Python File

handling



Name: ***Ujjwal Kakar***

**CERTIFICATE**

CLASS: XI-F YEAR: 2021-2022

This is to certify that Investigatory Project is successfully completed by Ujjwal Kakar of Class: XI F for the academic year 2021-2022.

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

Subject Teacher’s Signature Externam Examiner Signature

**Acknowledgement:**

I, Ujjwal Kakar of class XI- F would like to express our sincere gratitude to my computer science teacher Ms. Gurmeet Kaur, PGT COMPUTER SCIENCE, for her **constant** support and ideas for growth in my endeavour to pursue thoughtful and functional programs as we study python.

I also made use well of online tools to help guide us with more complex parts of syntax and documentation in labriaries, such as time.

**INDEX**

|  |  |
| --- | --- |
| 1 | *Brief Overview of Project* |
| 2 | ***Limitations of Project*** |
| 3 | ***Source Code of Project*** |
| 4 | ***Output Screens*** |
| 4 | ***Future Enhancement of Project*** |
| 5 | ***Bibliography*** |

**Brief Overview of Project:**

**Definition:**

This Project Aims to be able to store all student data of a school and access, edit, add, remove from it easily

**Objective:**

Adding, Removing, Displaying and Formatting Data of students using CSV files

**Making Detail:**

The project “Sudoku Solver” is developed by Ujjwal Kakar, it took approx. 3 days to develop this project, working 2 hours a day. All functions and datasets completed by us only as per my view and knowledge.

**Reason for choosing the Topic:**

Schools are the backbone of modern society.

In a well functioning school, all data of the students must be kept updated and accessible. This project aims to provide the interface for the same.

This project works by using CSV data structure which allows us to store values efficiently and in less space.

This can help in more efficient working of the school systems.

**Hardware Requirements:**

A Computer/Laptop with

Operating System-Windows 7 or above

5 GB free disk space.

**Software Requirements:**

Python 3.9

sd library (Visual Studio Correction Library)

time library

**Limitations of project:**

* It is not web based project
* No provision to print hard copies of student data
* Data is displayed only in a consule UI and not GUI.

**Source Code of Project:**

#File Data handling project

import csv,sd,time #imports Visual Studio Correction Library

fname="Student Data.csv"

#Functions

#Function1 : Special Print Algorithm which prints in table

def printstyled(dat):

maxl = max([len(line) for line in dat])

for line in dat:

while len(line)!=maxl:

line.append('')

#print(dat)

maxlist = []

for times in range(len(line)):

maxval = 0

for line in dat:

if len(line[times]) > maxval:

maxval = len(line[times])

maxlist.append(maxval)

strR = "|"

for length in maxlist:

strR = strR + r"%-"+str(length)+"s|"

print("\_"\*(sum(maxlist)+len(maxlist)+1))

for line in dat:

# print(line)

print(str(strR)%tuple(line))

print("‾"\*(sum(maxlist)+len(maxlist)))

#Introducing Message

print(

"""Welcome to Student database Storage System

This system allows for easy editing, addition, removal, and display of data in the school systems.""")

#base selector

option = 0

#Data is stored in Form of Admno,FirstName,LastName,Email,Gender,Address,Class,Section

formatting = ["Admno","First Name","Last Name","Email","Gender","Address","Class","Section"]

#Main Loop

while True:

#Main Loop Statement

print("Select Operation to perform:","1) Add Data","2) Edit Data","3) Remove Data","4) Display Data",

"5) Display Class List","6) Exit",sep="\n")

#Option Selection

option=int(input("Enter Option Number: "))

#Adding Data

if option==1:

with open(fname,'r+',newline='') as f1:

data = [line for line in csv.reader(f1)]

#General Statement

print("For adding a student, Enter the necessary Details: ")

#Assigns Unused Admission Numbers

print("Auto Assigning Admission Number: ")

