**Student Examination Portal**

**Submitted by**

**Name of the Students:** Apratim Haldar

**Enrolment Number:** 12022002002236

**Section:** C

**Class Roll Number:** 69  
**Stream:** CSE

**Subject:** Programming for Problem Solving with Python

**Subject Code:** IVC101

**Department:** Basic Science and Humanities

Under the supervision of

SWARNENDU GHOSH

**Academic Year: 2022-26**

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE FIRST SEMESTER

Logo, company name

Description automatically generated

**DEPARTMENT OF BASIC SCIENCE AND HUMANITITES**

**INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA**

Logo, company name

Description automatically generated

**CERTIFICATE OF RECOMMENDATION**

We hereby recommend that the project prepared under our supervision by **Upalabdha Sinha,** entitled **PYTHON DATABASE PROJECT** be accepted in partial fulfillment of the requirements for the degree of partial fulfillment of the first semester.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Head of the Department Project Supervisor

Basic Sciences and Humanities

IEM, Kolkata

# Introduction

Python is a versatile programming language which finds widespread applications in present day software development. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code. Python is a programming language that lets programmers work quickly and integrate systems more efficiently.

## Objective

The objective of this project is to display the functionality of databases in Python by creating and operating on 4 different databases using csv files and pandas dataframe which will store details of students, courses, batches and departments of a college.

## Organization of the Project

This Project is organized into the following modules---

* student.py: Performs operations on the student database which includes, updation, deletion of student records and generation of report cards.
* batch.py: Performs operations on the batch database which includes updation, deletion of batch records and generation of batch statistics.
* course.py: Performs operations on the course database which includes, updation, deletion of course records and generation of course statistics.
* department.py: Performs operations on the department database which includes, updation, deletion of department records and maintenance of batches.
* examination.py: Provides statistics of a particular examination as per user input

# Database Descriptions

This project is organized into 5 main csv files:

**i) Student.csv:** Used to store student related details. Consisting of 4 major columns: Student ID, Name, Class Roll Number, Batch ID

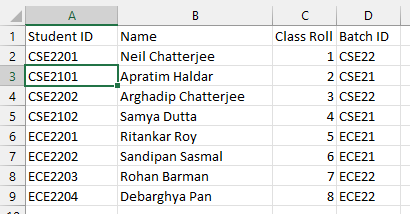
**ii) Course.py:** Used to store Course related details. Consisting of 3 major columns: Course ID, Course Name, Marks obtained

**iii) Batch.py:** Used to store batch related details. Consisting of 5 major columns: Batch ID, Batch Name, Department Name, List of Courses, List of Students

**iv) Department.csv:** Used to store department-related details. Consisting of 3 major columns: Department ID, Department Name, List of Batches

**v) Examination.py:** Used to store examination related details. Consisting of 3 major columns: Name, Class Roll, Marks

## Database Samples



Graphical user interface, text, application, table

Description automatically generated

Graphical user interface, table

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Application, table, Excel

Description automatically generated

# Data Flow and E-R Diagrams

Diagram

Description automatically generated

# Programs

Provide the python programs of the various modules.

1. **main.py**

import pandas as pd

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WELCOME TO STUDENT EXAMINATION PORTAL\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

while(1):

q=int(input("\nList of Operations.\n0.Exit Examination Portal Software\n1.View Student Database\n 5.Create new Student Detail\n 6.Update Student Database\n 7.Remove a student from the Database\n 8.Generate Student Report Card\n2.View Course Database\n 9.Create new Course\n 10.View student's performance in any Course\n 11.View Course Statistics\n3.View Batch Database\n 12.Update Batch Details\n 13.View list of all students in a batch.\n 14.View list of all courses.\n 15.View Student Performance in a Batch\n4.View Department Database.\n 16.Create new Department.\n 17.View all batches in a department\n18. Register a new examination and view statistics of the same."))

