

STUDENT MANAGEMENT SYSTEM USING PYTHON AND TKINTER

A PROJECT REPORT

Submitted By

*Shourya Gupta
Sounak Saha*

Under the Project Guidance of

Sofikul Mullick

At

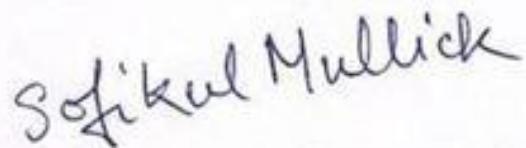
Ardent Computech Pvt. Ltd.



BONAFIED CERTIFICATE

This is to certify that **Shourya Gupta** and **Sounak Saha** have successfully completed the project titled “***Student Management System Using Python and Tkinter***” under my supervision during the period from April 1 2021 to April 19 2021 which is in fulfillment of their training in Python Programming.

Signature of Mentor



A handwritten signature in black ink, appearing to read "Sofikul Mullick".

**Prof. Sofikul Mullick Ardent
Computech Pvt. Ltd.
Salt Lake, Kolkata**

ACKNOWLEDGEMENT

The project “ ***Student Management System Using Python and Tkinter*** ” would not have been possible without the constant guidance of our guide **Sofikul Mullick** , Ardent Computech Pvt. Ltd., who guided us throughout this process. We are immensely thankful to him for his valuable ideas on improvement of the project.

INDEX

| Topic | Page number |
|-----------------------------------|-------------|
| ABSTRACT | 1 |
| INTRODUCTION | 2 |
| PROBLEM DEFINITION | 4 |
| PROBLEM OBJECTIVE | 4 |
| METHODOLOGY | 5 |
| PROJECT WORKFLOW | 6 |
| PROJECT IMPLEMENTATION | 7 |
| STEP-BY-STEP WORKING | 16 |
| PROJECT LIMITATIONS | 26 |
| FUTURE SCOPE | 27 |
| SUMMARY | 28 |
| BIBLIOGRAPHY | 29 |

ABSTRACT

Databases are an integral part of life, having the aim to store data efficiently which is easily retrievable. One of the most common examples includes students databases in various institutions. Educational institutions need to manage huge data of students for managing school affairs systematically. To streamline the process of managing student data, a lot of educational institutions are investing in a high-performance student database management system, in order to store students' information with utmost security, reliability and easy accessibility with minimal human intervention thus minimizing paperworks. The project aims to successfully create a database to store information of students with such flexibility.



INTRODUCTION

Before going to further discussion, a common question arises.

What is a database?

Basically a structured set of data held in a computer, especially one that is accessible in various ways is called database as discussed already.

In our attempt to adopt database in this project we opt to choose python language.

Now why using Python?

Python is a general purpose programming language. Hence, you can use the programming language for developing both desktop and web applications. Also, you can use Python for developing complex scientific and numeric applications. Python is designed with features to facilitate data analysis and visualization. So Python is an appropriate language to be used.



In our process of creation of the database python SQLite3 and Tkinter modules have been used.

SQLite3

SQLite is a self-contained, file-based SQL database. SQLite comes bundled with Python and can be used in any of our Python applications without having to install any additional software.

Tkinter

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.



PROBLEM DEFINITION

Institutions have to handle large sets of information of students. To make this procedure more flexible, efficient and less time consuming we need to have students database which should contain personal information of students. The student management system thus aims to create such a database granting access to details of students say, college details, course details, personal details, academic details etc.

PROBLEM OBJECTIVE

The objective is to develop an user-friendly student management system which will store and keep a track of the student details in the database and allow data updation or deletion through a standard object-oriented GUI(Tkinter).



METHODOLOGY

- **Creation of the main window**

Using the Tkinter module the main window is created. It is the standard object oriented GUI which is supposed to interact with the user and contain the widgets.

