**System Implementations**

**Recommended System Requirements**

Processors: Intel® Core™ i3 processor 4300M at 2.60 GHz.

Disk space: 4 to 8 GB.

Operating systems: Windows® 10, MACOS, and UBUNTU.

Python Versions: 3.X.X or Higher.

**Minimum System Requirements**

Processors: Intel Atom® processor or Intel® Core™ i3 processor.

Disk space: 1 GB.

Operating systems: Windows 7 or later, MACOS, and UBUNTU.

Python Versions: 2.7.X, 3.9.X.

**ACKNOWLEDGEMENT**TTT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully.

We would like to express my deep and sincere gratitude to my subject teacher, **Mr. Amit Udiwal**, for giving me the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision, sincerity and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as and honour to work and study under his guidance. We are very much thankful to our **Sr. Jasmin** for giving valuable time and moral support to develop this software. We would like to take opportunity to extend my sincere thanks and gratitude to our parents for being a source of inspiration and providing time and freedom to develop this software project. We also feel indebted to my friends for the valuable suggestions during the project work.

Divya Anjana

[Roll No.

Class XII

**CERTIFICATE**

This is to certify that the project on ‘School Management System’ is a work done by Divya Anjana fulfilment of CBSE’S AISSCE EXAMINATION 2022-23 and has been carried out under my direct supervision and guidance. This report or a similar report on the topic has not been submitted for any other examination and does not form any other examination and does not form any other course undergone by the candidate.

Name: Divya Anjana[Roll No.

………………….

Signature of Teacher / Guide

Name: Mr. Amit Udiwal

Designation:

………………. ….………………

**REFERENCE**

The order to work on this project on ‘School Management System’ the following books & literature are referred by me during the various phrases of department of the project.

• http://www.python.org/.

• http://www.itsourcecode.org/.

• http://www.wikipedia.org/.

• Informatics Practices for Class XII

- By Sumita Arora

• Together with informatics practices.

Other than the above mentioned books, the suggestions and supervision of my teacher and my class experience also helped me to develop this software project.

**Introduction**

The School Management System is a collection of technologies that enables the organization to run the institute efficiently. The first major benefit is that as a school, you can keep a proper record or track of student data.

In schools, where thousands of sensitive and vital data must be securely maintained, school management systems facilitate the flow of information. Students can simply update and retrieve their information online with adequate database management.

**Objective and**

**Scope of The Project**

The main objective of the Python Project on School Management System is to Manage all of the data via traditional methods is a difficult and time-consuming undertaking. You can lessen your workload by using school administration software. You will be able to complete your work more quickly this way. You can save time and effort by using this digital resource.

***Functions:***

* Add Students
* List all students
* Search students
* Remove Students

**School**

**Management System**

from tkinter import \*

from tkinter import ttk

import sqlite3

import tkinter.messagebox

from datetime import date

from tkinter import filedialog

import shutil

import os

from tkinter import Text,Tk

today=date.today()

print ('software is runing......')

firstw=Tk()

firstw.title("IT SOURCECODE")

firstw.geometry("1600x1000+0+0")

label=Label(text="SIMPLE SCHOOL MANAGEMENT SYSTEM IN PYTHON",font=("times new roman",35),bg="black",fg="white")

label.pack(side=TOP ,fill=X)

user1=Label(text="USERNAME",font=("arial",23))

user1.place(x=610,y=120)

user=Entry(width=17,bd=5,font=("arial",20))

user.place(x=570,y=200)

label.pack(side=TOP ,fill=X)

user2=Label(text="PASSWORD",font=("arial",23))

user2.place(x=610,y=280)

user3=Entry(width=17,show="\*",bd=5,font=("arial",20))

user3.place(x=570,y=360)

def second():

global secondw

secondw=Tk()

secondw.title("IT SOURCECODE")

secondw.geometry("1600x1000+0+0")

def distroy4():

secondw.destroy()

root()

def student():

student1=Tk()

student1.title("STUDENT DETAILS")

def studentid():

rot = Tk()

rot.title("IT SOURCECODE")

rot.geometry("1600x1000+0+0")

mainlabel = Label(rot, text="STUDENT DETAILS", font=("times new roman", 35), bg="black",fg="white")