if(q==1):

print(pd.read\_csv('Student.csv'))

if(q==2):

print(pd.read\_csv('Course.csv'))

if(q==3):

print(pd.read\_csv('Batch.csv'))

if(q==4):

print(pd.read\_csv('Department.csv'))

if(q==5):

from student import creation

if(q==6):

from student import updation

if(q==7):

from student import removal

if(q==8):

from student import report

if(q==9):

from Course import creation

if(q==10):

from Course import perf

if(q==11):

from Course import stat

if(q==12):

from Batch import creation

if(q==13):

from Batch import students

if(q==14):

from Batch import courses

if(q==15):

from Batch import performance

if(q==16):

from Department import creation

if(q==17):

from Department import batches

if(q==18):

import Examination

1. **student.py**

import pandas as pd

lstn=[]

lstrno=[]

lstbch=[]

lstid=[]

lst=[]

lstm=[]

grades=[]

pf=[]

def creation():

while(1>0):

i=input("Enter student id.")

n=input("Enter Student name.")

r=int(input("Enter student roll number."))

b=input("Enter student batch name.")

lstid.append(i)

lstn.append(n)

lstrno.append(r)

lstbch.append(b)

z=int(input("Enter 1 to add more students, 2 to show updated dataframe or press 0 to exit."))

if(z==0):

break

elif(z==2):

print(df)

else:

continue

df=pd.DataFrame({'Student ID':lstid,'Name':lstn,'Roll Number':lstrno,'Batch ID':lstbch})

df.to\_csv('Student.csv', mode='a',header=False)

def updation():

while(1):

b=int(input("Choose an option\n 1. Update Student ID \n 2. Update Student name.\n 3. Update Class Roll Number.\n 4. Update Batch ID."))

if(b==1):

a=input("Enter the Student ID to be updated.")

x=input("Enter the new student id.")

df=pd.read\_csv('Student.csv')

df.loc[df['Student ID']==a, 'Student ID']=x

df.to\_csv('Student.csv')

elif(b==2):

v=input("Enter the name to be updated.")

y=input("Enter the new student name.")

df=pd.read\_csv('Student.csv')

df.loc[df['Name']==v,'Name']=y

df.to\_csv('Student.csv')

elif(b==3):

u=input("Enter the roll number to be updated.")

z=input("Enter the new roll number.")

df=pd.read\_csv('Student.csv')

df.loc[df['Class Roll Number']==u,'Class Roll Number']=z

df.to\_csv('Student.csv')

elif(b==4):

s=input("Enter the Batch ID to be updated.")

w=input("Enter the new Batch ID.")

df=pd.read\_csv('Student.csv')

df.loc[df['Batch ID']==s,'Batch ID']=w

df.to\_csv('Student.csv')

else:

print("Invalid input. Try again.")

continue

c=(int)(input("Enter 1 to update more values, 2 to show the updated dataframe and 0 to exit."))

if(c==0):

break

elif(c==2):

print(df)

else:

continue

def removal():

i=0

while(1):

d=input("Enter the Student ID to be removed.")

df=pd.read\_csv('Student.csv')

if((d in df.values)== True):

while(2):

if(df.loc[i,'Student ID']==d):

df.drop([i],axis=0,inplace=True)

break

else:

i+=1

else:

print("Invalid input.Try again")

continue

v=(int)(input("Enter 1 to update more values, 2 to show the updated dataframe and 0 to exit."))

if(v==0):

break

elif(v==2):

print(df)

else:

continue

df.set\_index('Student ID')

df.to\_csv('Student.csv',index=False)

def report():

a=input("Enter the Student ID.")

b=int(input("Enter the number of subjects."))

for i in range(0,b):

c=input("Enter subject")

lst.append(c)

d=int(input("Marks scored out of 100 = "))

lstm.append(d)

if d>=90:

grades.append("A")

elif d>=80:

grades.append("B")

elif d>=70:

grades.append("C")

elif d>=60:

grades.append("D")

elif d>=50:

grades.append("E")

else:

grades.append("F")

if d<40:

pf.append("FAILED")

else:

pf.append("PASSED")

S=pd.read\_csv("Student.csv")

S=S.set\_index('Student ID')

print("\nPERFORMACE REPORT OF ",S.at[a,'Name'])

rep=pd.DataFrame({"Subjects":lst,"Marks":lstm,"Grades":grades,"Pass/Fail Status":pf})

print(rep)

rep.to\_csv('Report.csv',index=False,sep="\t")

