**JAWAHAR NAVODAYA VIDYALAYA**

**ARNIYAKALAN, SHAJAPUR(M.P.)**



STUDENT INFORMATION SYSTEM

**SESSION 2020-21**

**GUIDED BY:** **SUBMITTED BY:** MR.Ramavtar kumhar Vivek Malviya

(P.G.T c.sc ) CLASS –XII [P.C.b.]

Roll no. 1241

**index**

|  |  |  |
| --- | --- | --- |
| **s.no.** | **Description** | **Page no.** |
| **1.** | Acknowledgements | 3 |
| **2.** | Certificate | 4 |
| **3.** | Introduction | 5 |
| **4.** | Source code | 6-18 |
| **5.** | Output | 19-23 |
| **6.** | Bibliography | 24 |

**ACKNOWLEDMENT**

It is with pleasure that I acknowledge my sincere gratitude to our teacher, **Mr.R.Kumhar(PGT C.Sc.)** who taught and undertook the responsibility of teaching the subject computer science. I have been greatly benefited from his classes. My sincere thanks goes to our Principal **Mrs. Usha Sagar** who has always been a source of encouragement and support and without whose inspiration, this project would not have been a successful.

Finally, I would like to express my sincere appreciation for all the other students for my batch their friendship & the fine times that we all shared together.

Last but not least, I would like to thank all those who had helped directly or indirectly towards the completion of this project .

**Name : Vivek Malviya**

**XII (PCB)**

**CERTIFICATE**

This is to certify that Vivek Malviya of Class XII A (PCB) of Jawahar Navodaya Vidyalya Arniyakala Shajapur(M.P.) has successfully completed his investigatory project (STUDENT INFORMATION SYSTEM) as prescribed by CBSE în the year 2020-21.

He has taken interest and has shown utmost sincerity in completion of this project.

This project is submitted for evalution as a part of CBSE curriculum examination for the year 2020-21.

Date:

Signature of Signature of

Internal Examiner External Examiner

Signature of Principal

###### 

###### INTRODUCTION

A student information system (SIS), student management system(SMS), school administration software or student administration system(SAS) is a management information system for education establishments used to manage student data. Student information systems provide capabilities for registering students in courses; documenting grading, transcripts, results of student tests and other assessment scores; building student schedules; tracking student attendance; and managing many other student-related data needs in a school.

Information security is a concern, as universities house an array of sensitive personal information, making them potentially attractive targets for security breaches.The purpose of an information system is to turn raw data into useful information that can be used for decision making in an organization. Many information systems are designed to support a particular process within an organization or to carry out very specific analysis.

Let’s interact with the system source code.

**<<< SOURCE CODE >>>**

#import essential libraries

from cfonts import render

from prettytable import from\_csv,PrettyTable

import csv

import time

import datetime

# Define global variables

student\_fields = ['Name','Father/'s Name','Class','Roll No','DOB','Sex(M/F/T)','Phone']

student\_database = 'students.csv'

#banner artwork

banner = render("s.i.s", colors=['yellow', 'green'], align='center')

print(banner)

#ending banner

end\_banner = render("Thank You", colors=['green', 'yellow'], align='center')

#colours used in program

GREEN = '\x1b[1;32;40m'

RED = '\x1b[1;31;40m'

GREEN\_WHITE = '\x1b[0;30;42m'

PURPLE = '\x1b[1;36;40m'

U\_YELLOW = '\x1b[4;33;40m'

WHITE = '\x1b[1;37;40m'

YELLOW = '\x1b[1;33;40m'

END = '\x1b[0m'

#symbol used in program

arrow\_left = PURPLE + '>>>' + END

arrow\_right = PURPLE + '<<<' + END

block = '|'

space =' '

arrow = PURPLE + ' ##>' + END

arrow\_m = PURPLE + ' »' + END

tick = GREEN + '✓✓ ' + END

box = GREEN +' ['+END+YELLOW+'\*'+END+GREEN +'] ' +END

#msg box variables

welcome\_msg = GREEN + 'Welcome To Student Information System !' + END

info\_1 = GREEN + '\tAdd Student Information' + END

info\_2 = GREEN + '\tStudents Record !' +END

info\_3 = GREEN + '\tSearch Student !' +END

info\_4 = GREEN +'\tUpdate Student Information !'+END

info\_5 = RED +'\tDelete Student Record !'+END

info\_6 = YELLOW+'\tProject Development Team !'+END

#stylish numbers

one = YELLOW+' ['+END+'1'+YELLOW+']'+END

two = YELLOW+' ['+END+'2'+YELLOW+']'+END

three = YELLOW+' ['+END+'3'+YELLOW+']'+END

four = YELLOW+' ['+END+'4'+YELLOW+']'+END

five = YELLOW+' ['+END+'5'+YELLOW+']'+END

six = YELLOW+' ['+END+'6'+YELLOW+']'+END

seven = YELLOW+' ['+END+'7'+YELLOW+']'+END

#choices

w\_one = WHITE + ' Add New Student' + END

w\_two = WHITE + ' View Students' + END

w\_three = WHITE + ' Search Student' + END

w\_four = WHITE + ' Update Student' + END

w\_five = WHITE + ' Delete Student' + END

w\_six = WHITE + ' Project Development Team'+END

w\_seven = WHITE + ' Quit' + END

#input field colour

w\_input = WHITE + ' Enter Your Choice : ' + END

#welcome msg\_box

l = ''.join(['+'] + ['-' \*48] + ['+'])