mainlabel.pack(side=TOP, fill=X)

chat1 = ttk.Treeview(rot,height=20, columns=('name','sur','fee'), selectmode="extended")

chat1.heading('#0', text='ID', anchor=CENTER)

chat1.heading('#1', text=' NAME', anchor=W)

chat1.heading('#2', text='FEE', anchor=W)

chat1.heading('#3', text="LAST NAME", anchor=W)

chat1.column('#1', stretch=YES, minwidth=50, width=100)

chat1.column('#3', stretch=YES, minwidth=50, width=100)

chat1.column('#2', stretch=YES, minwidth=50, width=100)

chat1.column('#0', stretch=YES, minwidth=50, width=70)

chat1.place(x=470, y=130)

ttk.Style().configure("Treeview", background="black", foreground="coral1")

ttk.Style().configure("Treeview.Heading", background="blue", foreground="palevioletRed1")

rot.configure(background='white')

vsb=ttk.Scrollbar(rot, orient="vertical",command=chat1.yview)

vsb.place(x=827,y=150,height=400+20)

chat1.configure(yscrollcommand=vsb.set)

conn = sqlite3.connect("details.db")

with conn:

cur = conn.cursor()

cur.execute('SELECT id ,name, fee , sur FROM kistar ')

for row1 in cur.fetchall():

chat1.insert('', 0, text=row1[0], values=(row1[1] ,row1[2],row1[3]))

def viewenquiry2():

rt = Tk()

rt.title("IT SOURCECODE")

rt.geometry("1600x1000+0+0")

mainlabel =Label(rt, text="VISITOR", font=("times new roman", 35), bg="black",fg="white")

mainlabel.pack(side=TOP, fill=X)

chat1 = ttk.Treeview(rt,height=20 , columns=('EMAIL', 'ENQUIRY', 'DATE'), selectmode="extended")

chat1.heading('#0', text='NAME', anchor=CENTER)

chat1.heading('#1', text='EMAIL', anchor=CENTER)

chat1.heading('#2', text='ENQUIRY', anchor=CENTER)

chat1.heading('#3', text="DATE", anchor=CENTER)

chat1.column('#1', stretch=YES, minwidth=50, width=100)

chat1.column('#3', stretch=YES, minwidth=50, width=100)

chat1.column('#2', stretch=YES, minwidth=50, width=300)

chat1.column('#0', stretch=YES, minwidth=50, width=70)

vsb = ttk.Scrollbar(rt, orient="vertical", command=chat1.yview)

vsb.place(x=955, y=170, height=400 + 20)

chat1.configure(yscrollcommand=vsb.set)

chat1.place(x=400, y=170)

ttk.Style().configure("Treeview", background="#383838", foreground="coral1")

ttk.Style().configure("Treeview.heading", background="blue", foreground="palevioletRed1")

rt.configure(background="white")

conn = sqlite3.connect("details.db")

with conn:

cur = conn.cursor()

cur.execute('SELECT \* FROM golu')

for row in cur.fetchall():

chat1.insert('', 0, text=row[0], values=(row[1], row[2], row[3]))

def distroy5():

secondw.destroy()

window()

mainlabel= Label(secondw,text="SIMPLE SCHOOL MANAGEMENT SYSTEM IN PYTHON", font=("times new roman", 35), bg="black",fg="white")

mainlabel.pack(side=TOP, fill=X)

button = Button(secondw,width=15, font=("arial", 20), text="REGISTRATION", bg="black",fg="white", command=distroy4)

button.place(x=10, y=480)

enquiry = Button(secondw, width=15, font=("arial", 20), text="FEE DETAILS", bg="black",fg="white",command=distroy5)

enquiry.place(x=280, y=480)

fee\_details = Button(secondw, width=15, font=("arial", 20), text="ENQUIRY", bg="black",fg="white",command=enquiry1)

fee\_details.place(x=560, y=480)

viewenquiry= Button(secondw, width=15, font=("arial", 20), text="VIEW ENQUIRY", bg="black",fg="white",command=viewenquiry2)

viewenquiry.place(x=840, y=480)