- **Adding the Widgets**

In the main window the widgets are incorporated. The entry fields for students details along with labels are created and placed. It is followed by Buttons. The buttons when pressed will carry out specific tasks.

- **Creating the database and establishing connection with it**

The database must contain the details of the students say college ID, Name, Department etc. Within a table. The database and table is created using SQLite3 module.



•Creating a Treeview Scrollbar

Using the Treeview and Scrollbar widgets the features of the entries i.e., students' details can be displayed in the form of a table which can be scrolled. Basically it is supposed to display the details of all the students.

Project Work Flow



PROJECT IMPLEMENTATION

Source code of the project is provided below:

```
#Program in Python to make a Student Management system  
using Python Database module "SQLite3" and Python GUI  
module "Tkinter"
```

```
#Importing all the modules  
import sqlite3  
from tkinter import *  
from tkinter import ttk
```

```
#Creating the database and a table within it  
try:
```

```
    db=sqlite3.connect("SHOURYA_SOUNAK.db")  
    x=db.cursor()  
    x.execute("CREATE table Students_2021(College_id  
integer,Name text,Age integer,Department text,Address  
text,Email text,Gender text)")  
    db.commit()
```

```
except:  
    print("Table is already present within the database.")  
  
#Creating the window using tkinter  
w=Tk()  
  
#User inputs for various fields  
c=StringVar()  
n=StringVar()  
a=StringVar()  
d=StringVar()  
e=StringVar()  
g=StringVar()  
  
#Setting the specifications>Title,Size,Background color  
of the window  
w.title('Students Enrollment Form')  
w.geometry('2000x4000')  
w.configure(background='pink')  
  
#Setting an Heading in the window which will occupy  
the top and will expand accross the entire width  
H=Label(text='2021 ENROLLMENT  
FORM',font=('times new  
roman',30,'bold'),bd=4,relief=GROOVE,bg='purple',fg='  
gold')  
H.pack(side=TOP,fill=X)
```

#Information fields to get College_Id of the students
Lc=Label(w,text='College id',font=('times new roman',25),bg='pink')
Lc.place(x=10,y=60)

Ec=Entry(w,font=('times new roman',25),bg='white',textvar=c)
Ec.place(x=210,y=60)

#Information fields to get Name of the students
Ln=Label(w,text='Name',font=('times new roman',25),bg='pink')
Ln.place(x=10,y=120)

En=Entry(w,font=('times new roman',25),bg='white',textvar=n)
En.place(x=210,y=120)

#Information fields to get Age of the students
La=Label(w,text='Age',font=('times new roman',25),bg='pink')
La.place(x=10,y=180)

Ea=Entry(w,font=('times new roman',25),bg='white',textvar=a)
Ea.place(x=210,y=180)

#Information fields to get Department of the students
Ld=Label(w,text='Department',font=('times new roman',25),bg='pink')
Ld.place(x=10,y=240)

Ed=Entry(w,font=('times new roman',25),bg='white',textvar=d)
Ed.place(x=210,y=240)

#Information fields to get Address of the students
Lr=Label(w,text='Address',font=('times new roman',25),bg='pink')
Lr.place(x=10,y=300)

