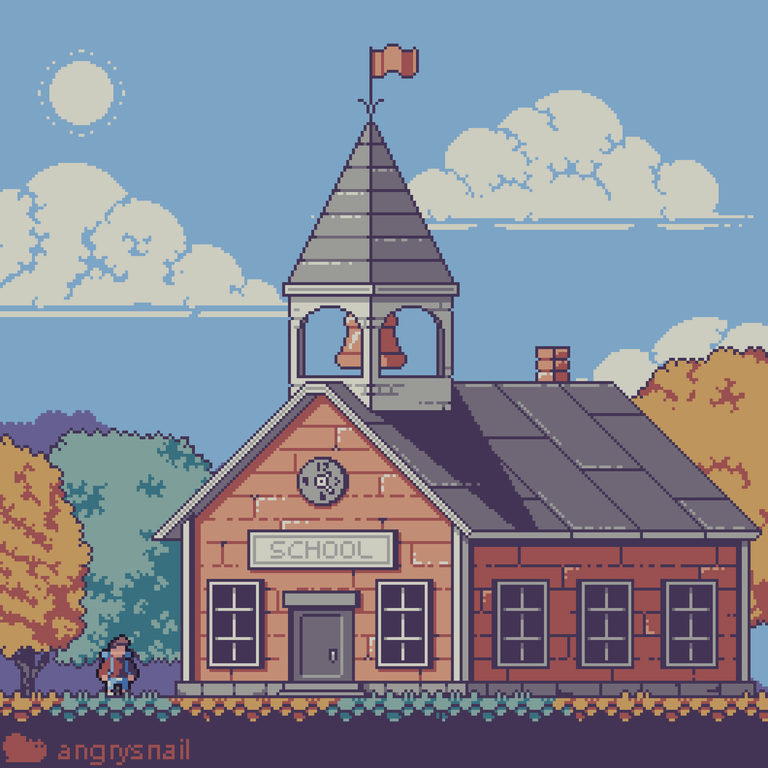
**PROJECT**

**STUDEnt management System.**



**BY**

**M. SAAD KHAN**

**CLASS -12 - H**

**Certificate**

**AMITY INTERNATIONAL SCHOOL MAYUR VIHAR**



**NAME**- M. SAAD KHAN (ROLL NO. )

**CLASS**- XII-H

**THIS IS TO CERTIFY THAT THE ABOVE MENTIONED STUDENT HAS SUCCESSFULLY COMPLETED THEIR COMPUTER SCIENCE PROJECT UNDER MY SUPER VISION ACCORDING TO CBSE GUIDANCE.**

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

**MS. DEEPSHIKHA SETHI**

**ACKNOWLEDGEMENT**

I acknowledge the valuable contribution of **Ms. Deepshikha Sethi** in providing me the proper guidance to complete the computer project. The project would not have been completed without her support and kind help.

**INTRODUCTION TO PROJECT**

**“STUDENT INFORMATION MANAGEMENT SYSTEM”**

This project on “student information management system” is useful for easy user interface. The system utilizes the powerful database management , data retrieval and data manipulation . The project provides more ease for managing the data than manually maintaining in the documents. The project is useful for saving valuable time and reduces the huge paper work.

It will help educational institutions like schools and colleges will keep track of their student records like personal details , marks details, etc . The internet is rapidly becoming a part of the everyday lives of a majority of people in the world. People perform various activities on the internet and one of them is storing their data-base where they are interested in. in these data base’s they can post the queries and they can retrieve the required data. Obviously there is a need of student information system software for management of student’s data.

**Modules used**

**TKINTER : Used to create the graphical user interface(GUI).**

**TKINTER.MESSAGEBOX: Used to create message boxes for confirmation of certain choices. These message boxes return a subset of (True, False, Yes, No, OK, None).**