viewenquiry1 = Button(secondw, width=15, font=("arial", 20), text="STUDENT DETAILS", bg="black",fg="white",command=studentid)

viewenquiry1.place(x=1100, y=480)

def distroy():

firstw.destroy()

def login():

if user.get()=="1" and user3.get()=="1":

second()

distroy()

else:

t = tkinter.messagebox.showinfo("INVALID USERNAME OR PASSWORD ", "YOU HAVE ENTERED INVALID USERNAME OR PASSWORD ")

user.delete(0,END)

user3.delete(0,END)

def root():

root=Tk()

root.geometry("1600x1000+0+0")

root.title("IT SOURCECODE")

global entry1

global entry2

global entry3

global entry4

global entry5

global box

global name

global radio1

global radio2

name = StringVar()

global sur

sur = StringVar()

global gander

gander = IntVar()

global var1

var1 = IntVar()

global var2

var2 = IntVar()

global branch

branch = StringVar()

global rollno

rollno = StringVar()

global email

email = StringVar()

global course

course = StringVar()

global python

python = IntVar()

global java

java = IntVar()

global c

c = IntVar()

global d

d = IntVar()

global calculate

calculate = StringVar()

id = IntVar()

search = IntVar()

NAME = name.get()

SUR = sur.get()

EMAIL = email.get()

BRANCH = branch.get()

GANDER = gander.get()

PYTHON = python.get()

JAVA = java.get()

C = c.get()

D = d.get()

CALCULATE = calculate.get()

calculation2 = 2000

label=Label(root,text="REGISTRATION FORM", font=("arial",25), bg="black",fg="white")

label.pack(side=TOP, fill=X)

label1 =Label(root,text="NAME:", font=("arial",17))

label1.place(x=300, y=150)

label2=Label(root,text="SURNAME:", font=("arial",17))

label2.place(x=300, y=210)

label3=Label(root,text="EMAIL:", font=("arial",17))

label3.place(x=300, y=270)

label3=Label(root,text="GENDER:", font=("arial",17))

label3.place(x=300, y=330)

label4=Label(root,text="COURSE:", font=("arial",17))

label4.place(x=300, y=390)

label4=Label(root,text="BRANCH", font=("arial",17))

label4.place(x=300, y=450)

label4=Label(root,text="TOTAL FEE", font=("arial",17))

label4.place(x=300, y=520)

entry5=Entry( root, textvar=calculate,state="readonly",width=20,font=("arial",15,"bold") ,bd=5)

entry5.place(x=500, y=515)

entry1=Entry(root,bd=5, width=20,textvar=name ,font=("arial",15))

entry1.place(x=500,y=150)

#entry22=Entry(main,bd=5, width=20,textvar=sam ,font=("arial",15))

#entry22.place(x=500,y=150)

entry2=Entry(root,bd=5, width=20, textvar=sur ,font=("arial",15))

entry2.place(x=500,y=210)

entry3=Entry(root,bd=5, width=20,textvar=email ,font=("arial",15))

entry3.place(x=500,y=270)

entry4=Entry(root,bd=5, text="enter roll no.",width=20,textvar=search ,font=("arial",15))

entry4.place(x=800,y=150)

search.set("")

# entry4=Entry(root,bd=5, text="enter roll no.",width=20,textvar=search ,font=("arial",15))

# entry4.place(x=800,y=150)

radio1=Radiobutton(root,text="MALE", variable=gander, value=1 ,font=("arial",13))

radio1.place(x=515, y=340)

radio2=Radiobutton(root,text="FEMALE", variable=gander, padx=20, value=0 ,font=("arial",13))

radio2.place(x=590, y=340)

gander.set(3)

box=ttk.Combobox(root,textvariable=branch,state="readonly", font=("arial",12,"bold"),width=22)

box['values']=['SELECT','COMPUTER SCIENCE','MACHENICAL','CIVIL','IT']

box.current(0)

box.place(x=503,y=395)

checkbutton1=Checkbutton(root,text="JAVA",variable=java)

checkbutton1.place(x=502,y=455 )

checkbutton1=Checkbutton(root,text="C",variable=c)

checkbutton1.place(x=555,y=455 )

checkbutton1=Checkbutton(root,text="C++",variable=d)

checkbutton1.place(x=600,y=455 ,)

checkbutton1=Checkbutton(root,text="PYTHON",variable=python)

checkbutton1.place(x=650,y=455)

python.set(0)

java.set(0)

c.set(0)

d.set(0)

def dis():

root.destroy()

second()