Tr=Text(w,width=48,height=3,font=("Times New Roman",10))
Tr.place(x=210,y=300)

```
#Information fields to get Email of the students
```

```
Le=Label(w,text='Email',font=('times new roman',25),bg='pink')
```

```
Le.place(x=10,y=360)
```

```
Ee=Entry(w,font=('times new roman',25),bg='white',textvar=e)
```

```
Ee.place(x=210,y=360)
```

```
#Information fields to get Gender of the students
```

```
Lg=Label(w,text='Gender',font=('times new roman',25),bg='pink')
```

```
Lg.place(x=10,y=420)
```

```
Cg=ttk.Combobox(w,font=('times new roman',23),state='readonly',  
textvar=g)
```

```
Cg['values']=('Male','Female','Other')
```

```
Cg.place(x=210,y=420)
```

```
#Creating a pair of frames
```

```
fm1=Frame(w,bd=4,relief=RIDGE,bg='yellow')
```

```
fm1.place(x=648,y=60,width=698,height=630)
```

```
fm2=Frame(w,bd=4,relief=RIDGE,bg='blue')
```

```
fm2.place(x=658,y=80,width=669,height=585)
```

```
#Function block to add the personal data of the students into the database
```

```
def submit():
```

```
    h=c.get()
```

```
    i=n.get()
```

```
    j=a.get()
```

```
    k=d.get()
```

```
    l=e.get()
```

```
    m=g.get()
```

```
    x.execute("insert into Students_2021(College_id,Name,Age,  
    Department,Address,Email,Gender)values  
    (? ,? ,? ,? ,? ,? ,?)", (h,i,j,k,Tr.get('1.0',END),l,m))
```

```
    show()
```

```
    clear()
```

```
    db.commit()
```

```
#Function block to modify a data field of an already existing  
student info within the database
```

```
def update():
```

```
    h=c.get()  
    i=n.get()  
    j=a.get()  
    k=d.get()  
    l=e.get()  
    m=g.get()
```

```
    x.execute("update Students_2021 set  
    Name=?,Age=?,Department=?,Address=?,Email=?,Gender=?  
    where College_id=?",(i,j,k,Tr.get('1.0',END),l,m,h))  
    clear()  
    show()  
    db.commit()
```

```
#Function block to delete a student data from the database  
def delete():
```

```
    h=c.get()  
    i=n.get()
```

```
    x.execute("delete from Students_2021 where College_id=? and  
    Name=?",(h,i))  
    show()  
    clear()  
    db.commit()
```

#Function block to show the contents of the database i.e the info of all enrolled students

```
def show():
    x.execute("select * from Students_2021")
    data=x.fetchall()
    if len(data)!=0:
        sd.delete(*sd.get_children())
    for i in data:
        sd.insert("",END,values=i)

    db.commit()
```

#Function block to clear out all the values in the user input information fields

```
def clear():
    c.set("")
    n.set("")
    a.set("")
    d.set("")
    Tr.delete('1.0',END)
    e.set("")
    g.set("")
```

#Function block to get cursor

```
def get_cursor(x):
    crow=sd.focus()
    content=sd.item(crow)
    row=content['values']
    c.set(row[0])
    n.set(row[1])
    a.set(row[2])
    d.set(row[3])
    Tr.delete('1.0',END)
    Tr.insert(END,row[4])
    e.set(row[5])
    g.set(row[6])
```

```
#Creating scrolls along x and y axis  
scroll_x=Scrollbar(fm1,orient=HORIZONTAL)  
scroll_y=Scrollbar(fm1,orient=VERTICAL)
```

```
#Creating a Treeview i.e. a window which shall display the database  
sd=ttk.Treeview(w,columns=('A','B','C','D','E','F','G'),xscrollcommand  
=scroll_x.set,yscrollcommand=scroll_y.set)  
sd.place(x=668,y=100,width=640,height=540)
```

```
#Setting the position and configuraton of the x and y scrolls  
scroll_x.pack(side=BOTTOM,fill=X)  
scroll_y.pack(side=RIGHT,fill=Y)
```

```
scroll_x.configure(command=sd.xview)  
scroll_y.configure(command=sd.yview)
```

```
#Setting headings in the treeview  
sd.heading("A",text='College_id')  
sd.heading("B",text='Name')  
sd.heading("C",text='Age')  
sd.heading("D",text='Department')  
sd.heading("E",text='Address')  
sd.heading("F",text='Email')  
sd.heading("G",text='Gender')  
  
sd['show']='headings'
```

```
#Specifying the width of each field column in the treeview window
sd.column("A",width=150)
sd.column("B",width=250)
sd.column("C",width=150)
sd.column("D",width=150)
sd.column("E",width=300)
sd.column("F",width=250)
sd.column("G",width=150)

sd.bind("<ButtonRelease-1>",get_cursor)

#Button to carry out submit operation
Bs=Button(w,text='Submit',font=('times new
roman',20),fg='red',activebackground='red',activeforeground='white'
,bd=4,relief=RIDGE,padx=10,pady=8,command=submit)
Bs.place(x=5,y=620)

#Button to carry out update operation
Bu=Button(w,text='Update',font=('times new
roman',20),fg='blue',activebackground='blue',activeforeground='whit
e',bd=4,relief=RIDGE,padx=10,pady=8,command=update)
Bu.place(x=120,y=620)

#Button to carry out delete operation
Bd=Button(w,text='Delete',font=('times new
roman',20),fg='green',activebackground='green',activeforeground='
white',bd=4,relief=RIDGE,padx=10,pady=8,command=delete)
Bd.place(x=235,y=620)
```

```
#Button to carry out showing operation  
Bs=Button(w,text='Show',font=('times new  
roman',20),fg='darkorange',activebackground='orange',activeforeground='white',bd=4,relief=RIDGE,padx=18,pady=8,command=show  
)  
Bs.place(x=350,y=620)
```

```
#Button to carry out exit operation  
Be=Button(w,text='Exit',font=('times new  
roman',20),fg='violet',activebackground='violet',activeforeground='white',bd=4,relief=RIDGE,padx=26,pady=8,command=w.destroy)  
Be.place(x=465,y=620)
```

