Enter Admission No : ")

inp=input("enter column name to be updated: ")

temp=input("Enter new value : ")

try:

sql = "Update fee set "

sql=sql+inp+"="+"'"+temp+"'"+" where admno="+tempst

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def delete3():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM fee"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

admno= c[0]

fee=c[1]

month=c[2]

print ("(admno=%d,fee=%d,month=%s)"%(admno,fee,month))

except:

print ("Error: unable to fetch data")

temp=int(input("\nEnter adm no to be deleted : "))

try:

sql = "delete from fee where admno='%d'" % (temp)

ans=input("Are you sure you want to delete the record(y/n) : ")

if ans=='y' or ans=='Y':

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def enquiry3():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

mth=input("enter month: ")

try:

a="select \* from fee where month="

a=a+mth

cursor.execute(a)

results = cursor.fetchall()

for c in results:

admno= c[0]

fee=c[1]

month=c[2]

print ("(admno=%d,fee=%d,month=%s)"%(admno,fee,month))

except:

print ("Error: unable to fetch data")

db.close()

def insert4():

sname=input("Enter Student Name : ")

admno=int(input("Enter Admission No : "))

per=float(input("Enter percentage : "))

cls=int(input("Enter class: "))

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

sql="INSERT INTO exam(sname,admno,per,cls) VALUES ( '%s' ,'%d','%s','%d')"%(sname,admno,per,cls)

try:

cursor.execute(sql)

db.commit()

except:

db.rollback()

db.close()

def display4():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM exam"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

per=c[2]

cls=c[3]

print ("(sname=%s,admno=%d,per=%s,cls=%d)"%(sname,admno,per,cls))

except:

print ("Error: unable to fetch data")

db.close()

def update4():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM exam"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

per=c[2]

cls=c[3]

print ("(sname=%s,admno=%d,per=%s,cls=%d)"%(sname,admno,per,cls))

except:

print ("Error: unable to fetch data")

print()

tempst=input("Enter Admission No : ")

inp=input("enter column name to be updated: ")

temp=str(input("Enter new value : "))

try:

sql = "Update exam set "

sql=sql+inp+"="+"'"+temp+"'"+" where admno="+tempst

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def delete4():

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

try:

sql = "SELECT \* FROM exam"

cursor.execute(sql)

results = cursor.fetchall()

for c in results:

sname = c[0]

admno= c[1]

per=c[2]

cls=c[3]

print ("(sname=%s,admno=%d,per=%s,cls=%d)"%(sname,admno,per,cls))

except:

print ("Error: unable to fetch data")

temp=int(input("\nEnter adm no to be deleted : "))

try:

sql = "delete from exam where admno='%d'" % (temp)

ans=input("Are you sure you want to delete the record(y/n) : ")

if ans=='y' or ans=='Y':

cursor.execute(sql)

db.commit()

except Exception as e:

print (e)

db.close()

def bar(y,labels,z,t="practice test"):

x=np.arange(1,z+1)

plt.bar(x,y,width=0.25)

plt.xticks(x,labels)

plt.xlabel("subjects->")

plt.ylabel("marks->")

plt.title(t)

t=t+".png"

plt.show()

plt.savefig(t)

def mkanalysis():

print("what do you want to calculate?")

print("1.percentage")

print("2.result analysis")

a=int(input(" "))

t=input("enter name of test :")

c,b=0,0

l,m,n,o=[],[],[],[]

z=int(input("enter number of subjects "))

for i in range(z):

l1=input("enter the subject: ")

m1=int(input("enter the marks: "))

n1=int(input("out of: "))

o1=((m1/n1)\*1000)//10

l.append(l1)

m.append(m1)

n.append(n1)

o.append(o1)

c+=m1

b+=n1

d=(c/b)\*100

dic={}

lis=[]

for i in range(z):

per=((m[i]/n[i])\*1000)//10

sub=l[i]

dic[sub]=per

lis.append(per)

if a==1:

for i in dic:

print("you have secured",dic[i],"% in",i)

print("your overall % is: ",d)

bar(lis,l,z,t)

else:

for i in o:

a=o.index(i)

if int(i)<=33.0:

print("you have failed in ",l[a],":improvement is neccesary")

elif int(i)>=90:

print("you have done very well in ",l[a]," :)")

r=int(input("enter your target in final examination in %: "))

if math.fabs(d-r)>5:

print("you must give more effort!")

else:

print("keep raising your target with more effort!")