wlc\_msg = l + '\n'+block+space+arrow\_left+space+welcome\_msg+space+arrow\_right+space+block+'\n' + l

#add student msg\_box

m = ''.join(['+'] + ['-' \*40] + ['+'])

result\_1 = m + '\n' +info\_1 + '\n' + m

#view student msg\_box

n = ''.join(['+'] + ['-' \*30] + ['+'])

result\_2 = n + '\n' + info\_2 + '\n' + n

#search student msg\_box

p = ''.join(['+'] + ['-' \*30] + ['+'])

result\_3 = p + '\n' + info\_3 + '\n' + p

#update student msg\_box

q = ''.join(['+'] + ['-' \*40] + ['+'])

result\_4 = q + '\n' + info\_4 + '\n' + q

#delete student recode msg\_box

r = ''.join(['+'] + ['-' \*35] + ['+'])

result\_5 = r + '\n' + info\_5 + '\n' + r

#about us msg\_box

s = ''.join(['+'] + ['-'\*40] + ['+'])

result\_6 = s + '\n' + info\_6 + '\n' + s

#press enter to continue msg

def continue\_msg():

print('\n')

input(arrow + YELLOW + " Press Enter To Continue : " + END)

print('\n')

#choice menu

def display\_menu():

print(wlc\_msg)

print(arrow+one+w\_one)

print(arrow+two+w\_two)

print(arrow+three+w\_three)

print(arrow+four+w\_four)

print(arrow+five+w\_five)

print(arrow+six+w\_six)

print(arrow+seven+w\_seven)

print('\n')

#add student function

def add\_student():

print(result\_1)

student\_data = []

def validate\_name():

if name.replace(" ", "").isalpha():

pass

else:

print("Input is invalid")

quit()

name = input(arrow\_m +YELLOW+ " Name of Student : " + END)

validate\_name()

father = input(arrow\_m + YELLOW+" Father's Name' :" + END)

validate\_name()

def validate\_class():

try:

val = int(clas)

except ValueError:

print("That's not an !")

quit()

clas = input(arrow\_m +YELLOW +" Class :"+END)

validate\_class()

def validate\_roll():

try:

val = int(roll)

except ValueError:

print("Sorry Invalid Input!!!")

quit()

roll = input(arrow\_m + YELLOW +" Roll No. : "+ END)

validate\_roll()

def validate\_dob():

date\_format = '%d-%m-%Y'

try:

date\_obj = datetime.datetime.strptime(dob, date\_format)

print(date\_obj)

except ValueError:

print("Incorrect data format,It should be YYYY-MM-DD")

dob = input(arrow\_m + YELLOW +" Date Of Birth :"+ END)

validate\_dob()

valid\_sex = ['F','M','T','t','m','f','other']

def validate\_sex():

if sex in valid\_sex:

pass

else:

print("wrong input")

sex = input(arrow\_m + YELLOW +" Sex(m/f/t):"+ END)

validate\_sex()

def validate\_phone():

try:

val = int(phone)

except ValueError:

print("Incorrect Input!!")

phone = input(arrow\_m +YELLOW+" Phone no.:-"+ END)

validate\_phone()

student\_data.append(name)

student\_data.append(father)

student\_data.append(clas)

student\_data.append(roll)

student\_data.append(dob)

student\_data.append(sex)

student\_data.append(phone)

with open(student\_database, "a", encoding="utf-8") as f:

writer = csv.writer(f)

writer.writerows([student\_data])

x = PrettyTable()

x.field\_names = [GREEN+'Name','Fathers Name','Class','Roll No','DOB','Sex(M/F/T)','Phone'+END]

x.add\_row([PURPLE+student\_data[0],student\_data[1],student\_data[2],student\_data[3],student\_data[4],student\_data[5],student\_data[6]+END])

print('\n')

print(x)

print('\n')

print(tick+GREEN\_WHITE + " Data Saved Successfully ! " + END)

continue\_msg()

return

#function view students

def view\_students():

print(result\_2)

print("\n")

fp = open("students.csv", "r")

file = from\_csv(fp)

fp.close()

print(file)

continue\_msg()

#function search student

def search\_student():

print(result\_3)

roll = input(arrow\_m+ YELLOW + " Enter Roll No. To Search: " + END)

with open(student\_database, "r", encoding="utf-8") as f:

reader = csv.reader(f)

for row in reader:

if len(row) > 0:

if roll == row[3]:

print('\n')