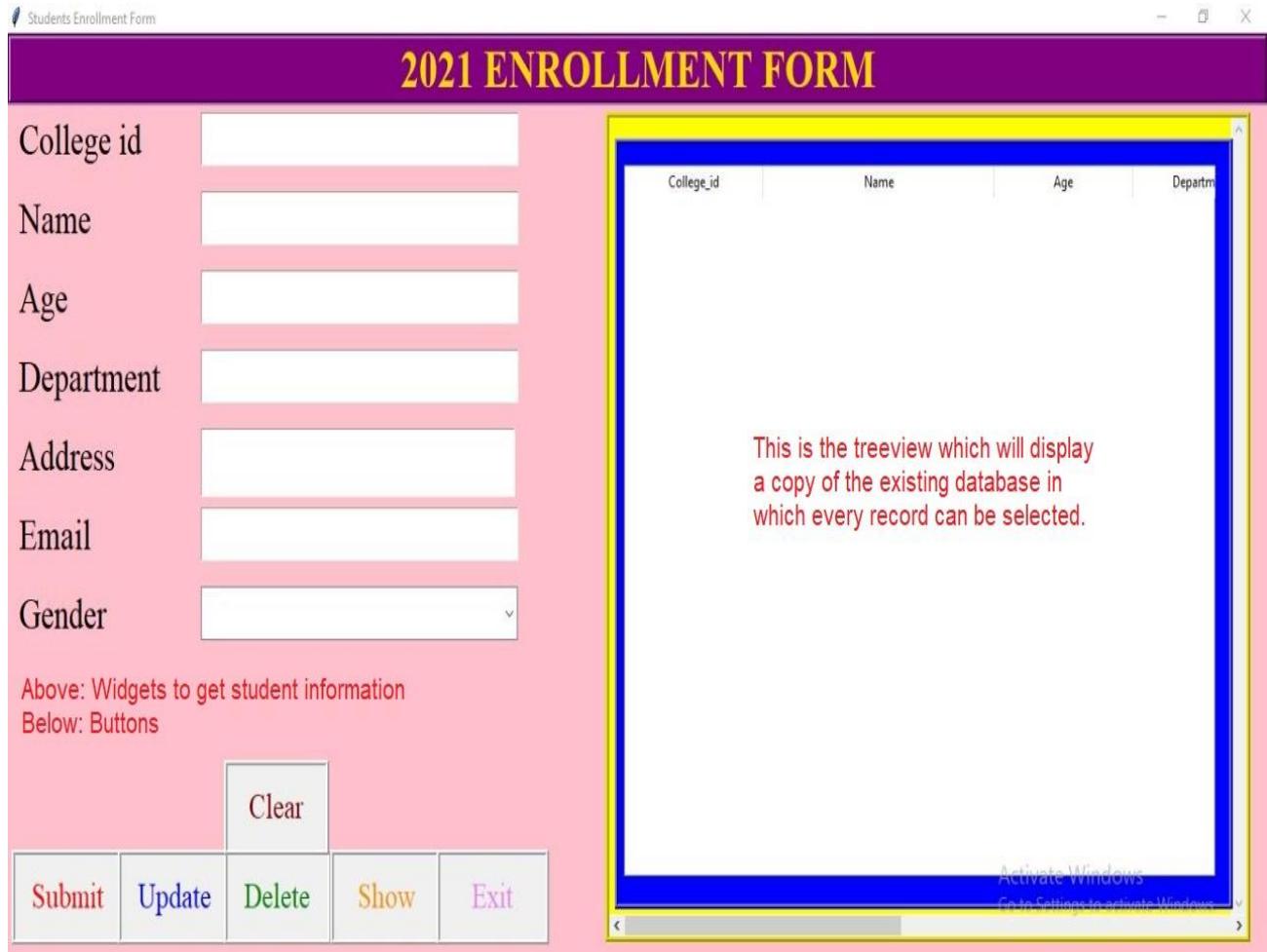
```
#Button to carry out clear operation  
Bc=Button(w,text='Clear',font=('times new  
roman',20),fg='maroon',activebackground='maroon',activeforeground='white',bd=4,relief=RIDGE,padx=15,pady=8,command=clear)  
Bc.place(x=235,y=552)
```

```
w.mainloop()
```