def full():

lim=input("enter minimum pass% :")

cls=input("enter class")

a="select sname,per from exam where per<"

a=a+lim+" and class="+cls

db=mysql.connector.connect(host="localhost",user="kurare",passwd="student",database=edu)

cursor = db.cursor()

cursor.execute(a)

results=cursor.fetchall()

print("failed students are : ")

for c in results:

sname = c[0]

per=c[1]

print("(sname=%s,per=%d)"%(sname,per) )

print()

db.close()

loop=True

while loop:

selection()

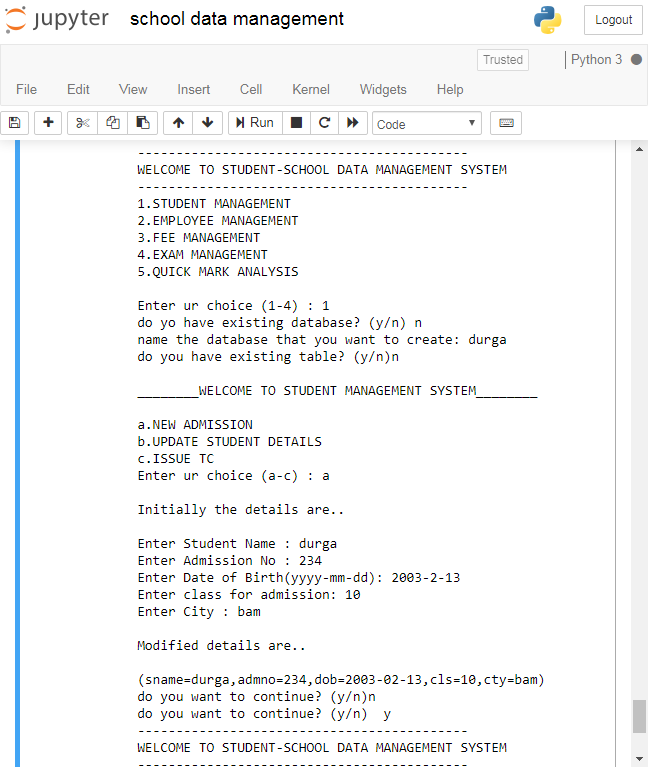
inp=input("do you want to continue? (y/n) ")

if inp=="n":

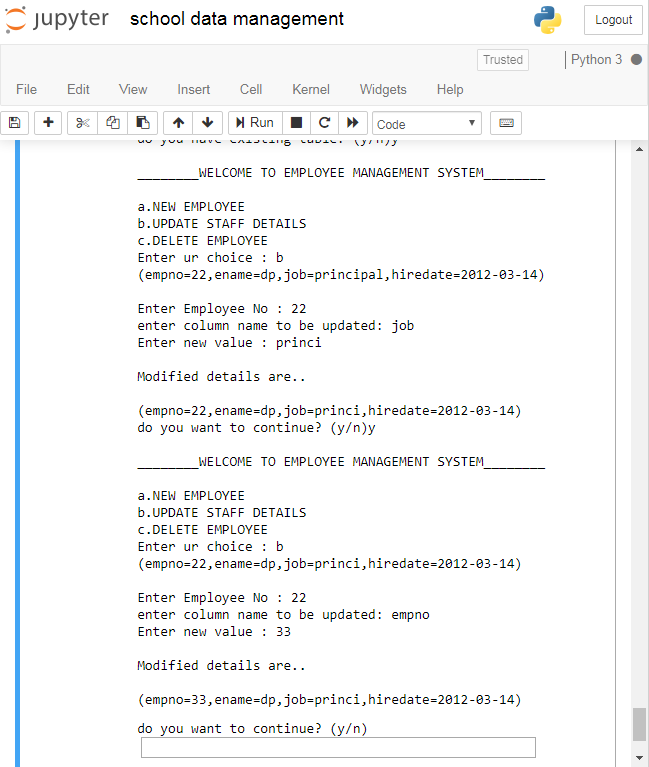
loop=False

print("thank you!! :) ")

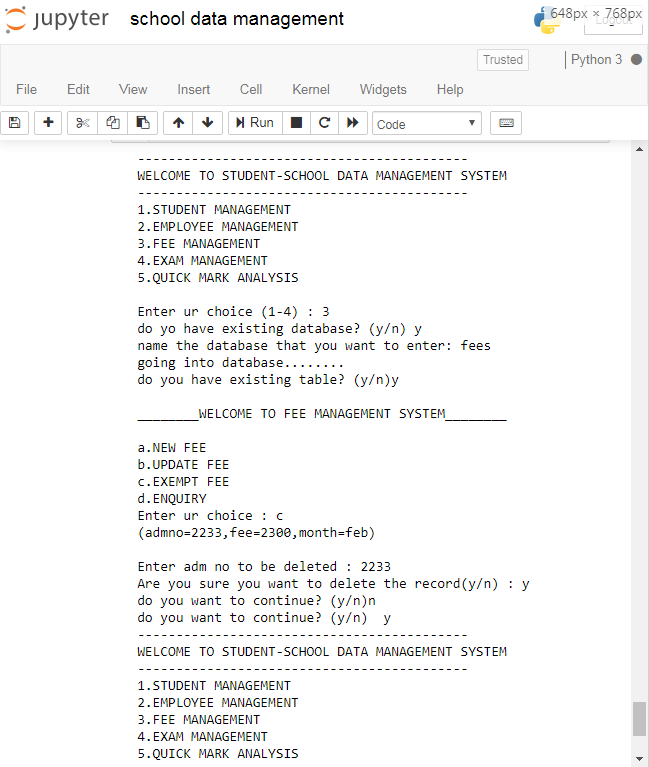
OUTPUT1:☟(INSERTING)