#False Delay

for wait1 in range(3):

for wait2 in range(3):

print(".",end="",flush=True)

time.sleep(0.5)

print("\b\b\b\b \b\b\b",flush=True,end="")

time.sleep(1)

print()

#Tells Admission Number and asks rest Details

print("Assigned Admission Number is",len(data)+1)

name = input("Enter Full Name: ")

name = name.split()

email=input("Enter Email: ")

gender = 0

while True:

gender=input("Enter Gender(Male/Female): ")

if gender in ("Male","Female"): break

else: print("Entered Incorrectly")

address=input("Enter Address: ")

stClass = input("Enter Class: ")

section = input("Enter Section: ").upper()

#consolidates and adds to data

details = [len(data)+1,name[0],name[1],email,gender,address,stClass,section]

csv.writer(f1).writerow(details)

print("Data Added Successfully\n\n\n",flush=True)

time.sleep(3)

if option==2:

with open(fname,'r+',newline='') as f1:

data = [line for line in csv.reader(f1)]

#General Statement

name\_inp= input("Enter Name to Edit: ")

#finds name instances confirms, only then permits editing

for num in range(len(data)):

if name\_inp.lower() in str(data[num][1]+' '+data[num][2]).lower():

printstyled([formatting,data[num]])

check = input("Is it correct?(Y/N): ")

if check.lower()=="y":

if int(input("Confirm Admission Number: "))==int(data[num][0]):

name = input("Enter Full Name: ")

name = name.split()

email=input("Enter Email: ")

gender = 0

while True:

gender=input("Enter Gender(Male/Female): ")

if gender in ("Male","Female"): break

else: print("Entered Incorrectly")

address=input("Enter Address: ")

stClass = input("Enter Class: ")

section = input("Enter Section: ").upper()

data[num] = [data[num][0],name[0],name[1],email,gender,address,stClass,section]

else:

print("Wrong Confirmation, No Data Edited")

break

else:

continue

else:

print("Name not in Database")

f1.seek(0,0)

csv.writer(f1).writerows(data)

#delay and confirmatory message

print("Data Edited Successfully\n\n\n",flush=True)

time.sleep(3)

if option==3:

with open(fname,'r+',newline='') as f1:

data = [line for line in csv.reader(f1)]

#General Statement

name\_inp= input("Enter Name to remove: ")

#finds name instances confirms, only then permits removal

for row in data:

if name\_inp.lower() in str(row[1]+' '+row[2]).lower():

printstyled([formatting,row])

check = input("Is it correct?(Y/N): ")

if check.lower()=="y":

if int(input("Confirm Admission Number: "))==int(row[0]):

data.pop(int(row[0])-1)

f1.seek(0)

csv.writer(f1).writerows(data)

print("Entry Removed")

else:

print("Wrong Confirmation, No Data Removed")

break

else:

continue

else:

print("Name not in Database")

elif option==4:

with open(fname,'r+',newline='') as f1:

data = [line for line in csv.reader(f1)]

#General Statement

print("What Parameter do you want to display list with:","1) Show All","2) Name","3) Class","4) Admno",sep="\n")

choice=int(input("Enter Choice: "))

specificDat= [formatting]

specinfo = 0

#takes info where required

if choice==2:

specinfo = input("Enter Name: ")

if choice==3:

specinfo = input("Enter Class: ")

if choice==4:

specinfo = input("Enter Admno: ")

for row in data:

if row != []:

if choice==1:

specificDat=data

elif choice==2:

if specinfo.lower() in (row[1]+" "+row[2]).lower():

specificDat.append(row)

elif choice==3:

if specinfo == row[6]:

specificDat.append(row)

elif choice==4:

if specinfo in row[0]:

specificDat.append(row)

printstyled(specificDat)

elif option==5:

with open(fname,'r+',newline='') as f1:

data = [line for line in csv.reader(f1)]

#General Statement

print("Which Class list do you want to display")