['Name','Fathers Name','Class','Roll No','DOB','Sex(M/F/T)','Phone']

print(GREEN+"\t----- Student Found -----" + END)

print(box+YELLOW+"Student Name:"+ END,PURPLE +row[0]+END)

print(box+YELLOW+"Father's Name:"+END,PURPLE+row[1]+END)

print(box+YELLOW+"Class: "+END,PURPLE+row[2]+END)

print(box+YELLOW+"Roll No: "+END,PURPLE+row[3]+END)

print(box+YELLOW+"DOB (DD/MM/YYYY): "+END,PURPLE+row[4]+END)

print(box+YELLOW+"Sex(M/F/T): "+END,PURPLE+row[5]+END)

print(box+YELLOW+"Phone No. : "+END,PURPLE+row[6]+END)

continue\_msg()

break

else:

print(box+RED+"ALERT!"+END)

print(RED+' ×'\*20+END)

print(RED+" Student Not Found In Our Database !!!" +END)

continue\_msg()

#function update student

def update\_student():

print(result\_4)

roll = input(arrow\_m+ YELLOW + " Enter Roll No. To Update : " + END)

index\_student = None

updated\_data = []

with open(student\_database, "r", encoding="utf-8") as f:

reader = csv.reader(f)

counter = 0

for row in reader:

if len(row) > 0:

if roll == row[3]:

index\_student = counter

print('\n')

print(PURPLE+'\t------RECORD FOUND------'+END)

print(box+GREEN+"Student Found At Index No "+END,index\_student)

student\_data = []

for field in student\_fields:

value = input(arrow\_m +YELLOW+ " Enter New " + field + ": " + END)

student\_data.append(value)

updated\_data.append(student\_data)

else:

updated\_data.append(row)

counter += 1

# Check if the record is found or not

if index\_student is not None:

with open(student\_database, "w", encoding="utf-8") as f:

writer = csv.writer(f)

writer.writerows(updated\_data)

print('\n')

print(tick+GREEN\_WHITE + " Data Updated Successfully ! " + END)

continue\_msg()

else:

print(box+RED+"ALERT!"+END)

print(RED+' ×'\*20+END)

print(RED+" Student Not Found In Our Database !!!" +END)

continue\_msg()

#function delete students

def delete\_student():

print(result\_5)

roll = input(arrow\_m+ YELLOW + " Enter Roll No. To Delete : " + END)

student\_found = False

updated\_data = []

with open(student\_database, "r", encoding="utf-8") as f:

reader = csv.reader(f)

counter = 0

for row in reader:

if len(row) > 0:

if roll != row[3]:

updated\_data.append(row)

counter += 1

else:

student\_found = True

if student\_found is True:

with open(student\_database, "w", encoding="utf-8") as f:

writer = csv.writer(f)

writer.writerows(updated\_data)

print('\n')

print(tick+GREEN\_WHITE+"Roll no.", roll, "Deleted successfully"+END)

continue\_msg()

else:

print(box+RED+"ALERT!"+END)

print(RED+' ×'\*20+END)

print(RED+" Roll No. Not Found In Our Database !!!" +END)

continue\_msg()

name\_1 = 'Vivek Malviya'

name\_2 = 'Gopal Kalsiya'

name\_3 = 'Deepesh Rathore'

name\_4 = 'Naval Barodh'

designation\_1 = 'Student'

designation\_2 = 'Student'

designation\_3 = 'Student'

designation\_4 = 'Student'

class\_1 = 'XII(PCB)'

class\_2 = 'XII(PCM)'

class\_3 = 'XII(PCM)'

class\_4 = 'XII(PCM)'

#about us wala function

def about\_us():

print(result\_6)

z = PrettyTable()

z.field\_names = [PURPLE+"Team member","Designation","class"+END]

z.add\_row([name\_1, designation\_1,class\_1])

z.add\_row([name\_2, designation\_2,class\_2])

z.add\_row([name\_3, designation\_3,class\_3])

z.add\_row([name\_4, designation\_4,class\_4])

print(z)

continue\_msg()

while True:

display\_menu()

choice = input(arrow+w\_input)

if choice == '1':

add\_student()

elif choice == '2':

view\_students()

elif choice == '3':

search\_student()

elif choice == '4':

update\_student()

elif choice == '5':

delete\_student()

elif choice == '6':

about\_us()

elif choice == '7':

print('\n')

print(box+PURPLE+"Thanks For Using Our Student Management System"+END)

print(box+YELLOW+"Quitting...."+END)

time.sleep(2)

print(end\_banner)

quit()

