**STUDENT EXAMINATION PORTAL**

**Submitted by**

**Name of the Student:** SARNAAVHO PAL

**Enrolment Number:** 12022002016036

**Section:** D

**Class Roll Number:** 31

**Stream:** C.S.E. (AIML)

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

**Subject Code:** IVC101

**Department:** Basic Science and Humanities

***Under the supervision of***

**DR. SWARNENDU GHOSH**

**Academic Year: 2022-26**

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE FIRST SEMESTER

**DEPARTMENT OF BASIC SCIENCE AND HUMANITITES**

**INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA**



**CERTIFICATE OF RECOMMENDATION**

We hereby recommend that the project prepared under our supervision by Name of the Student, entitled **“STUDENT EXAMINATION PORTAL”** 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:-**

If we see the present scenario we can clearly understand that every educational institutions or big companies need a system to keep a record of the data of their students and employees respectively. The best way to maintain these records is by creating separate Databases and storing the necessary data. In this project we have mainly used the PYTHON Programming Language to make a database which can be further used to store necessary data. PYTHON is an easy to understandable and user friendly language so anyone can make a program to make such data bases according to their needs.

**OBJECTIVE:-**

The main objective of this project is to develop a program for creating a database by which we can take data from the user and store it in the desired cells, Because of these project we got to learn "How to create a Database", "Relationship between several databases" , and "How to create a database using PYTHON Programming Language".

The objective of the project is to make the students data easily accessible. The project also makes the data more organisable. It also helps to find the statistical data for each enrolled student.

Also the objective of this product is to display the functionality of databases in Python by creating and operating on 4 different databases which will store details of students, courses, batches and departments of a college.

**ORGANISATION OF THE PROJECT:-**

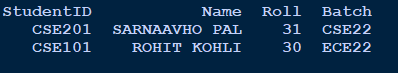
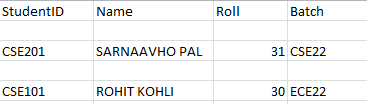
This project consists of three sections  
i) Taking data from the user: When we run the programme a few terminal prompts instruct us to give the correct input.  
ii) Storing the data into different databases: After taking the inputs from the user the code analyses data and store it in its respective databases.

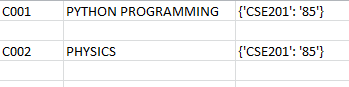
**DATABASE DESCRIPTIONS:-**

There are 4 CSV files that act as databases in this project. They are **student.csv, course.csv, batch.csv, department.csv**.

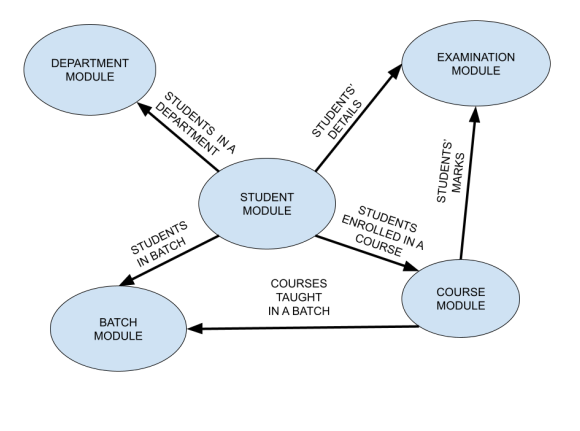
There are four databases:  
1)STUDENT: Stores details of a student  
2)COURSE: Stores details of all courses  
3)BATCH: Stores details of all courses  
4)DEPARTMENT: Stores details of all courses

**DATABASE SAMPLES:-**

student.csv

C:\Users\HP\Desktop\Screenshot 2023-01-02 224440.pngcourse.csv

**DATA FLOW AND E-R DIAGRAMS:-**



**PROGRAMS:-**

import csv

import pandas as pnd

import numpy as nmp

from matplotlib import pyplot as pplt

import time

while True:

ch=int(input("\nType:\n0.Exit\n1.Student\n2.Course\n3.Batch\n4.Department\n5.Examination\nEnter u r choice:"))

def grades(marks):

if marks>=90:

return ["A","Pass"]

elif marks>=80:

return ["B","Pass"]

elif marks>=70:

return ["C","Pass"]

elif marks>=60:

return ["D","Pass"]

elif marks>=50:

return ["E","Pass"]

else:

return ["F","Fail"]

def mrkgrd(df):

L1=[]

L2=[]

for i in df.loc[:,"Marks"]:

L=g(i)

L1.append(L[0])

L2.append(L[1])

df['Grade']=L1

df['Pass/Fail']=L2

return df

def add1(L,a):

str1=''

for i in L:

str1=str1+i+a

str1=str1[:-1]

return str1

if ch==0:

break

elif ch==1:

ch1=int(input("TYPE:\n0.Create Database\n1.input student details\n2.Update data\n3.Delete data\n4.Report card(To be generated as a text file result.txt)\nEnter u r choice:"))

if ch1==0:

L1=[['StudentID','Name','Roll','Batch']]

with open("student.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

elif ch1==1:

n=int(input("Enter the no. of entries:"))

L1=[]

for i in range(n):

L=[]

StudentID=input("Enter student id:")

Name=input("Enter name:")

Roll=input("Enter roll:")

Batch=input("Enter batch:")

L=[StudentID,Name,Roll,Batch]

L1.append(L)

with open("student.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

df=pnd.read\_csv("student.csv")

print(df)

elif ch1==3:

df=pnd.read\_csv("student.csv")

np\_ar = df.to\_numpy()

nmp.delete(np\_ar,0,axis=1)

p=input("Enter student Id :")

L=[]

for i in np\_ar:

if i[0]!=p:

print(i)

L.append(i)

df = pnd.DataFrame(L, columns = ['StudentID','Name','Roll','Batch'])

print(df)

df.to\_csv('student.csv', mode='w',index=False)

elif ch1==2:

df=pnd.read\_csv("student.csv")

np\_ar = df.to\_numpy()

nmp.delete(np\_ar,0,axis=1)

print(np\_ar)

p=input("Enter student Id :")

L=[]

for i in np\_ar:

print(i)

if i[0]!=p:

L.append(i)

else:

p1=input("1.Name\n2.Roll\n3.Batch\n")

if p1=='1':

print(i)

p2=input("Enter new name:")

i[1]=p2

print(i)

L.append(i)

elif p1=='2':

print(i)

p2=input("Enter new roll:")

i[2]=p2

print(i)

L.append(i)

elif p1=='3':

print(i)

p2=input("Enter new batch:")

i[3]=p2

print(i)

L.append(i)

print(L)

df = pnd.DataFrame(L, columns = ['StudentID','Name','Roll','Batch'])

print(df)

df.to\_csv('student.csv', mode='w',index=False)

elif ch1==4:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='StudentID')

p=mrkgrd(df)

dup\_df=p.loc[:,["StudentID","Name","Roll","CourseName","Marks","Grade","Pass/Fail"]]

with open("Result.txt", 'w') as f:

grp =dup\_df.groupby(['StudentID'])

for i,j in grp:

f.write("\n"+i+"\n")

str1 = j.to\_string(header=True, index=False)

f.write(str1)

elif ch==2:

ch2=int(input("0.Create Course database\n1.input course details\n2.View Student performance\n3.Show course histogram\nEnter u r choice:"))

if ch2==0:

L1=[['CourseID','CourseName','Marks']]

L3=[['CourseID','CourseName','Marks','StudentID']]

with open("course.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

with open("course1.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L3)

elif ch2==1:

n=int(input("Enter the no. of entries:"))

L1=[]

L3=[]

Dict={}

for i in range(n):

L=[]

CourseID=input("Enter course id:")

CourseName=input("Enter course name:")

StudentID=input("Enter student id:")

Marks=input("Enter marks:")

Dict[StudentID]=Marks

L=[CourseID,CourseName,Dict]

L2=[CourseID,CourseName,Marks,StudentID]

L1.append(L)

L3.append(L2)

with open("course.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

with open("course1.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L3)

df=pnd.read\_csv("course.csv")

print(df)

elif ch2==2:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='StudentID')

print(df.loc[:,["Roll","Name","Marks"]])

grp=df.groupby(['CourseName'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["Roll","Name","Marks"]])

elif ch2==3:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='StudentID')

p=mrkgrd(df)

dup\_df=p.loc[:,["StudentID","Name","Roll","CourseName","Marks","Grade","Pass/Fail"]]

pplt.hist(df['Grade'])

pplt.xlabel('Grade')

pplt.ylabel('No. of students')

pplt.title('Course statistics')

elif ch==3:

ch2=int(input("0.Create Batch database\n1.input Batch Details\n2.View list of all students in a batch\n3.View list of all courses taught in the batch\n4.View complete performance of all students in a batch\n5.Pie Chart of Percentage of all students\nEnter u r choice:"))

if ch2==0:

L1=[['BatchID','BatchName','Department Name','List of course','List of students']]

with open("batch.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

L2=[['BatchID','BatchName','Department Name','CourseID']]

L3=[['BatchID','BatchName','Department Name','StudentID']]

with open("batch2.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L2)

with open("batch3.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L3)

elif ch2==1:

n=int(input("Enter the no. of entries:"))

L1=[]

L2=[]

L3=[]

for i in range(n):

L=[]

BatchID=input("Enter BatchID:")

BatchName=input("Enter BatchName:")

DepartmentName=input("Enter department name:")

t1=list(eval(input("Enter list of courses:")))

t2=list(eval(input("Enter list of students:")))

str1=add1(t1,";")

str2=add1(t2,",")

L=[BatchID,BatchName,DepartmentName,str1,str2]

L1.append(L)

if len(t1)!=1:

for i in t1:

L2p=[BatchID,BatchName,DepartmentName,i]

L2.append(L2p)

else:

L2p=[BatchID,BatchName,DepartmentName,t1[0]]

L2.append(L2p)

if len(t2)!=1:

for i in t2:

L3p=[BatchID,BatchName,DepartmentName,i]

L3.append(L3p)

else:

L3p=[BatchID,BatchName,DepartmentName,t2[0]]

L3.append(L3p)

with open("batch.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

with open("batch2.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L2)

with open("batch3.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L3)

df=pnd.read\_csv("batch.csv")

print(df)

elif ch2==2:

df2=pnd.read\_csv("batch3.csv")

df1=pnd.read\_csv("student.csv")

df=pnd.merge(df1,df2,on='StudentID')

grp=df.groupby(['BatchName'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["Roll","Name","StudentID"]])

elif ch2==3:

df2=pnd.read\_csv("batch2.csv")

df1=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='CourseID')

grp=df.groupby(['BatchName'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["BatchID","BatchName","CourseName","CourseID"]])

elif ch2==4:

df2=pnd.read\_csv("batch2.csv")

df1=pnd.read\_csv("course1.csv")

df4=pnd.merge(df1,df2,on='CourseID')

df3=pnd.read\_csv("student.csv")

df=pnd.merge(df3,df4,on='StudentID')

grp=df.groupby(['BatchName'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["Roll","Name","StudentID",'CourseName','Marks']])

elif ch2==5:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("department1.csv")

df=pnd.merge(df1,df2,on='BatchID')

df3=pnd.read\_csv("course1.csv")

df4=pnd.merge(df,df3,on='StudentID')

df5=df4.groupby('StudentID')['Marks'].mean()

df5.plot(kind='pie')

elif ch==4:

ch2=int(input("0.Create Department database\n1.input Department\n2.View all batches in a department \n3.View average performance of all batches in the department\n4.Show department statistics Line plot–Average percentage of all students for each batch)\nEnter u r choice:"))

if ch2==0:

L1=[['DepartmentID','Department Name','List of batches']]

with open("department.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

L9=[['DepartmentID','Department Name','Batch']]

with open("department1.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L9)

elif ch2==1:

n=int(input("Enter the no. of entries:"))

L1=[]

L2=[]

for i in range(n):

L=[]

L3=[]

DepartmentID=input("Enter departmentid:")

DepartmentName=input("Enter department name:")

t1=list(eval(input("Enter list of batches:")))

str1=add1(t1,":")

L=[DepartmentID,DepartmentName,str1]

L1.append(L)

if len(t1)!=1:

for i in t1:

L3=[DepartmentID,DepartmentName,i]

L2.append(L3)

else:

L3=[DepartmentID,DepartmentName,t1[0]]

L2.append(L3)

with open("department1.csv","a") as r2:

csvwrit=csv.writer(r2)

csvwrit.writerows(L2)

with open("department.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

df=pnd.read\_csv("department.csv")

print(df)

elif ch2==2:

df=pnd.read\_csv("department1.csv")

grp=df.groupby(['DepartmentID'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["BatchID"]])

elif ch2==3:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("department1.csv")

df=pnd.merge(df1,df2,on='BatchID')

df3=pnd.read\_csv("course1.csv")

df4=pnd.merge(df,df3,on='StudentID')

df5=df4.groupby('BatchID')['Marks'].mean()

print(df5)

elif ch2==4:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("department1.csv")

df=pnd.merge(df1,df2,on='BatchID')

df3=pnd.read\_csv("course1.csv")

df4=pnd.merge(df,df3,on='StudentID')

df5=df4.groupby('BatchID')['Marks'].mean()

df5.plot(kind='line')

elif ch==5:

ch2=int(input("0.Create Examination database\n1.input Examination details\n2.View performance of all students in the examination\n3.Show examination statistics:\nEnter u r choice:"))

if ch2==0:

L1=[['Marks','CourseName','StudentID']]

with open("examination.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

elif ch2==1:

n=int(input("Enter the no. of entries:"))

L1=[]

for i in range(n):

L=[]

Marks=input("Enter marks:")

CourseName=input("Enter coursename:")

StudentID=input("Enter studentid:")

L=[Marks,CourseName,StudentID]

L1.append(L)

with open("examination.csv","a") as r1:

csvwrit=csv.writer(r1)

csvwrit.writerows(L1)

df=pnd.read\_csv("examination.csv")

print(df)

elif ch2==2:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='StudentID')

grp =df.groupby(['StudentID'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["CourseName","Marks"]])

elif ch2==3:

df1=pnd.read\_csv("student.csv")

df2=pnd.read\_csv("course1.csv")

df=pnd.merge(df1,df2,on='StudentID')

grp =df.groupby(['BatchName'])

for i,j in grp:

print("\n"+i)

print(j.loc[:,["Roll","Name","Marks"]])

**OUTPUTS:-**