**PYMYSQL: Used to execute MySQL commands in Python. It is a pure-Python MySQL client.**

**DATABASE AND SQL QUERIES**

**'''**

**Create database record;**

**Use record;**

**Create table student**

**(Roll int primary key,**

**Name varchar(30),**

**Marks int);**

**'''**

**Main Source Code**

**from tkinter import \***

**import tkinter.messagebox as tkmb**

**import pymysql as msc**

**''' User = GROUP1 and password = 12345'''**

**con=msc.connect(host="localhost",user="root",passwd="divkin3118",database="record")**

**#This above code is used to connect Python to Database.**

**cursor=con.cursor()**

**window = Tk()**

**window.geometry('400x250')**

**#The above code is used to give dimension to our window screen.**

**window.title("Student Management Software 1.0")**

**# The above code is used to give title to our window screen.**

**window.withdraw() #makes window disappear but program keeps running**

**oldRoll=0**

**# It is used to store roll no.**

**windowlg = Tk()**

**windowlg.geometry('250x100')**

**#The above code is used to give dimension to our window screen.**

**windowlg.title("User Login")**

**# The above code is used to give title to our window screen.**

**lblUser = Label(windowlg, text="User Name")**

**#Used for labelling**

**lblUser.grid(column=0, row=0)**

**#The above code is used to give postion to the dialog box.**

**lblPwd = Label(windowlg, text="Password")**

**#Used for labelling**

**lblPwd.grid(column=0, row=1)**

**#The above code is used to give postion to the dialog box.**

**txtUser =Entry(windowlg,width=10)**

**#Used for Entering the data**

**txtUser.grid(column=2,row=0)**

**#The above code is used to give postion to the dialog box.**

**txtPwd =Entry(windowlg,show="\*",width=10)**

**#Used for Entering the data**

**txtPwd.grid(column=2,row=1)**

**#The above code is used to give postion to the dialog box.**

**def btnLoginClicked():**

**'''**

**\* This Function is used to check the Login Details are matched to the database**

**\* If the login details are matched to the server then the window get destroy by itself and opens a new window**

**\* If the login details login details doesn't matched to the server**

**then a dialog box will appear which shows warning(INCORRECT PASSWORD)'''**

**user = txtUser.get()**

**pwd=txtPwd.get()**

**if user=="GROUP1" and pwd=="12345":**

**window.deiconify()**

**windowlg.destroy()**

**else:**

**tkmb.showwarning("Login","Incorrect Password")**

**#It shows Warning**

**def btnCancelClicked():**

**'''**

**\* It makes sure that you really wants to cancel the window**

**\* It make a dialog box to check your confirmation**

**'''**

**msg = tkmb.askyesno("Login Page","Are you sure you want to cancel?")**

**if msg:**

**txtUser.delete(0,END)**

**txtPwd.delete(0,END)**

**btnLogin = Button(windowlg, text="Login", command=btnLoginClicked)**

**#It creates a button(login) which is used to login to the new window**

**btnLogin.grid(column=0,row=3)**

**#It give position to the Login button.**

**btnCancel = Button(windowlg, text="Cancel", command=btnCancelClicked)**

**#It creates a button(cancel) which is used to Cancel the current window**

**btnCancel.grid(column=1,row=3)**

**#It give position to the Cancel button.**

**def checkInsert():**

**'''**

**It is used to check wheather all the entries are not null.**

**'''**

**if len(txtRoll.get())==0 or len(txtName.get())==0 or len(txtMarks.get())==0:**

**return False**

**else:**

**return True**

**def checkSearch():**

**'''**

**It is used to check wheather the search box is not null**

**'''**

**if len(txtRollSearch.get())==0:**

**return False**

**else:**

**return True**

**def clearlistBox():**

**listBox.delete(0, END)**

**def clearEntry():**

**'''**

**It is used to wipe out the written data after the inseration of data**

**'''**

**txtRoll.delete(0,END)**

**txtName.delete(0,END)**

**txtMarks.delete(0,END)**

**def btnDeleteAllClicked():**

**'''**

**\* It is used to delete the data which we want to delete from the database**

**\* After the complimation of delete it shows that**

**(All records have been deleted)**

**'''**

**cmd="delete from student"**

**cursor.execute(cmd)**

**con.commit()**

**tkmb.showinfo("Record Delete","All records have been deleted")**

**def btnInsertRecordClicked():**

**'''**

**\* It is used to insert the entered data in the database.**

**\* First off all it use the checkInsert() Function to make sure the all fields are not null**

**-> If all the fields are filled then it insert the given data into the database and a dialog box appears which shows(Record Saved)**