#buttton

button1=Button(root,text="CALCULATE FEE",width=14,font=("arial",10),bg="black",fg="white" ,command=calculation)

button1.place(x=530 , y=630)

button12 = Button(root, text="BACK", width=17, font=("arial", 17), bg="red",fg="black",command=dis )

button12.place(x=0, y=0)

button2=Button(root,text="SUBMIT FORM",width=14,font=("arial",10),bg="black",fg="white",command= msg )

button2.place(x=660 , y=630)

button3=Button(root,text="RESET",width=14,font=("arial",10),bg="black",fg="white",command= golu )

button3.place(x=395 , y=630)

button4=Button(root,text="SEARCH",width=14,font=("arial",10),bg="black",fg="white" ,command=all )

button4.place(x=1100 , y=150)

#button7 = Button(root, text="UPLOAD PHOTO", width=14, font=("arial", 10), bg="indianred1",command=file)

#button7.place(x=1100, y=210)

button4=Button(root,text="UPDATE",width=14,font=("arial",10),bg="black",fg="white" ,command=update)

button4.place(x=950 , y=630)

button5=Button(root,text="DELETE",width=14,font=("arial",10),bg="black",fg="white",command=delete )

button5.place(x=800 , y=630)

#button6=Button(root,text="ENQUIRY",width=14,font=("arial",10),bg="indianred1",command=window )

#button6.place(x=300 , y=630)

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

# cur.execute('CREATE TABLE IF NOT EXISTS kistar(id INTEGER primary key autoincrement ,name text,sur text,email, branch text,gander text, fee int, python int,java int,c int,d int)')

# cur.execute('CREATE TABLE IF NOT EXISTS golu (NAME TEXT, PHONE INT ,PURPOSE TEXT,DATE)')

# cur.execute('CREATE TABLE IF NOT EXISTS FEEINSTALLMENT (id int ,TOTEL FEE INT, REMAIN FEE INT, PAID FEE INT ,INSTALLMENT INT,DATE)')

def ka():

NAMEE=entry23.get()

PHONE=entry24.get()

PURPOSE=box2.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('INSERT INTO golu(NAME,PHONE,PURPOSE,DATE)VALUES(?,?,?,?)',(NAMEE,PHONE,PURPOSE,today,))

conn.commit()

def r():

j()

ka()

def enquiry1():

enquiry1=Tk()

enquiry1.title("ENQUIRY")

enquiry1.geometry("1600x1000+0+0")

purpose=StringVar()

global entry23

global entry24

global box2

def enquiry1destroy():

enquiry1.destroy()

second()

label22 = Label(enquiry1, text="ENQUIRY", font=("arial", 25), bg="black",fg="white")

label22.pack(side=TOP, fill=X)

label1 = Label(enquiry1, text="NAME:", font=("arial", 17))

label1.place(x=300, y=150)

label2 = Label(enquiry1, text="PHONE NO.:", font=("arial", 17))

label2.place(x=300, y=210)

label3 = Label(enquiry1, text="PURPOSE:", font=("arial", 17))

label3.place(x=300, y=270)

entry23 = Entry(enquiry1, bd=5, width=20, font=("arial", 15))

entry23.place(x=500, y=150)

button = Button(enquiry1, text="submit", width=30, bg="black",fg="white", command=r)

button.place(x=500, y=320)

button1=Button(enquiry1, text="<< BACK", width=30, bg="red",fg="black",command=enquiry1destroy)

button1.place(x=0,y=0)

entry24 = Entry(enquiry1, bd=5, width=20, font=("arial", 15))

entry24.place(x=500, y=210)

box2 = ttk.Combobox(enquiry1, textvariable=purpose, state="readonly", font=("arial", 12, "bold"), width=22)

box2['values'] = ['SELECT', 'TO LEARN PROGRAMMING', 'TO LEARN MACHINE LEARNING', 'FEE DETAILS']

box2.current(0)

box2.place(x=500, y=270)

def cat():

z = IntVar()