STEP BY STEP

WORKING

After implementation of the code the main window looks like:



Show:

By pressing the show button the already existing records in the database will be displayed.

Students Enrollment Form

2021 ENROLLMENT FORM

| | |
|------------|----------------------|
| College id | <input type="text"/> |
| Name | <input type="text"/> |
| Age | <input type="text"/> |
| Department | <input type="text"/> |
| Address | <input type="text"/> |
| Email | <input type="text"/> |
| Gender | <input type="text"/> |

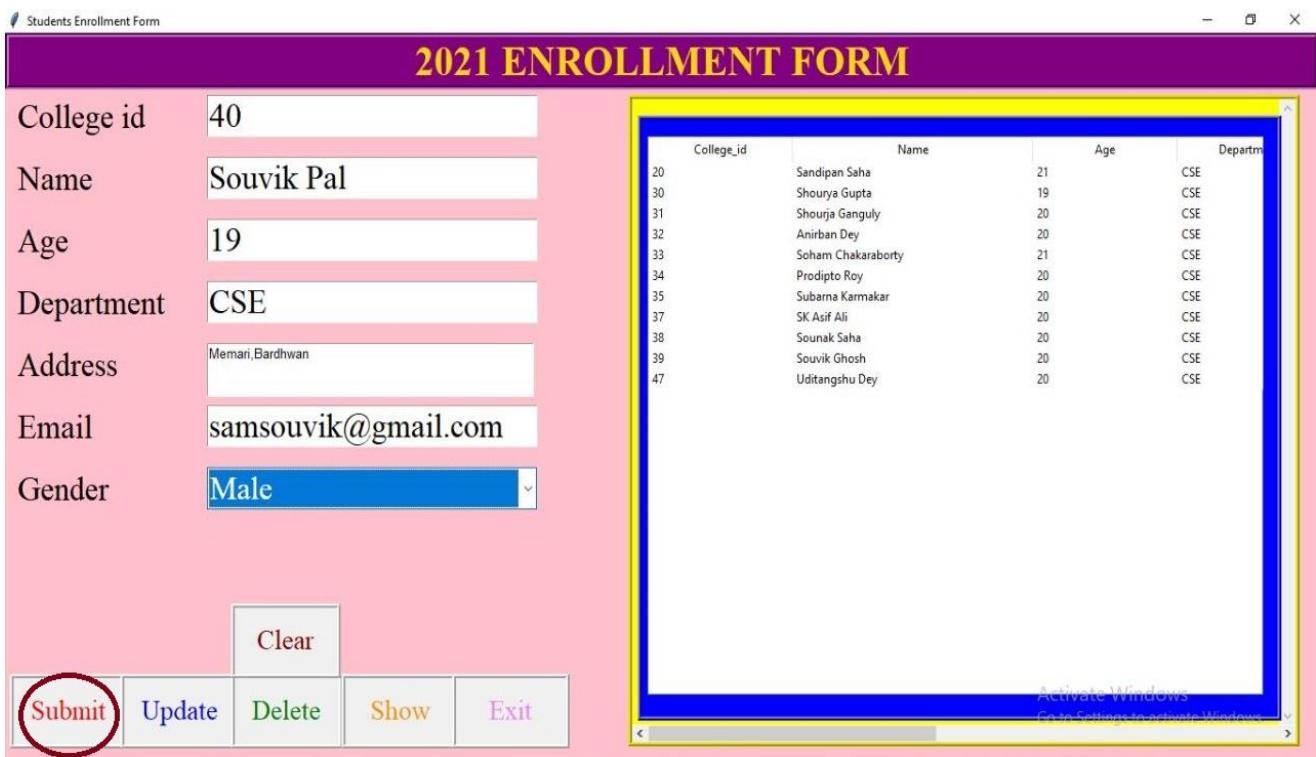
| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 21 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 35 | Subarna Karmakar | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

This is required in case the user wants to confirm if a record already exist in the database or not.

Submit:

When a new record is required to be stored in the database the submit button is to be used.

Step 1: First the user information is provided in the entry fields(constructed by entry, combobox and text).
Step 2: Press the submit button.



The screenshot shows a Windows application window titled "Students Enrollment Form". The main area contains a form with the following fields and their values:

| | |
|------------|---------------------|
| College id | 40 |
| Name | Souvik Pal |
| Age | 19 |
| Department | CSE |
| Address | Memari, Bardhwan |
| Email | samsouvik@gmail.com |
| Gender | Male |

Below the form are five buttons: "Submit" (circled in red), "Update", "Delete", "Show", and "Exit". To the right of the form is a separate window titled "2021 ENROLLMENT FORM" displaying a table of student records:

| College_id | Name | Age | Dept |
|------------|-------------------|-----|------|
| 20 | Sandipan Saha | 21 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 35 | Subarna Karmakar | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

After following the above steps the entry is stored in the database.

| College_id | Name | Age | Department | Address | Email | Gender |
|------------|-----------------------|--------|------------|--------------------------------|-----------------------------|--------|
| Filter | Filter | Filter | Filter | Filter | Filter | Filter |
| 1 | 20 Sandipan Saha | 21 | CSE | Kanchrapara,South 24 Pargan... | sahasandipan@rediffmail.com | Male |
| 2 | 30 Shourya Gupta | 19 | CSE | Kalyani,Nadia... | gshourya2001@gmail.com | Male |
| 3 | 31 Shourja Ganguly | 20 | CSE | Chandannagar,Hooghly... | shourja@gmail.com | Male |
| 4 | 32 Anirban Dey | 20 | CSE | Hooghly... | dey_anib@gmail.com | Male |
| 5 | 33 Soham Chakaraborty | 21 | CSE | Baghmore... | soham_king@gmail.com | Male |
| 6 | 34 Prodipto Roy | 20 | CSE | Guwahati,Assam... | podipto200@gmail.com | Male |
| 7 | 35 Subarna Karmakar | 20 | CSE | Bandel,Hooghly... | subarna2000@gmail.com | Male |
| 8 | 37 SK Asif Ali | 20 | CSE | Serampore,Hooghly... | alibaba@gmail.com | Male |
| 9 | 38 Sounak Saha | 20 | CSE | Ranaghat... | sounak.saha@aot.edu.in | Male |
| 10 | 39 Souvik Ghosh | 20 | CSE | HOOGHLY... | sghosh@rediffmail.com | Male |
| 11 | 40 Souvik Pal | 19 | CSE | Memari,Bardhwan... | samsouvik@gmail.com | Male |
| 12 | 47 Uditangshu Dey | 20 | CSE | Baghmore,Kanchrapara... | dey_uditangshu@gmail.com | Male |