**-> If any field is not field then it gives ERROR!! in a dialog box that all fields are required**

**'''**

**try:**

**if checkInsert()==True:**

**roll=int(txtRoll.get())**

**name=txtName.get()**

**marks=int(txtMarks.get())**

**cmd="insert into student values({},'{}',{})".format(roll,name,marks)**

**cursor.execute(cmd)**

**con.commit()**

**cmd="select \* from student"**

**cursor.execute(cmd)**

**data=cursor.fetchall()**

**count=cursor.rowcount**

**tkmb.showinfo("Data Entry","Record "+ str(count)+" saved!")**

**clearEntry()**

**else:**

**tkmb.showwarning("Error!!!","All fields required.")**

**except Exception as e:**

**tkmb.showerror("Error!!!",str(e.args[1])+str(e.args[0]))**

**def btnSearchRecordClicked():**

**'''**

**\* It is used to access the details of the data which we want**

**\* First of all it check that the search box is not null**

**->If the search box is null then it shows Warning ( Please Enter a Roll number to search )**

**->After that it check that whether the roll no. exist or not in the database**

**->>If the Roll no. Exist in the Database then It shows all the corresponding details of the database.**

**->>If the roll no. Doesn't Exist then it shows Warning (No such record found!!)**

**'''**

**clearEntry()**

**if checkSearch()==True:**

**global oldroll**

**oldroll=int(txtRollSearch.get())**

**cmd="Select \* from student where roll={}".format(oldroll)**

**cursor.execute(cmd)**

**data=cursor.fetchall()**

**if len(data):**

**txtRoll.insert(INSERT,data[0][0])**

**txtName.insert(INSERT,data[0][1])**

**txtMarks.insert(INSERT,data[0][2])**

**else:**

**tkmb.showwarning("Error!!!","No such record found!!!")**

**else:**

**tkmb.showwarning("Error!!!","Please Enter a Roll Number to Search!")**

**def btnUpdateRecordClicked():**

**'''**

**\* It is used to update the entered data in the database.**

**\* First off all it use the checkInsert() Function to make sure the all fields are not null**

**-> If all the fields are filled then it update the given data into the database and a dialog box appears which shows(Data Updated successfully)**

**-> If any field is not field then it gives ERROR!! in a dialog box that all fields are required**

**'''**

**try:**

**if checkInsert()==True:**

**roll=int(txtRoll.get())**

**name=txtName.get()**

**marks=int(txtMarks.get())**

**oldroll=int(txtRoll.get())**

**cmd="update student set roll={},name='{}',marks={} where roll={}".format(roll,name,marks,oldroll)**

**cursor.execute(cmd)**

**con.commit()**

**msg=tkmb.showinfo("Update Data","Data updated successfully")**

**clearEntry()**

**clearlistBox()**

**cmd="select \* from student"**

**cursor.execute(cmd)**

**data=cursor.fetchall()**

**for i in range(len(data)):**

**listBox.insert(i,data[i])**

**else:**

**tkmb.showwarning("Error!!!","All fields required.")**

**except Exception as e:**

**tkmb.showerror("Error!!!",str(e.args[1]))**

**def btnDeleteRecordClicked():**

**'''**

**\* It is used to Delete the entered data in the database.**

**\* First off all it use the checkInsert() Function to make sure the all fields are not null**

**-> If all the fields are filled then it Delete the provided data into the database and a dialog box appears which shows(Record Deleted)**