Class , Section = int(input("Enter Class: ")) , input("Enter Section: ").upper()

specificDat = []

if Class<13 and Class>5 and len(Section)==1:

for row in data:

if int(row[6])==Class and row[7]==Section:

specificDat.append(row)

printstyled(sorted(specificDat,key = lambda x: x[1]))

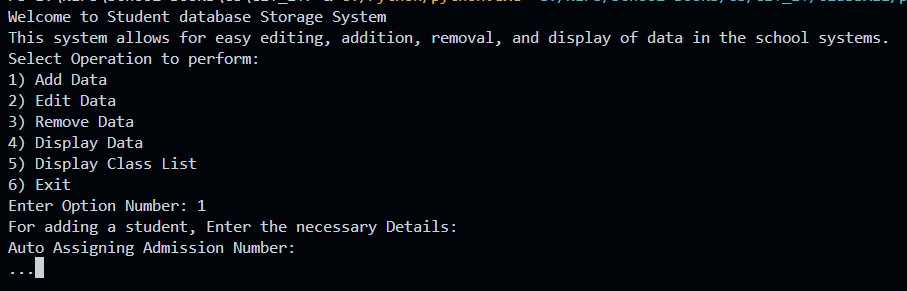
else:

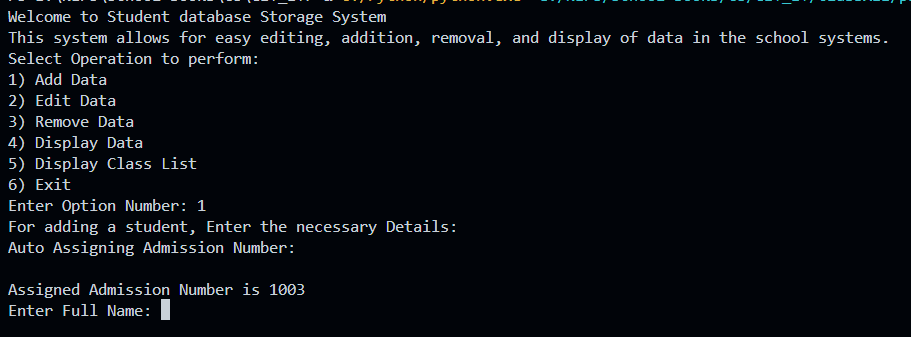
print("Invalid Class or Section")

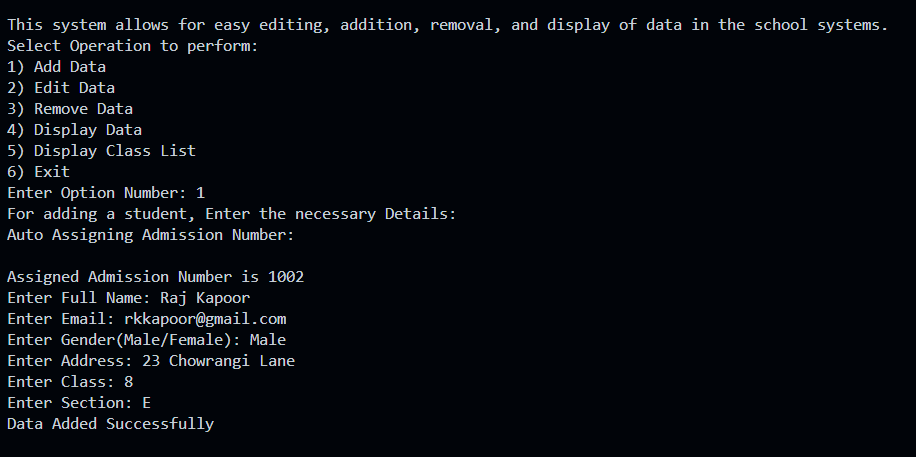
elif option==6:

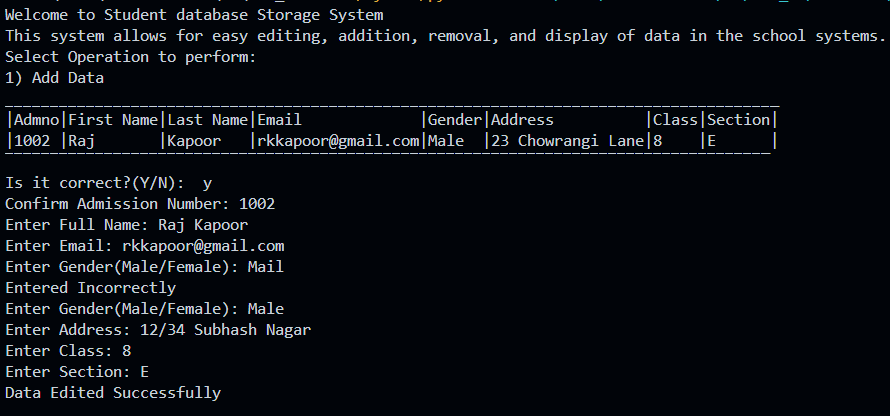
break

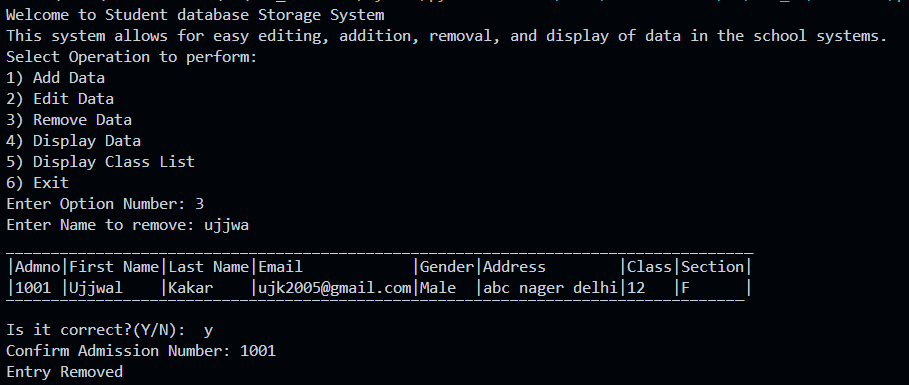
**Output Screens:**

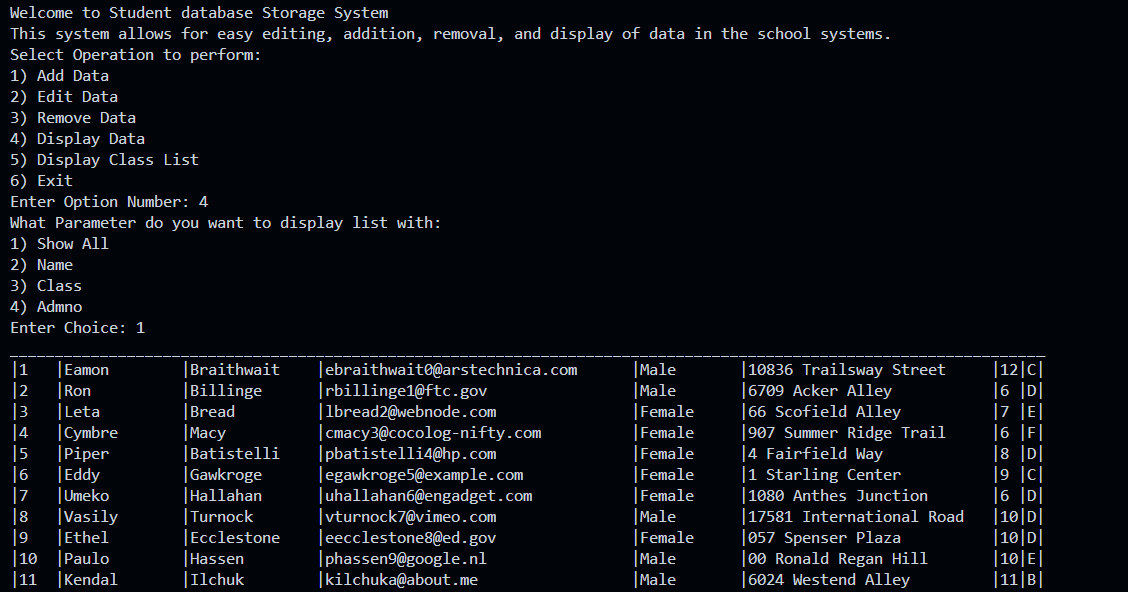
****

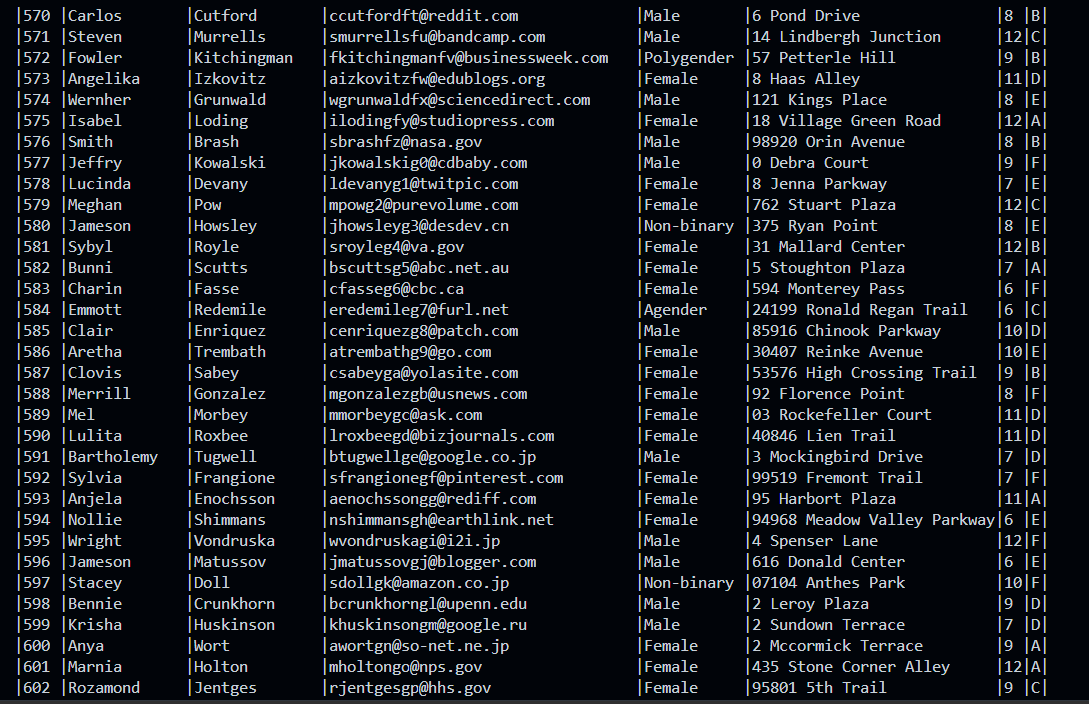