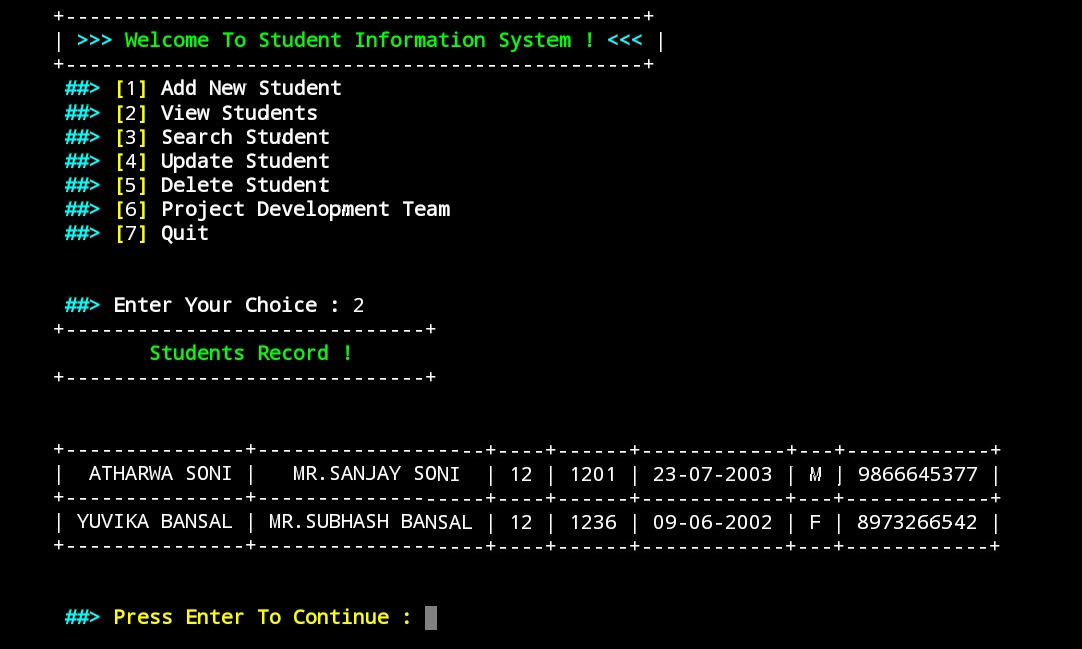
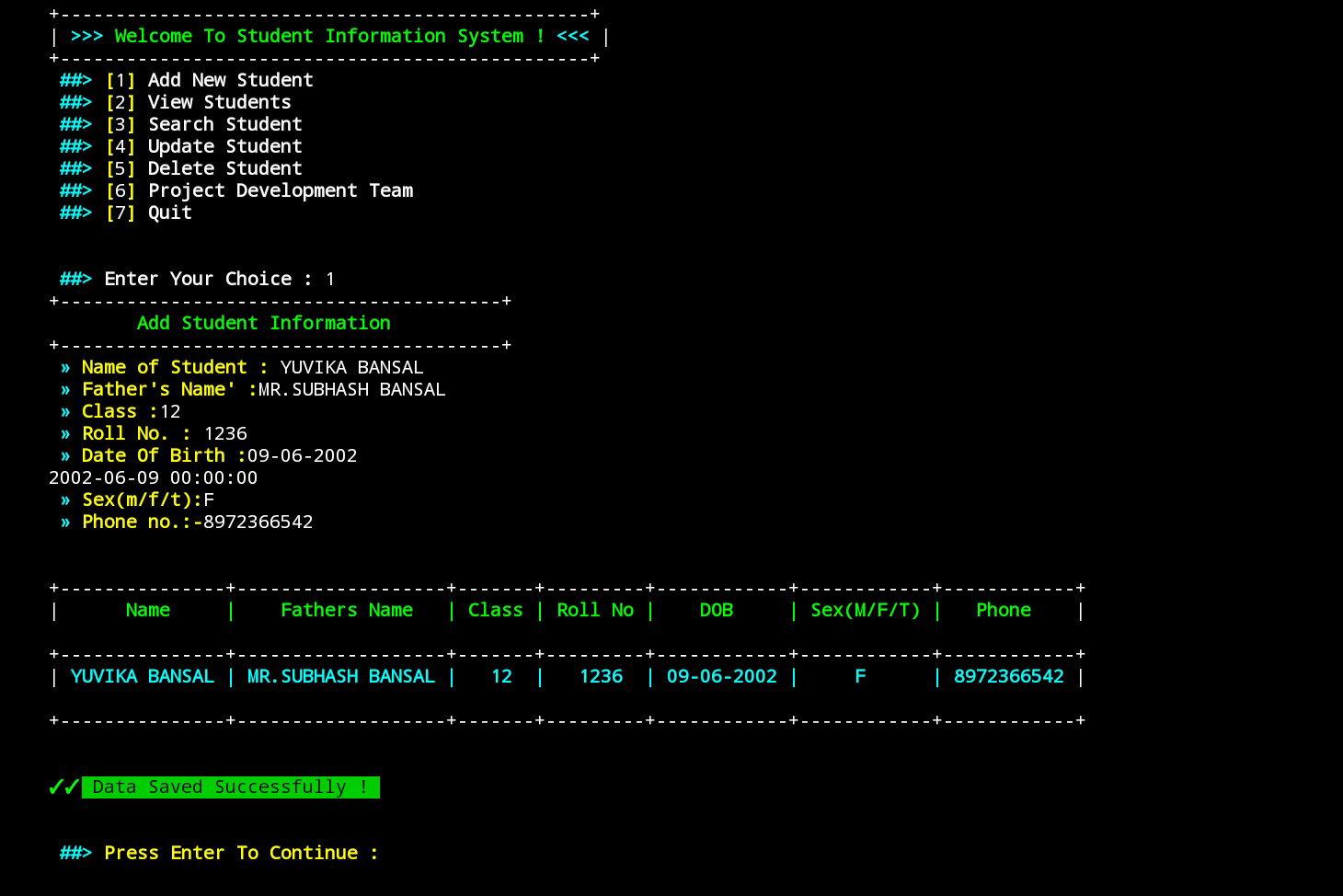
else:

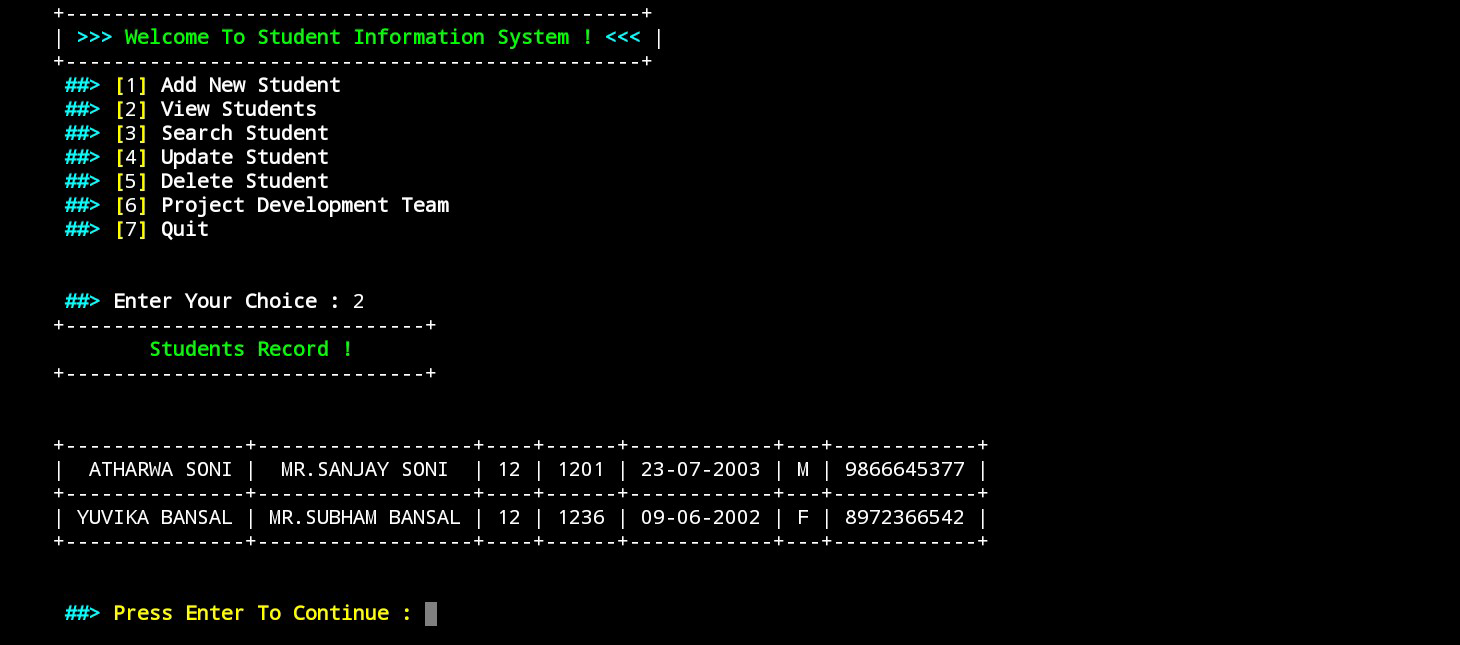
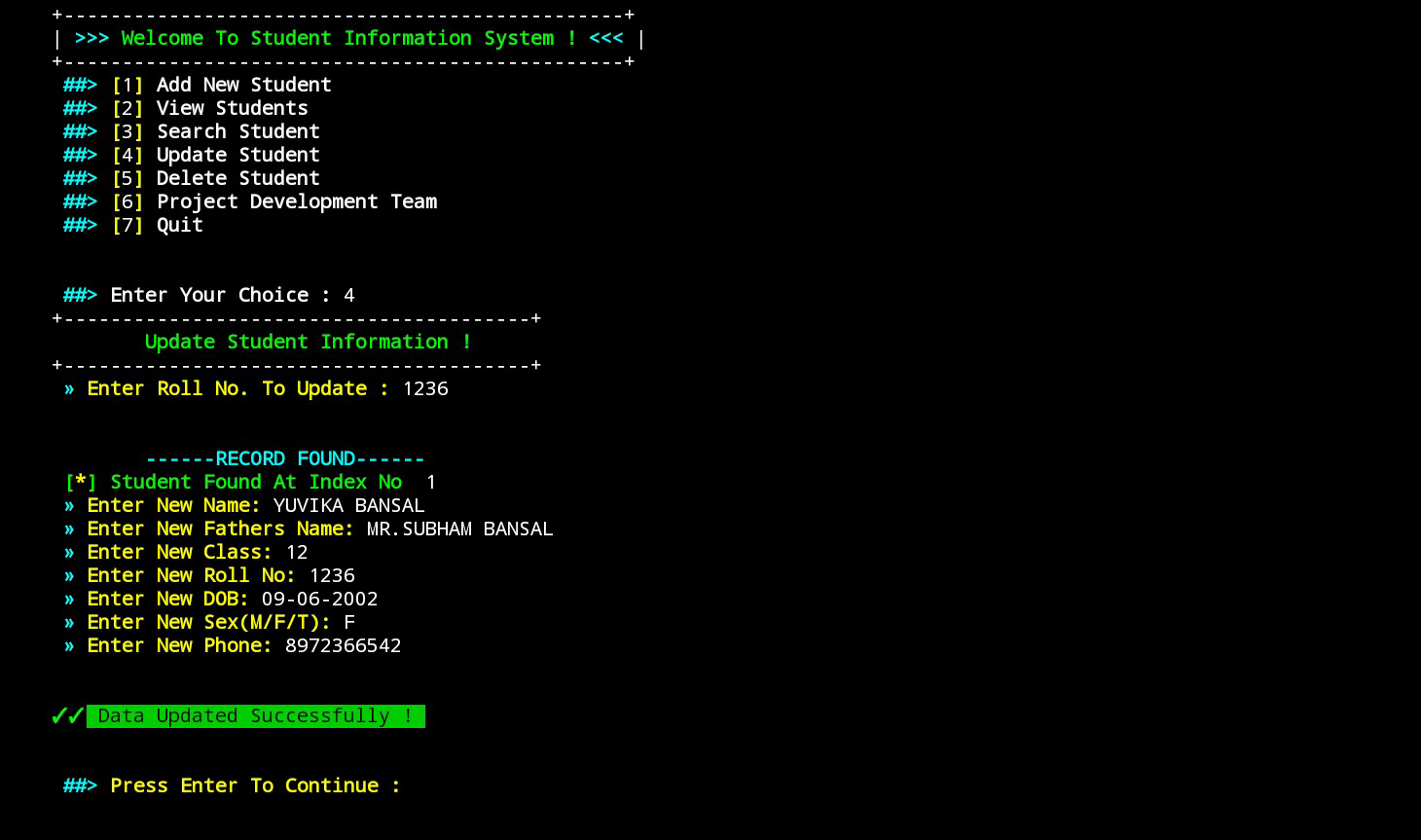