**-> If any field is not field then it gives ERROR!! in a dialog box that (Please Enter a Roll Number to Delete!)**

**'''**

**if checkSearch()==True:**

**roll=int(txtRollSearch.get())**

**cmd="delete from student where roll={}".format(roll)**

**cursor.execute(cmd)**

**con.commit()**

**tkmb.showinfo("Delete Record","Record Deleted")**

**clearEntry()**

**else:**

**tkmb.showwarning("Error!!!","Please Enter a Roll Number to Delete!")**

**def btnExitClicked():**

**'''**

**\* It is used to exit from the window screen**

**\* It make sure that you really want to exit the window for which it shows a dialog box for confirmation.**

**'''**

**msg = tkmb.askyesno("Data Entry","Are you sure you want to Exit?")**

**if msg:**

**window.destroy()**

**def btnDisplayClicked():**

**'''**

**\* It is used to show the entire data from database on the window**

**'''**

**clearlistBox()**

**cmd="select \* from student"**

**cursor.execute(cmd)**

**data=cursor.fetchall()**

**for i in range(len(data)):**

**listBox.insert(i,data[i])**

**def CurSelected(evt):**

**value=listBox.get(ACTIVE)**

**clearEntry()**

**txtRoll.insert(INSERT,value[0])**

**txtName.insert(INSERT,value[1])**

**txtMarks.insert(INSERT,value[2])**

**lblRollSearch=Label(window,text="Enter Roll")**

**#It is used for labelling**

**lblRollSearch.grid(column=0,row=1)**

**#It is used to give position.**

**txtRollSearch=Entry(window, width=6)**

**#It is used for Entering the value**

**txtRollSearch.grid(column=1,row=1)**

**#It is used to give position.**

**btnSearchRecord=Button(window,text="Search",command=btnSearchRecordClicked)**

**#It is used to make a button.**

**btnSearchRecord.grid(column=2,row=1)**

**#It is give position.**

**lblRoll = Label(window, text="Roll")**

**#It is used for labelling**

**lblRoll.grid(column=0, row=3)**

**#It is give position.**

**lblName = Label(window, text="Name")**

**#It is used for labelling**

**lblName.grid(column=0, row=4)**

**#It is give position.**

**lblMarks = Label(window, text="Marks")**

**#It is used for labelling**

**lblMarks.grid(column=0, row=5)**

**#It is give position.**

**txtRoll =Entry(window,width=10)**

**#It is used for Entering the value**

**txtRoll.grid(column=1,row=3)**

**#It is give position.**

**txtName =Entry(window,width=10)**

**#It is used for Entering the value**

**txtName.grid(column=1,row=4)**

**#It is give position.**

**txtMarks =Entry(window,width=10)**

**#It is used for Entering the value**

**txtMarks.grid(column=1,row=5)**

**#It is give position.**

**btnUpdateRecord=Button(window,text="Update",command=btnUpdateRecordClicked)**

**#It is used to make a button.**

**btnUpdateRecord.grid(column=0,row=7)**

**#It is give position.**

**btnInsertRecord=Button(window,text="Insert New",command=btnInsertRecordClicked)**

**#It is used to make a button.**

**btnInsertRecord.grid(column=1,row=7)**

**#It is give position.**

**btnDeleteRecord=Button(window,text="Delete Record",command=btnDeleteRecordClicked).place(x=20,y=120)**

**#It is used to make a button.**

**btnExit=Button(window,text="EXIT",height=3,width=10,command=btnExitClicked).place(x=20,y=160)**

**#It is used to make a button.**

**lblAllRecord=Label(window,text="All Records").place(x=215,y=0)**

**#It is used for labelling.**

**btnDisplay=Button(window,text="Show All Records",command=btnDisplayClicked).place(x=200,y=190)**

**#It is used to make a button.**

**btnDeleteAll=Button(window,text="Delete All Records",command=btnDeleteAllClicked).place(x=200,y=220)**

**#It is used to make a button.**

**scrollbar = Scrollbar(window)**

**#It is used to drag the data in the List box if the data exceds the window size.**

**scrollbar.place(x=290,y=20)**

**# It is used to give Position to the scroll bar**

**listBox = Listbox(window,height=10,width=15)**

**#It is used to give size to the list box**

**listBox.place(x=200,y=20)**

**#It is used to give position to the list box**

**listBox.bind('<<ListboxSelect>>',CurSelected)**

**#It is used to bind listbox to curselect**

**listBox.config(yscrollcommand=scrollbar.set)**

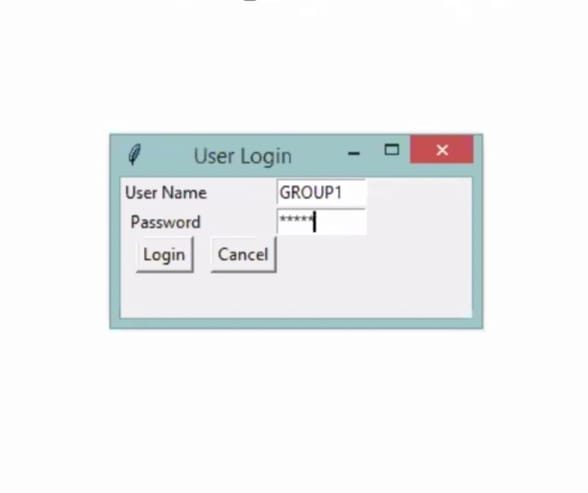
**scrollbar.config(command=listBox.yview)**