Which is also visible in the treeview as well.

The screenshot shows a Windows application window titled "2021 ENROLLMENT FORM". The form contains fields for "College id", "Name", "Age", "Department", "Address", "Email", and "Gender". Below the form is a "Clear" button. At the bottom are buttons for "Submit", "Update", "Delete", "Show", and "Exit". A red arrow points from the "Gender" field towards a secondary window. This secondary window displays a table with columns "College_id", "Name", "Age", and "Department". The data in the table matches the entries in the main form's table.

| College_id | Name | Age | Department |
|------------|--------------------|-----|------------|
| 20 | Sandipan Saha | 21 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakaraborty | 21 | CSE |
| 34 | Prodipto Roy | 20 | CSE |
| 35 | Subarna Karmakar | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

Update :

By pressing the update button we can alter and modify the entries provided the entry already exists in the database.

Step 1: First an entry is selected from the treeview window.

Step 2: Changes in the entry field(s) are done.

Step 3: Press the update button.

Students Enrollment Form

2021 ENROLLMENT FORM

| | |
|------------|--------------------------------|
| College id | 20 |
| Name | Sandipan Saha |
| Age | 20 |
| Department | CSE |
| Address | Kanchrapara, South 24 Parganas |
| Email | sahasandipan@rediffmail.com |
| Gender | Male |

The first record is selected, age field is modified.

Activate Windows
Go to Settings to activate Windows 10

Clear

Submit Update Delete Show Exit

The record is updated

Students Enrollment Form

2021 ENROLLMENT FORM

| | |
|------------|----------------------|
| College id | <input type="text"/> |
| Name | <input type="text"/> |
| Age | <input type="text"/> |
| Department | <input type="text"/> |
| Address | <input type="text"/> |
| Email | <input type="text"/> |
| Gender | <input type="text"/> |

| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodipto Roy | 20 | CSE |
| 35 | Subarna Karmakar | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Souvik Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

Delete:

The delete button is to be used when there is a necessity to remove a pre-existing record from the database.

Step 1 : The record is to be selected from the treeview window.

Step 2: Press the delete button.

2021 ENROLLMENT FORM

| | |
|------------|-----------------------|
| College id | 35 |
| Name | Subarna Karmakar |
| Age | 20 |
| Department | CSE |
| Address | Bandel, Hooghly |
| Email | subarna2000@gmail.com |
| Gender | Male |

| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 35 | Subarna Karmakar | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

Clear

Submit

Update

Delete

Show

Exit

Activate Windows

Go to Settings to activate Windows

Now after completion of the process the record is removed from the database.

2021 ENROLLMENT FORM

| | |
|------------|--|
| College id | |
| Name | |
| Age | |
| Department | |
| Address | |
| Email | |
| Gender | |

| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

The record with
College_ID- 35 has
been deleted.

Clear

Submit

Update

Delete

Show

Exit

Activate Windows

Go to Settings to activate Windows

Clear:

Clear button is to be used to reset or clear out the entry fields. The necessity may arise if while trying to update or delete a record it is found that the record is already updated or deleted.