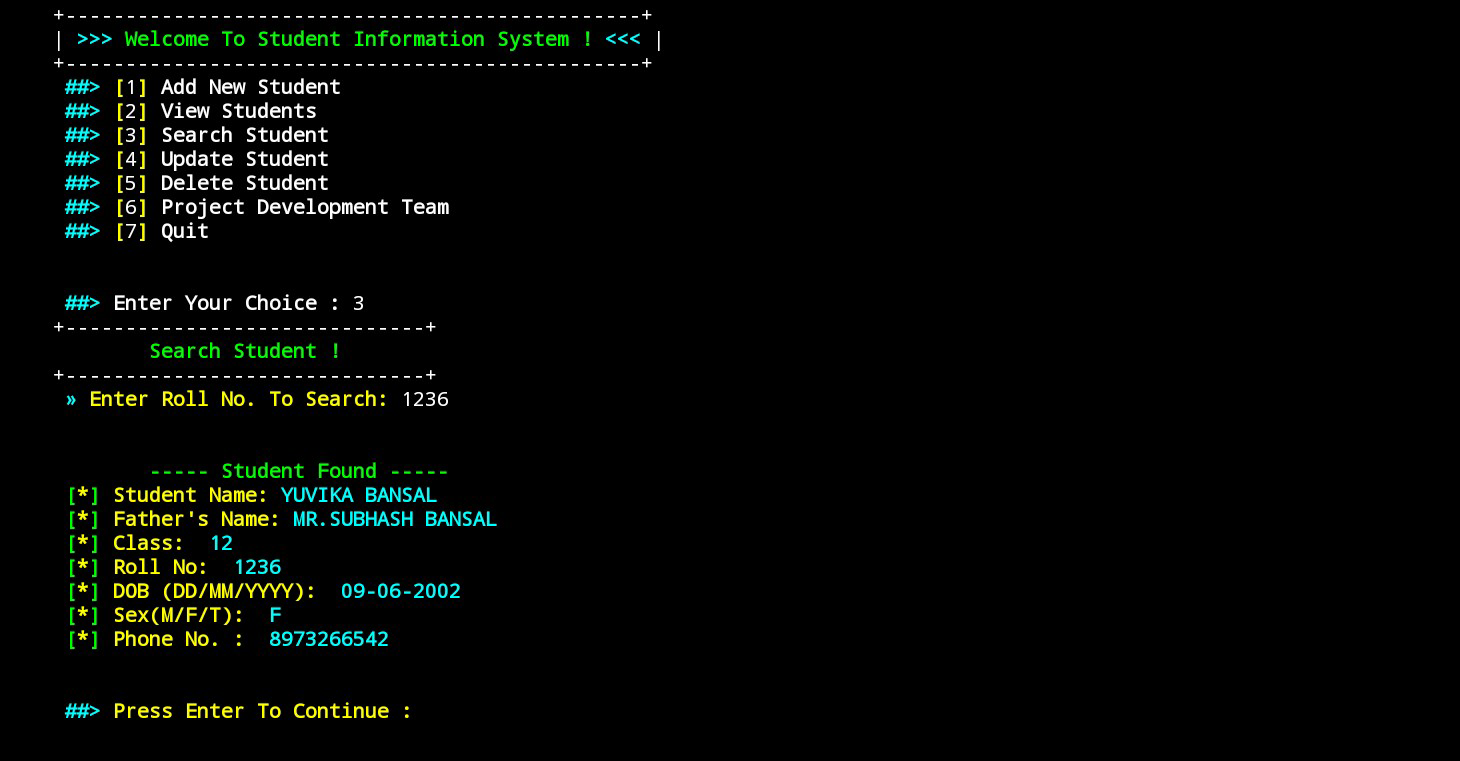
print(box+RED+'Sorry! Enter a valid input !'+END)