**# attach listbox to scrollbar**

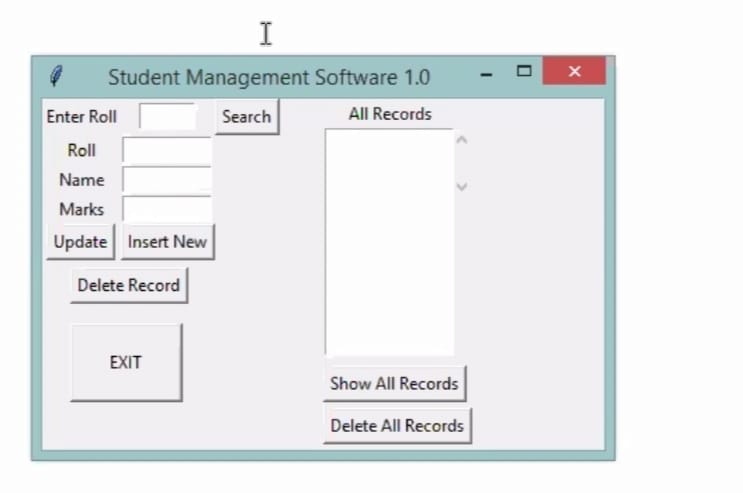
**window.mainloop()**

**OUTPUT SCREEN**

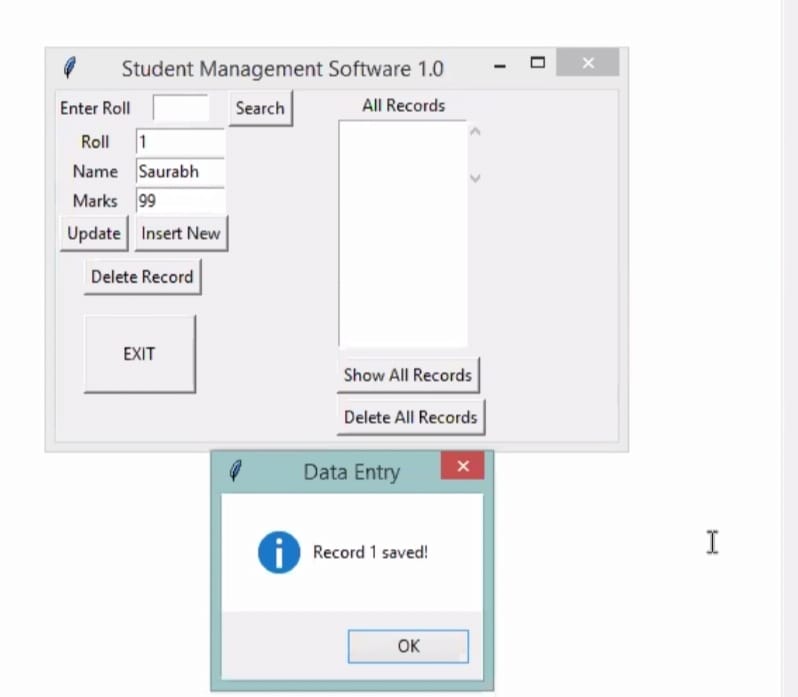
* User Login Window Screen.
* User Name = GROUP1 , Password = 12345
* In the below Window User can login to the Student Management System by clicking Login Button