FE = entry25.get()

x = entry26.get()

y = entry29.get()

FE=entry25.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT fee FROM kistar WHERE id=?',(FE,))

for row24 in cur.fetchall():

entry26.configure(state="normal")

entry26.delete(0, END)

entry26.insert(0,row24)

entry26.configure(state="disable")

cur.execute(' SELECT SUM(INSTALLMENT) FROM FEEINSTALLMENT WHERE id=? GROUP BY id ', (FE,))

for row23 in cur.fetchall():

entry27.delete(0, END)

entry27.insert(0, row23)

ye = entry27.get()

z = int(float((entry26.get()))) - int(float((entry27.get())))

#cur.execute('INSERT INTO FEEINSTALLMENT(id , TOTEL,INSTALLMENT,PAID ,REMAIN, DATE)VALUES(?,?,?,?,?,?)',(FE, x, y, ye, z, today,))

entry28.configure(state="normal")

entry28.delete(0, END)

entry28.insert(0, z)

print(row23)

entry27.configure(state="disable")

entry26.configure(state="disable")

entry28.configure(state="disable")

conn.commit()

print(x)

print(FE)

print(today)

def reset2():

entry26.configure(state="normal")

entry25.configure(state="normal")

#entry24.configure(state="normal")

entry27.configure(state="normal")

entry28.configure(state="normal")

entry29.configure(state="normal")

entry26.delete(0,END )

entry25.delete(0, END)

entry27.delete(0,END)

entry28.delete(0,END)

entry29.delete(0,END)

#box2.set("SELECT")

entry27.configure(state="disable")

entry26.configure(state="disable")

entry28.configure(state="disable")

def fee\_add():

z=IntVar()

FE=entry25.get()

x=entry26.get()

y=entry29.get()

entry27.configure(state="normal")

entry28.configure(state="normal")

entry26.configure(state="normal")

cur.execute('INSERT INTO FEEINSTALLMENT(id , TOTEL,INSTALLMENT, DATE)VALUES(?,?,?,?)', (FE, x,y, today,))

cur.execute(' SELECT SUM(INSTALLMENT) FROM FEEINSTALLMENT WHERE id=? GROUP BY id ',(FE,))

for row23 in cur.fetchall():

entry27.delete(0,END)

entry27.insert(0,row23)

ye=entry27.get()

z=int(float((entry26.get())))-int(float((entry27.get())))

cur.execute('UPDATE FEEINSTALLMENT SET PAID=? WHERE id=?' , (ye,FE,))

cur.execute('UPDATE FEEINSTALLMENT SET REMAIN=? WHERE id=?',(z,FE,))

entry28.configure(state="normal")

entry28.delete(0,END)

entry28.insert(0,z)

print(row23)

entry27.configure(state="disable")

entry26.configure(state="disable")

entry28.configure(state="disable")

conn.commit()

print(x)

print(FE)

print(today)

def installment2():

if int(entry29.index("end"))>int(0):

fee\_add()

else:

x=tkinter.messagebox.showinfo("NO FEE ADDED","YOU HAVE NOT ADDED ANY FEE ")

def j():

PURPOSE=box2.get()

print(PURPOSE)

def r():

j()

ka()

def window():

global main

global namee

global phone

global purpose

global entry23

global entry24

global entry25

global entry26

global entry27

global entry28

global box2

global key

global fee3

global KEY

global ley

global sey

global ADDFEE

global entry29

#entry29=IntVar()

#entry26=IntVar()

#entry27=IntVar()

#key=StringVar()

#fee3=StringVar()

#ADDFEE=IntVar()

main=Tk()

main.geometry("1600x1000+0+0")

main.title("Enquiry")

namee=StringVar()

phone=IntVar()

purpose=StringVar()

fe=StringVar()

key=IntVar()

ley=StringVar()

sey=StringVar()

#NAMEE=namee.get()

#PHONE=phone.get()