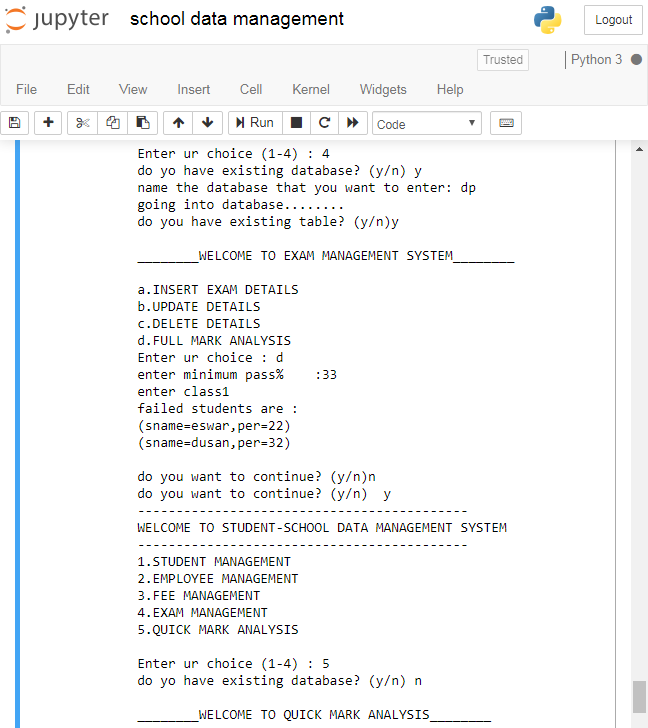
OUTPUT2:☟(UPDATING)



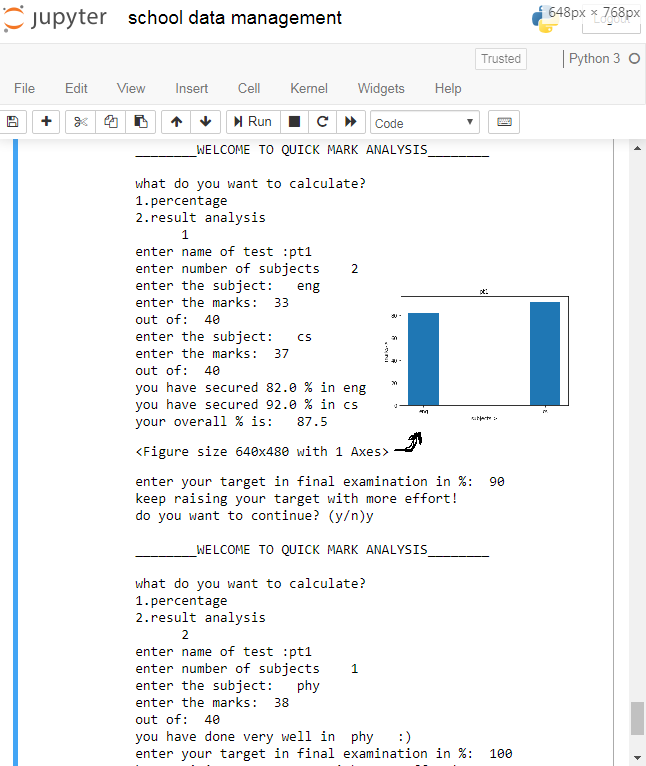
OUTPUT3:☟(DELETING)



OUTPUT4:☟(SEARCHING)



OUTPUT5:☟(ANALYSIS)



CERTIFICATE

This is to certify that Durga Pratap Behera of class XII-SCI has successfully completed the computer science project entitled,

“Student-school data management system using- interface python with SQL*”*

This work is the result of his endeavours & research in python programming and mysql queries. It is finalized under our guidance and supervision in the academic year 2019-2020.

Signature of student signature of teacher

Signature of external examiner