****

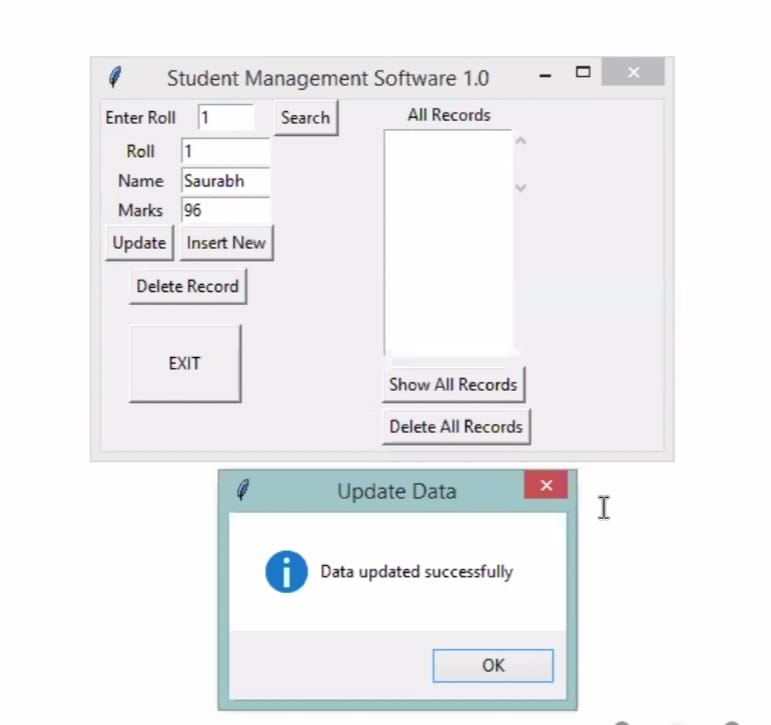
* This window will appear after the Successfull Login.



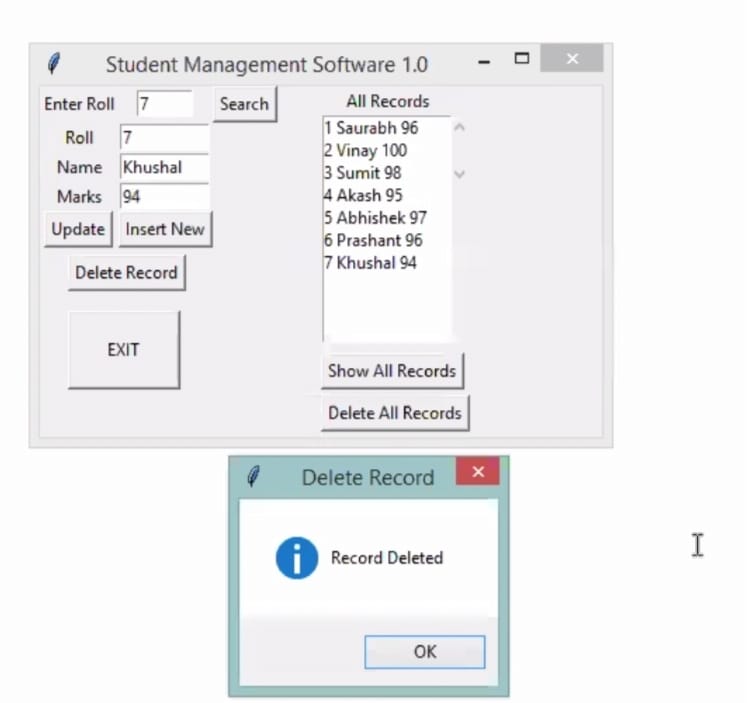
* In the below window Data is inserted by clicking Insert New Button.



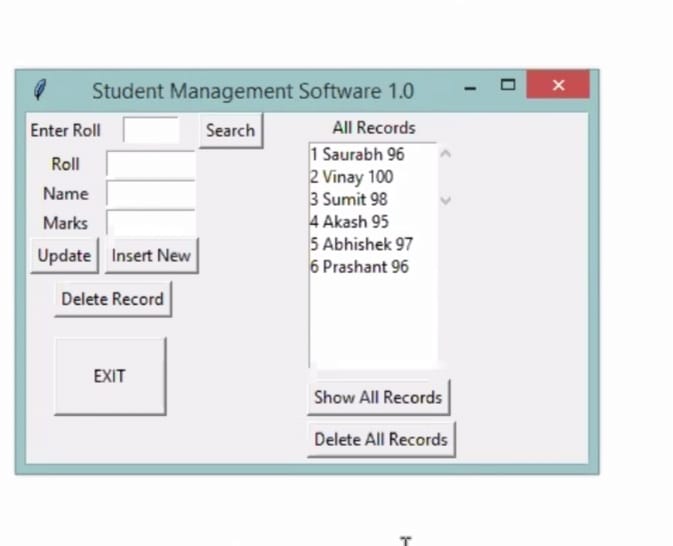
* In the below Window the data(marks) is updated by clicking Update Button.