#PURPOSE=purpose.get()

def distroy3():

main.destroy()

second()

button = Button(main, text="BACK", width=30, bg="red",fg="black", command=distroy3)

button.place(x=0, y=0)

label3=Label(main,text="ENTER STUDENT ID", font=("arial",17))

label3.place(x=400, y=100)

label3 = Label(main, text="ENTER AMOUNT", font=("arial", 17))

label3.place(x=650, y=100)

button22=Button(main,text="LOGIN",width=26,font=("arial",10),bg="black",fg="white",command=cat )

button22.place(x=400, y=310)

button23=Button(main,text="ADD FEE",width=26,font=("arial",10),bg="black",fg="white",command=installment2 )

button23.place(x=650 , y=310)

entry29=Entry(main,bd=5, width=20 ,font=("arial",15))

entry29.place(x=650,y=200)

button28 = Button(main, text="RESET", width=26, font=("arial", 10), bg="red",fg="black", command=reset2)

button28.place(x=1150,y=0)

label31=Label(main,text="TOTEL FEE", font=("arial",17))

label31.place(x=900, y=550)

label32=Label(main,text="PAID FEE", font=("arial",17))

label32.place(x=600, y=550)

label33=Label(main,text="REMAIN FEE", font=("arial",17))

label33.place(x=300, y=550)

entry25=Entry(main,bd=5, width=20 ,font=("arial",15))

entry25.place(x=400,y=200)

entry26=Entry(main,bd=5, width=20 ,font=("arial",15))

entry26.place(x=900,y=600)

entry27=Entry(main,bd=5, width=20 ,font=("arial",15))

entry27.place(x=600,y=600)

entry28=Entry(main,bd=5, width=20 ,font=("arial",15))

entry28.place(x=300,y=600)

#entry27=Entry(main,bd=5,textvariable=fee3, state="readonly", width=20 ,font=("arial",15))

#entry27.place(x=960,y=400)

#entry28=Entry(main,bd=5, width=20 ,font=("arial",15))

#entry28.place(x=900,y=400)

#function

calculation2=2000

def calculation():

NAME = entry1.get()

SUR = entry2.get()

EMAIL = entry3.get()

BOX = box.get()

GANDER = gander.get()

PYTHON = python.get()

JAVA = java.get()

C = c.get()

D = d.get()

print(PYTHON)

print(GANDER)

CALCULATE = calculate.get()

if NAME==("") and SUR==("")and EMAIL==("") and BOX==("SELECT") and GANDER==(3) and JAVA==(0) and PYTHON==(0) and C==(0) and D==(0):

kal=tkinter.messagebox.showinfo(" DETAILS INVALID","FILL ALL THE DETAILS")

else:

global x

if box.get()=="COMPUTER SCIENCE" and gander.get()==0:

x=(calculation2-calculation2\*20/100)

entry5.configure(state="normal")

entry5.delete(0,END)

entry5.insert(0,x)

entry5.configure(state="disable")

if box.get()=="COMPUTER SCIENCE" and gander.get()==1:

x=(calculation2-calculation2\*10/100)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

if box.get()=="MACHENICAL" and gander.get()==1:

x=(calculation2)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

if box.get()=="MACHENICAL" and gander.get()==0:

x=(calculation2-calculation2\*10/100)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

if box.get()=="IT" and gander.get()==0:

x=(calculation2-calculation2\*10/100)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

if box.get()=="CIVIL" and gander.get()==1:

x=(calculation2)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

if box.get()=="CIVIL" and gander.get()==0:

x=(calculation2-calculation2\*10/100)

entry5.configure(state="normal")

entry5.delete(0, END)

entry5.insert(0, x)

entry5.configure(state="disable")

def msg():

if branch.get()=="SELECT" or gander.get()==3 or ( python.get()==0 and java.get==0 and c.get()==0 and d.get()==0):

calculate.set("PLESE FILL ALL")

if "@" and ".com" not in entry3.get() :

kal=tkinter.messagebox.showinfo(" INVALID DETAILS","ENTER VALID EMAIL ADDRESS")

entry3.delete(0,END)

else:

msg=tkinter.messagebox.askyesno("Form filling confarmation"," WARNING: All data will be erase after 'YES' for new applicant" )

if msg>0:

NAME=entry1.get()

SUR=entry2.get()

EMAIL=entry3.get()

BRANCH=box.get()