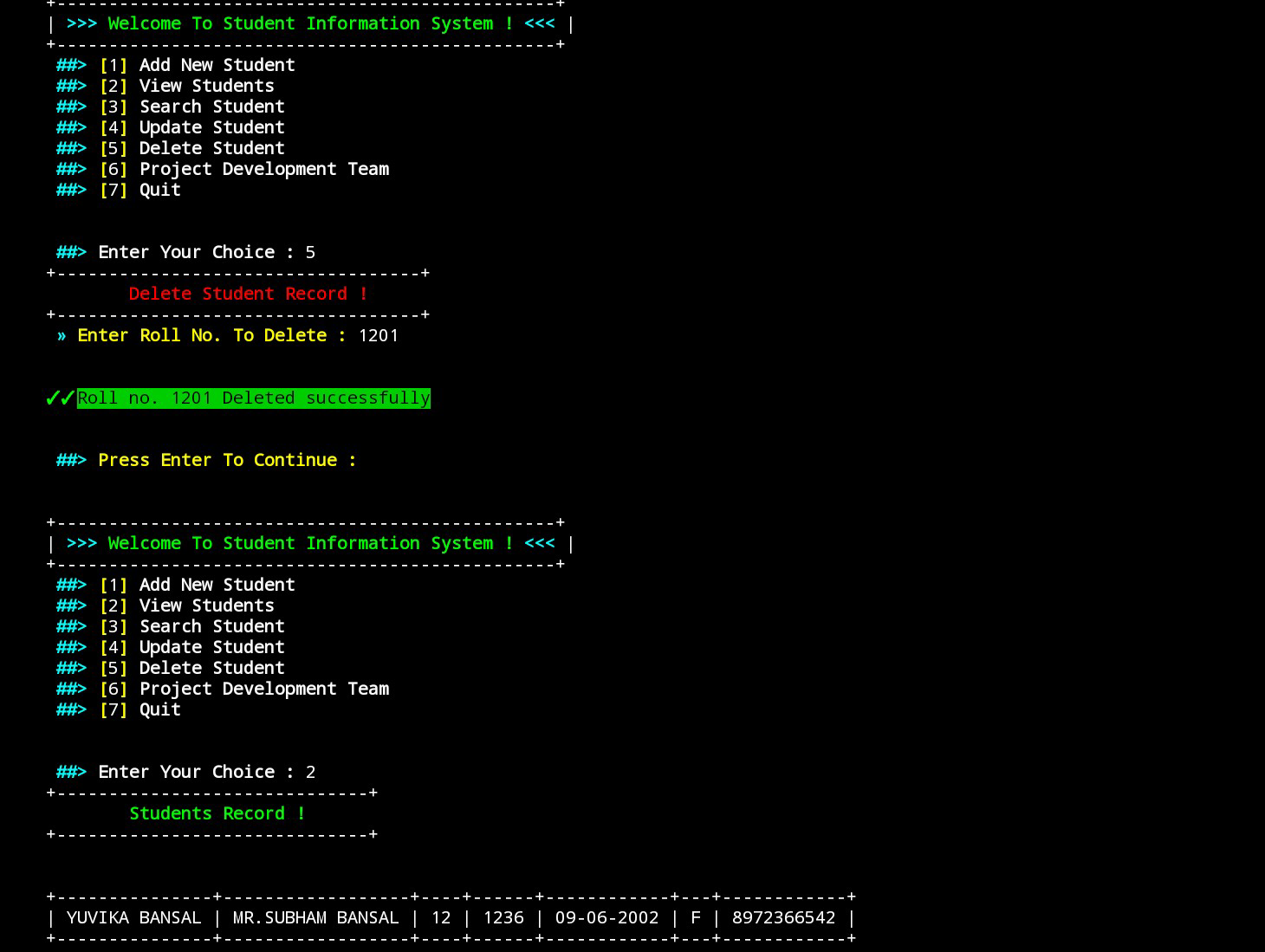
break

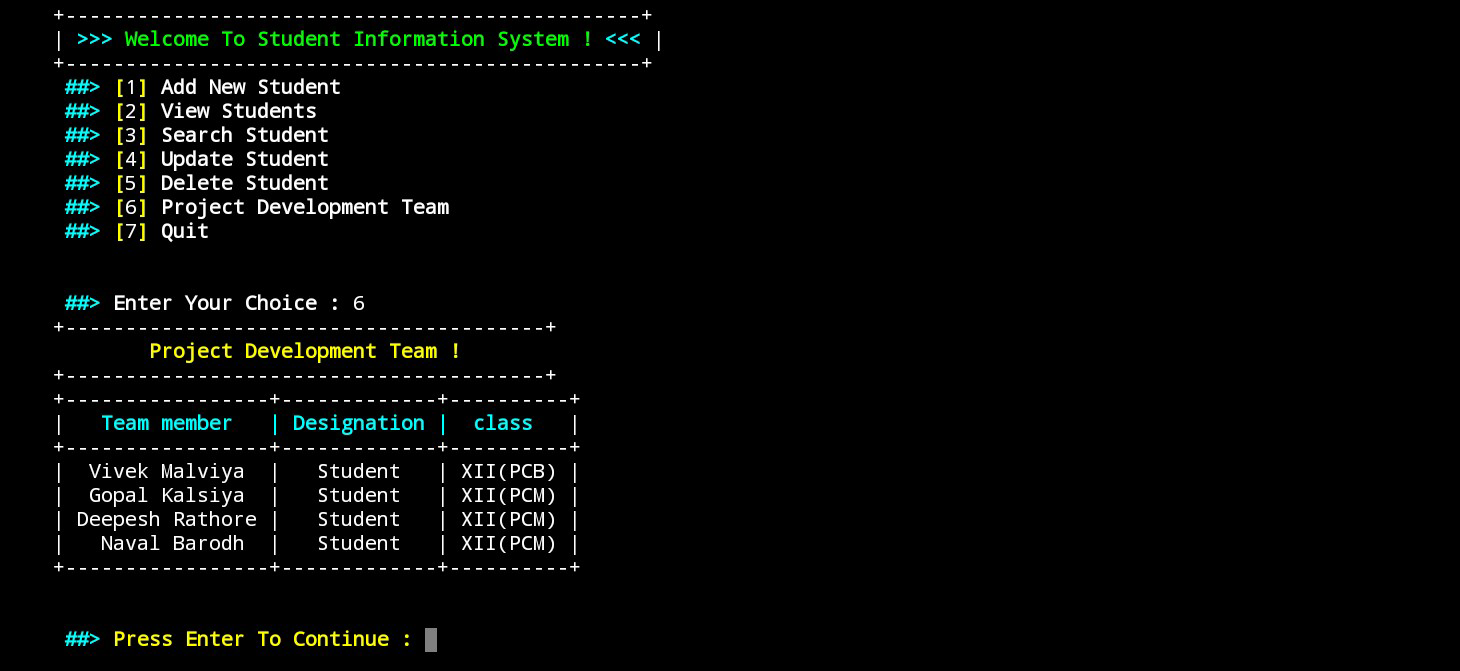
**<<< Output >>>**

Screenshots











BIBLIOGRAPHY

1. Computer Science with Python XII Sumita Arora

2. Python Projects for Beginners: A Ten-Week Bootcamp Approach to … Connor P. Milikan

3. Python Projects Alan Gauld and Laura Cassell

4. Computer Science with Python XI Sumita Arora

5. <https://docs.python.org/3/>

6. <https://www.tutorialspoint.com/python/index.htm>