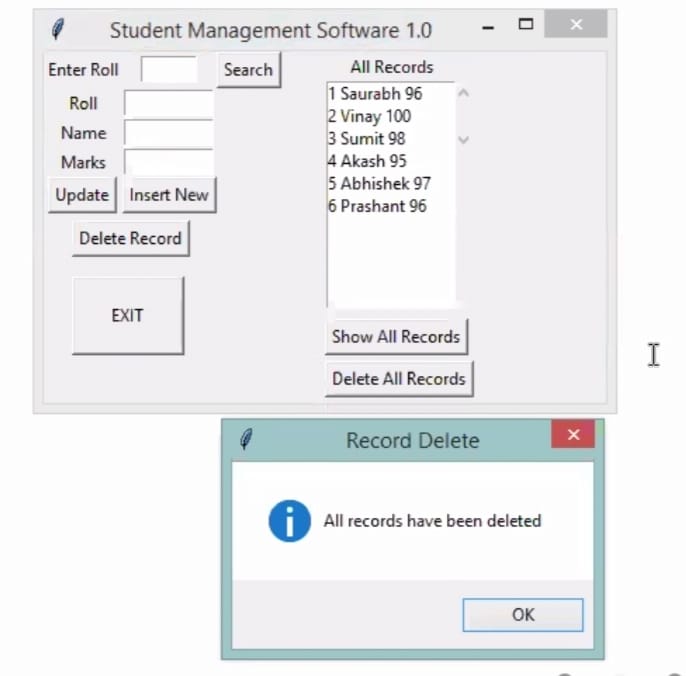
* In the below Window the selected data get deleted by clicking Delete record button.



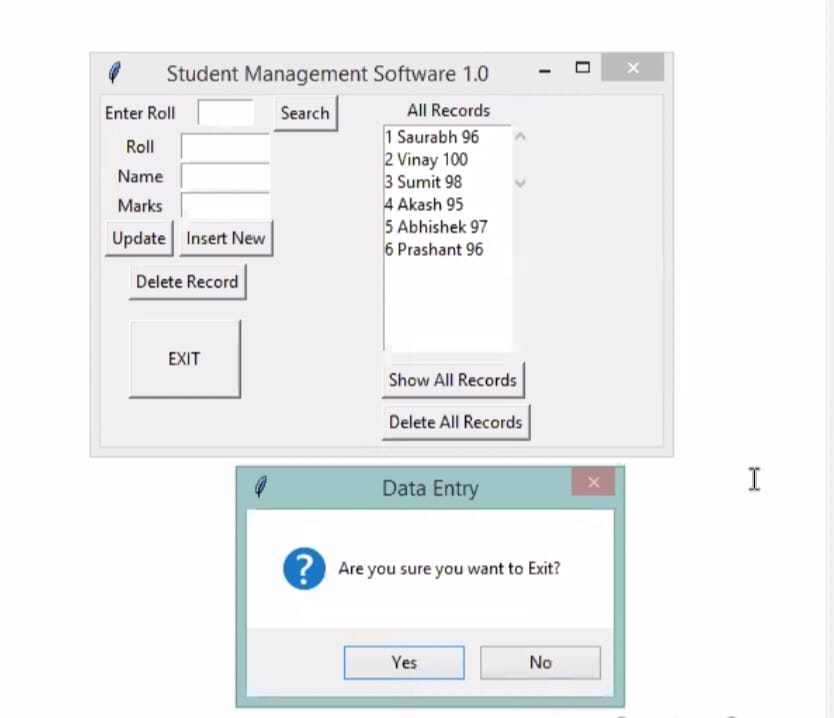
* In the below Window All records are shown by clicking the Show All Records button.



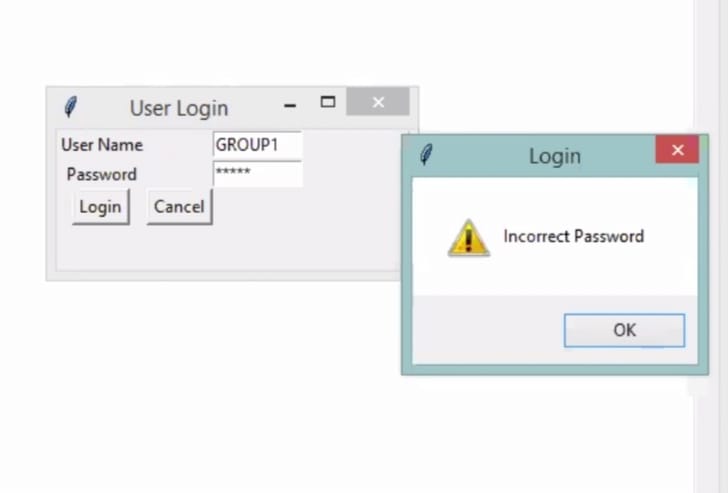
* In the below Window All records get deleted by clicking Delete All Records.