****

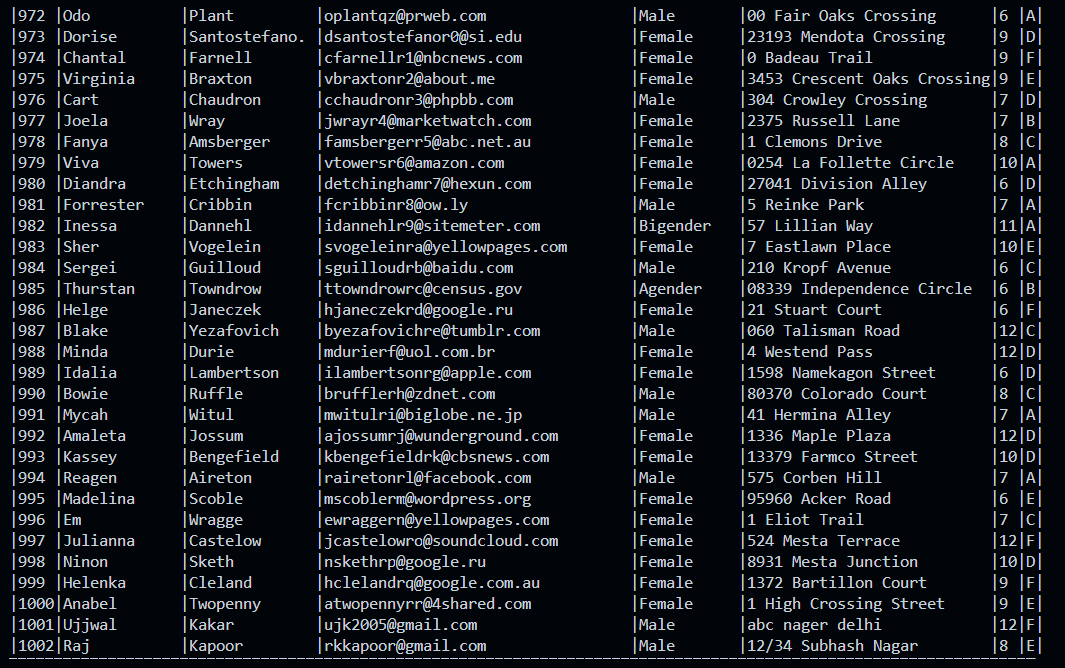


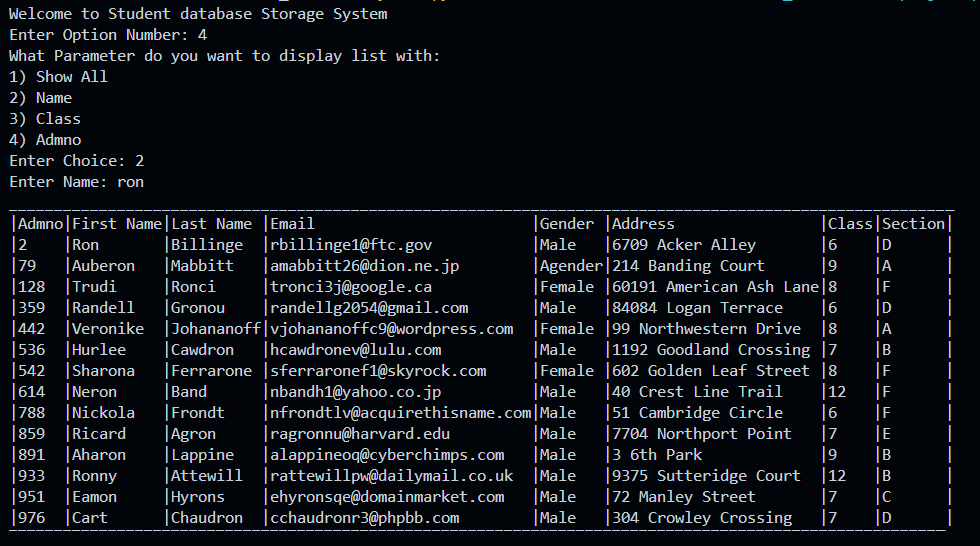
****

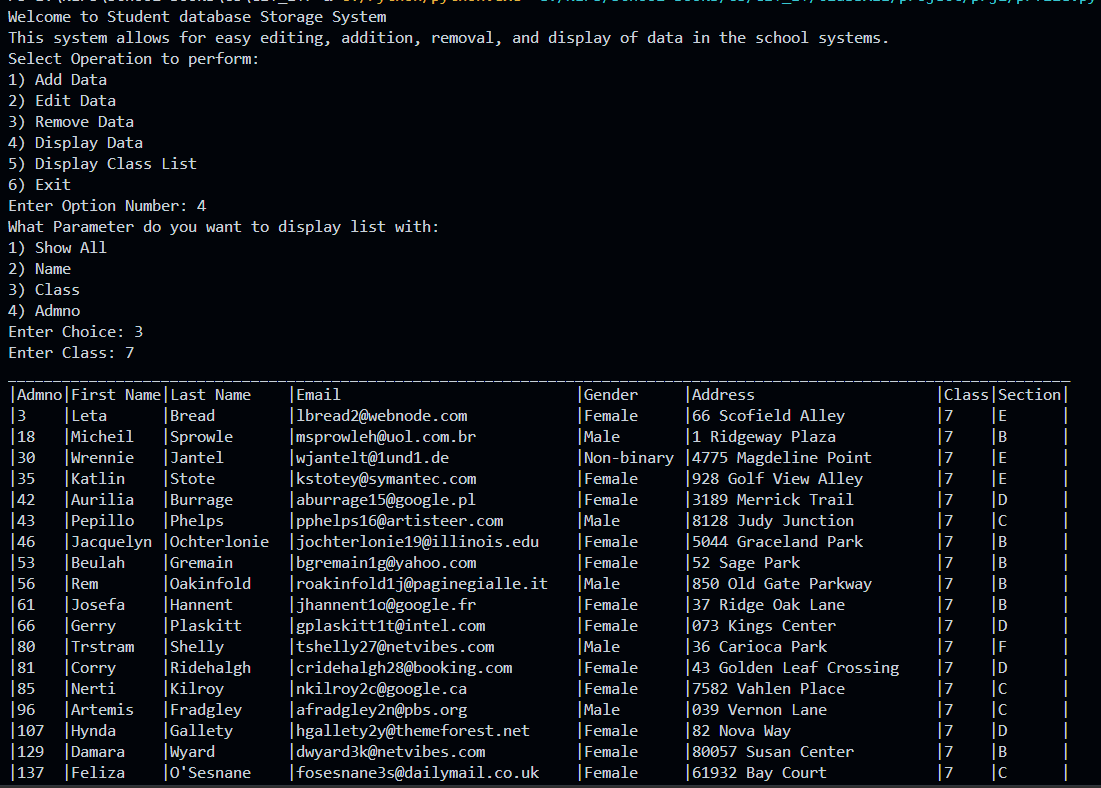
**

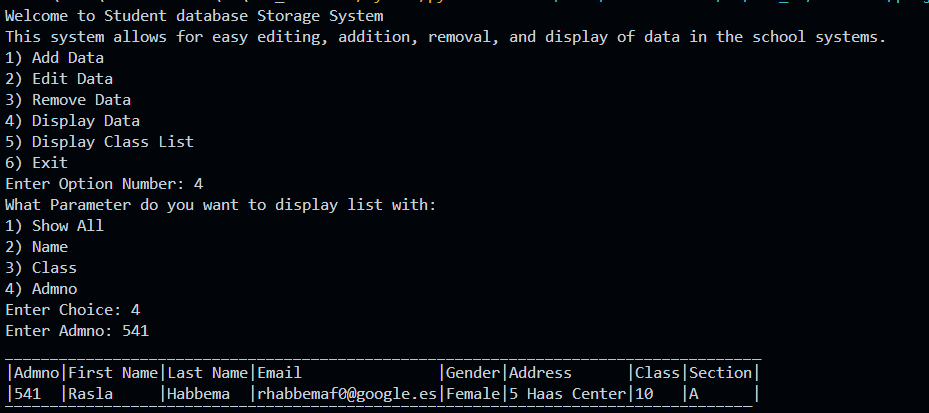