1. **Course.py**

import pandas as pd

import ast

import matplotlib.pyplot as plt

lstcid=[]

lstcn=[]

dictm={}

lstm=[]

lst=[]

i=1

lstv=[]

lst=[]

cdf=pd.DataFrame()

def creation():

while(1):

d=input("Enter the new course id.")

lstcid.append(d)

e=input("Enter the course name.")

lstcn.append(e)

while(1):

f=input("Enter the student id.")

g=int(input("Enter the marks obtained for this student."))

dictm[f]=g

x=int(input("Enter 1 to enter more students into the course or 0 to exit."))

if(x==0):

break

lstm.append(dictm)

y=int(input("Enter 1 to enter more courses or 0 to exit the Course Portal."))

if(y==0):

break

else:

continue

cdf=pd.DataFrame({'Course ID':lstcid,'Course Name':lstcn,'Marks obtained':lstm})

cdf.to\_csv('Course.csv', mode='a',header=False,index=False)

def perf():

df=pd.read\_csv('Course.csv')

df=df.set\_index("Course ID")

x=input("Enter the course ID for which you wish to see the students' performance.")

d=ast.literal\_eval(df.at[x,'Marks obtained'])

lstk=list(d.keys())

lstv=list(d.values())

ndf=pd.DataFrame({'Student ID':lstk,'Marks Obtained':lstv})

print(ndf)

x=[]

for i in range(0,len(lstv)):

if(lstv[i]>=90):

x.append('A')

if(lstv[i]>=80):

x.append('B')

if(lstv[i]>=70):

x.append('C')

if(lstv[i]>=60):

x.append('D')

if(lstv[i]>=50):

x.append('E')

if(lstv[i]>=40):

x.append('F')

plt.hist(x)

plt.xlabel("Grades")

plt.ylabel("Number of Students.")

plt.title("Course Statistics.")

plt.show()

1. **Batch.py**

import pandas as pd

import ast

import matplotlib.pyplot as plt

lstbd=[]

lstbn=[]

lstdn=[]

lstc=[]

lstst=[]

lst=[]

st=[]

c=[]

s=[]

bdf=pd.DataFrame()

def creation():

while(1):

d=input("Enter the new batch id.")

lstbd.append(d)

e=input("Enter the batch name.")

lstbn.append(e)

f=input("Enter the department name.")

lstdn.append(f)

while(1):

f=input("Enter course name under this batch.")

lst.append(f)

x=int(input("Enter 1 to enter more courses into the batch or 0 to exit."))

if(x==0):

break

lstc.append(lst)

while(1):

g=input("Enter students enrolled under this batch.")

s.append(g)

x=int(input("Enter 1 to enter more students into the batch or 0 to exit."))

if(x==0):

break

st.append(s)

y=int(input("Enter 1 to create more batches or 0 to exit the Batch Portal."))

if(y==0):

print (y)

break

bdf=pd.DataFrame({'Batch ID':lstbd,'Batch Name':lstbn,'Department Name':lstdn,'List of Courses':lstc,'List of Students':st})

bdf.to\_csv('Batch.csv', mode='a',header=False,index=False)

def students():

df=pd.read\_csv('Batch.csv')

df=df.set\_index('Batch ID')

while(1):

z=input("Enter the batch id.")

print("List of all students in the batch---\n", df.at[z,'List of Students'])

x=int(input("Enter 1 to view more batches or 0 to exit."))

if(x==0):

break

def courses():

df=pd.read\_csv('Batch.csv')

df=df.set\_index('Batch ID')

while(1):

z=input("Enter the batch id.")

print("List of all students in the batch---\n", df.at[z,'List of Courses'])

x=int(input("Enter 1 to view more batches or 0 to exit."))

if(x==0):

break

def performance():

a=input("Enter Batch ID.")