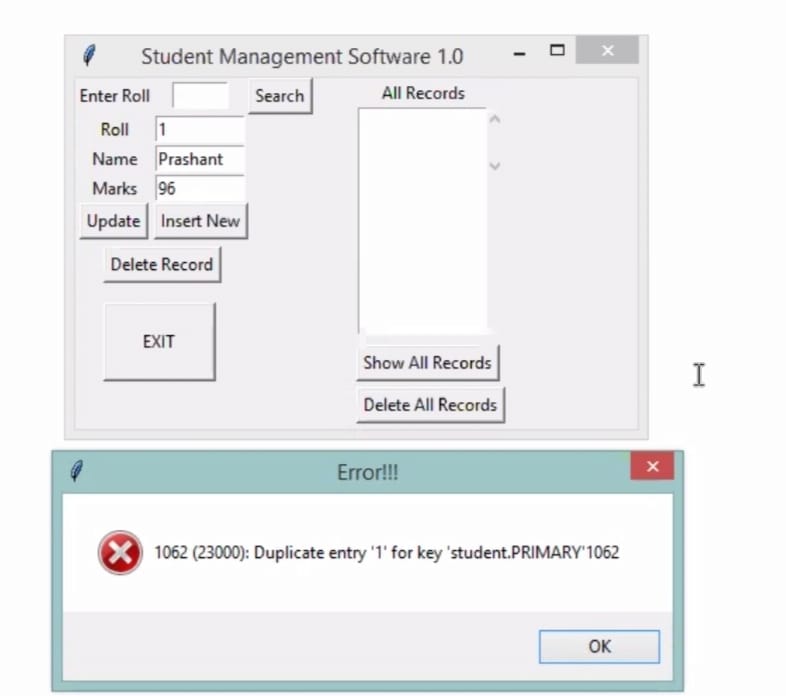
* In the below Window the user get exit by clicking EXIT button.



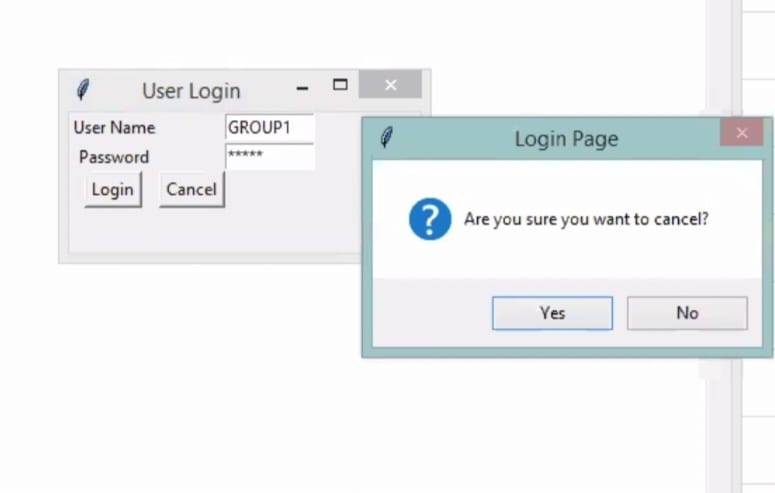
* If you entered wrong password then it shows following window Error.



* If you Enter any Roll no. in new record which already exist in the database then it shows following window Error



* If you want to cancel the user login window then the following window will appears.



**BIBLIOGRAPHY**

# Computer Science with Python (Sumitra Arora) – XI & XII

# [www.stackoverflow.com](http://www.stackoverflow.com)

* [**www.mysql.com**](http://www.mysql.com)
* [**www.python.org**](http://www.python.org)
* [**www.greeksforgeeks.com**](http://www.greeksforgeeks.com)
* [**www.codeacademy.com**](http://www.codeacademy.com)