**

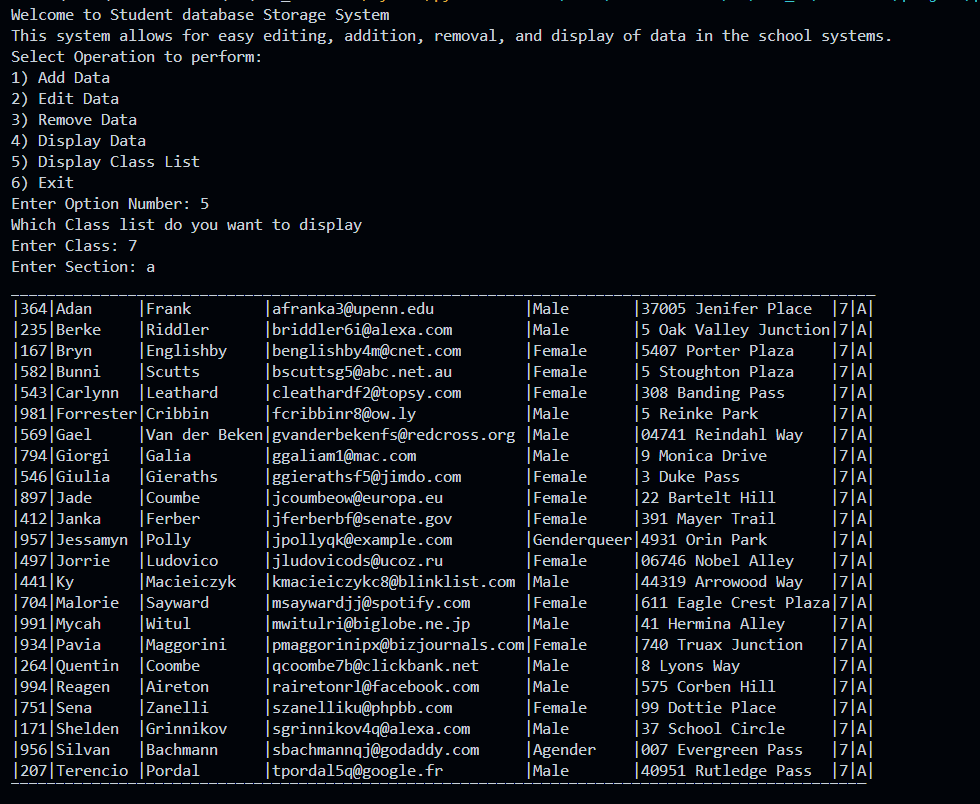
**

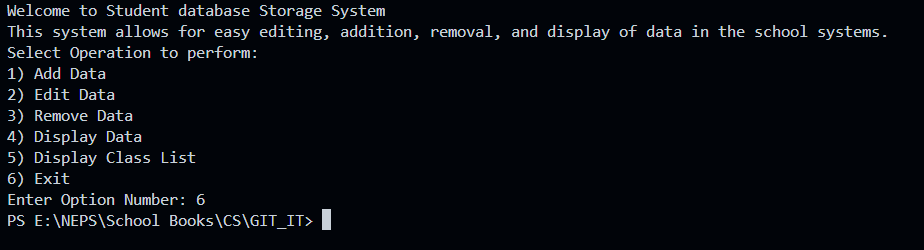
**

**

**

**

**

**

**Future Enhancement of Project:**

We can make a GUI for this project to show it to people easily.

We can export it into a complete and standalone file.

We can also use more efficient algorithms like binary and radial sort.

**Bibliography:**

Text book - Class XII – Computer Science, Preeti Arora

cbse.nic.in

https://www.geeksforgeeks.org/

https://www.w3schools.com/python/

https://en.wikipedia.org

https://mockaroo.com