df=pd.read\_csv('Batch.csv')

df=df.set\_index('Batch ID')

listn=[]

listroll=[]

s=0

listp=[]

df1=pd.read\_csv("Student.csv")

df1=df1.set\_index('Student ID')

df2=pd.read\_csv("Course.csv")

df2=df2.set\_index('Course ID')

lstst=(df.at[a,'List of Students'])

w=""

for i in range(0,len(lstst)):

if(lstst[i].isalnum()):

w=w+lstst[i]

if(lstst[i]==","):

st.append(w)

w=""

if(i==(len(lstst)-1)):

st.append(w)

listc=(df.at[a,'List of Courses'])

w=""

for i in range(0,len(listc)):

if(listc[i].isalnum()):

w=w+listc[i]

if(listc[i]==","):

c.append(w)

w=""

if(i==(len(listc)-1)):

c.append(w)

for i in range(0,len(st)):

listn.append(df1.at[st[i],'Name'])

listroll.append(df1.at[st[i],'Class Roll Number'])

p=int(input("Enter percentage considering all subjects."))

listp.append(p)

print("Performance Report of Batch ", a)

df3=pd.DataFrame({'Name':st,'Class Roll':listroll,'Percentage':listp})

print(df3)

per=["(90-100)%","(80-90)%"]

a=0

b=0

C=0

d=0

e=0

f=0

for i in range(0,len(listp)):

if(listp[i]>=90):

a=a+1

if(listp[i]>=80):

b=b+1

if(listp[i]>=70):

C+=1

if(listp[i]>=60):

d+=1

if(listp[i]>=40):

e=e+1

if(listp[i]<40):

f=f+1

pi=[a,b]

plt.figure(figsize=(8,10))

plt.pie(pi,labels=per,shadow=True)

plt.legend(title="Percenatge Distribution")

plt.show()

performance()

1. **Department.py**

import pandas as pd

dn=[]

lstbh=[]

id=[]

l=[]

ddf=pd.DataFrame()

def creation():

while(1>0):

i=input("Enter Department ID.")

n=input("Enter Department name.")

while(1):

b=input("Enter batches under this department and press 0 to exit.")

id.append(i)

dn.append(n)

l.append(b)

lstbh.append(l)

z=int(input("Enter 1 to add more departments, 2 to show updated dataframe or press 0 to exit."))

if(z==0):

break

elif(z==2):

print(pd.read\_csv('Department.csv'))

else:

continue

ddf=pd.DataFrame({'Department ID':id,'Department Name':dn,'List of Batches':lstbh})

ddf.to\_csv('Department.csv', mode='a',header=False)

def batches():

df=pd.read\_csv('Department.py')

df=df.set\_index('List of Batches')

while(1):

z=input("Enter the department id.")

print("List of all Batches in the department---\n", df.at[z,'List of Batches'])

x=int(input("Enter 1 to view more batches or 0 to exit."))

if(x==0):

break

def courses():

df=pd.read\_csv('Batch.py')

df=df.set\_index('Batch ID')

while(1):

z=input("Enter the batch id.")

print("List of all students in the batch---\n", df.at[z,'List of Courses'])

x=int(input("Enter 1 to view more batches or 0 to exit."))

if(x==0):

break

1. **Examination.py**

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

listN=[]

listr=[]

listM=[]

crs=[]

listB=[]

stu=[]

col=(np.random.random(), np.random.random(),np.random.random())

while(1):

e=input("Enter the Course name.")

crs.append(e)

n=int(input("Enter the number of students who have taken this examination."))

df1=pd.read\_csv("Student.csv")

df1=df1.set\_index('Student ID')

for j in range(0,n):

a=input("Enter Student ID.")

stu.append(a)

listN.append(df1.at[a,'Name'])

listr.append(df1.at[a,'Class Roll Number'])

listB.append(df1.at[a,'Batch ID'])

b=input("Enter marks scored.")

listM.append(b)

df3=pd.DataFrame({'Name':listN,'Class Roll':listr,'Marks':listM})

df3.to\_csv('Examination.csv',index=False)

print("PERFORMANCE REPORT")

print(pd.read\_csv('Examination.csv'))

plt.scatter(listM,listN,color=col)

plt.show()

z=int(input("Enter 1 to view more course performances or 0 to exit."))

if(z==0):

break

# Outputs

.

Text

Description automatically generated

Chart

Description automatically generated

A picture containing chart

Description automatically generated