GANDER=gander.get()

PYTHON=python.get()

JAVA=java.get()

C=c.get()

D=d.get()

CALCULATE=calculate.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('INSERT INTO kistar (name,sur, email, branch, gander,fee ,python,java,c,d ) VALUES(?,?,?,?,?,?,?,?,?,?)',(NAME,SUR,EMAIL,BRANCH,GANDER,CALCULATE,PYTHON,JAVA,C,D,))

golu()

def golu():

entry1.delete(0,END)

entry2.delete(0,END)

entry3.delete(0,END)

box.set("SELECT")

gander.set(3)

python.set(0)

java.set(0)

c.set(0)

d.set(0)

calculate.set("")

entry4.delete(0,END)

def search\_id():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT name FROM kistar WHERE id=?',(SEARCH))

for row1 in cur.fetchone():

name.set(row1)

def search\_sur():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT sur FROM kistar WHERE id=?',(SEARCH,))

for row2 in cur.fetchone():

sur.set(row2)

def search\_email():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT email FROM kistar WHERE id=?',(SEARCH,))

for row3 in cur.fetchone():

email.set(row3)

def search\_branch():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT branch FROM kistar WHERE id=?',(SEARCH,))

for row4 in cur.fetchone():

branch.set(row4)

def search\_gander():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT gander FROM kistar WHERE id=?',(SEARCH,))

for row5 in cur.fetchone():

gander.set(row5)

def search\_course():

SEARCH=entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('SELECT python FROM kistar WHERE id=?',(SEARCH,))

for row6 in cur.fetchone():

python.set(row6)

cur.execute('SELECT java FROM kistar WHERE id=?',(SEARCH,))

for row7 in cur.fetchone():

java.set(row7)

cur.execute('SELECT c FROM kistar WHERE id=?',(SEARCH,))

for row8 in cur.fetchone():

c.set(row8)

cur.execute('SELECT d FROM kistar WHERE id=?',(SEARCH,))

for row9 in cur.fetchone():

d.set(row9)

cur.execute('SELECT fee FROM kistar WHERE id=?',(SEARCH,))

for row10 in cur.fetchone():

calculate.set(row10)

def update():

box1=tkinter.messagebox.askyesno("CONFIRMATION","if you update you will be unable to see previous data again")

if box1>0:

SEARCH=entry4.get()

NAME=entry1.get()

SUR=entry2.get()

EMAIL=entry3.get()

BRANCH=box.get()

GANDER=gander.get()

PYTHON=python.get()

JAVA=java.get()

C=c.get()

D=d.get()

CALCULATE=entry5.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute('UPDATE kistar SET name=? WHERE id=?',(NAME,SEARCH,))

cur.execute('UPDATE kistar SET sur=? WHERE id=?',(SUR,SEARCH,))

cur.execute('UPDATE kistar SET email=? WHERE id=?',(EMAIL,SEARCH,))

cur.execute('UPDATE kistar SET branch=? WHERE id=?',(BRANCH,SEARCH,))

cur.execute('UPDATE kistar SET gander=? WHERE id=?',(GANDER,SEARCH,))

cur.execute('UPDATE kistar SET python=? WHERE id=?',(PYTHON,SEARCH,))

cur.execute('UPDATE kistar SET java=? WHERE id=?',(JAVA,SEARCH,))

cur.execute('UPDATE kistar SET c=? WHERE id=?',(C,SEARCH,))

cur.execute('UPDATE kistar SET d=? WHERE id=?',(D,SEARCH,))

conn.commit()

def delete():

box=tkinter.messagebox.askyesno("WARNING","DATA WILL NOT BE RECOVER AGAIN")

if box>0:

SEARCH = entry4.get()

conn=sqlite3.connect("details.db")

with conn:

cur=conn.cursor()

cur.execute("DELETE FROM kistar WHERE id=?",(SEARCH))

conn.commit()

golu()

def all():

search\_id()

search\_sur()

search\_email()

search\_branch()

search\_gander()

search\_course()

INQUIRY=Button(text="LOGIN",width=17,font=("arial",20),bg="black",fg="white",command=login )

INQUIRY.place(x=560 , y=480)

firstw.mainloop()