Students Enrollment Form

2021 ENROLLMENT FORM

| | |
|------------|-------------------------|
| College id | 47 |
| Name | Uditangshu Dey |
| Age | 20 |
| Department | CSE |
| Address | Baghmore,Kanchrapara |
| Email | dey_uditangshu@gmail.co |
| Gender | Male |

| College_id | Name | Age | Department |
|------------|--------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakaraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

Activate Windows
Go to Settings to activate Windows

2021 ENROLLMENT FORM

College id

Name

Age

Department

Address

Email

Gender

Entry fields are cleared

| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

Activate Windows
Get Settings In-activating Windows

Exit:

By pressing the exit button the Tk window will be closed.

2021 ENROLLMENT FORM

College id

Name

Age

Department

Address

Email

Gender

| College_id | Name | Age | Department |
|------------|-------------------|-----|------------|
| 20 | Sandipan Saha | 20 | CSE |
| 30 | Shourya Gupta | 19 | CSE |
| 31 | Shourja Ganguly | 20 | CSE |
| 32 | Anirban Dey | 20 | CSE |
| 33 | Soham Chakraborty | 21 | CSE |
| 34 | Prodip Roy | 20 | CSE |
| 37 | SK Asif Ali | 20 | CSE |
| 38 | Sounak Saha | 20 | CSE |
| 39 | Souvik Ghosh | 20 | CSE |
| 40 | Souvik Pal | 19 | CSE |
| 47 | Uditangshu Dey | 20 | CSE |

The treeview window is scrollable

| Name | Age | Department | Address |
|------|-----|-------------------------------|---------|
| 20 | CSE | Kanchrapara,South 24 Parganas | |
| 19 | CSE | Kalyani,Nadia | |
| 20 | CSE | Chandannagar,Hooghly | |
| 20 | CSE | Hooghly | |
| 21 | CSE | Baghmore | |
| 20 | CSE | Guwahati,Assam | |
| 20 | CSE | Serampore,Hooghly | |
| 20 | CSE | Ranaghat | |
| 20 | CSE | HOOGHLY | |
| 19 | CSE | Memari,Bardhwan | |
| 20 | CSE | Baghmore,Kanchrapara | |

PROJECT LIMITATIONS

The project has some minor limitations:-

- (i) Submission of a record more than once is prohibited in Database Management System. So in case submit button is pressed twice, an error is displayed but no exception handling has been used.

- (ii) While deleting the records, even after all the records are deleted, the last record that is deleted is still displayed in the treeview. It still remains. This problem is fixed by closing the Tk window and opening it again.

FUTURE SCOPE

The project is based on “Student Management System” . Basically a database of students records, the main aim is to construct a user friendly interface and also ensure easy retrievability of all records. Keeping that in mind the concepts of this project can be used in future to make strong and efficient databases capable of storing huge datasets. Also further information fields can be added in extra to the existing ones, in the process of transforming it into a broader scale. Overall the project aims to make Data Handling easier, simpler and faster.



SUMMARY

This documentation outlines the procedure of implementing Student Management System by using Python sqlite3 and tkinter module. The steps involved in the procedure –

- Learning and doing research on the topic.
- Implementing the Project.
- Testing and assessing the scope of the Project.
- Presenting the project report.

Even though implemented for managing student records, the project can be further modified and used in other sectors, business organisation , health sectors etc., as data handling is an important task for every organisation. This project aims at developing a strong database where records can be stored and retrieved with ease with minimal human intervention.



BIBLIOGRAPHY

- <https://www.wikipedia.org/>
- <https://www.programiz.com/>
- <https://www.geeksforgeeks.org/>
- <https://stackoverflow.com/>
- <https://www.w3schools.com/>
- <https://youtu.be/tUc6FMPsZDg>
- <https://youtu.be/287lnVXph6o>



*~:Thank
You:-*