**Voting Management System**

**By Using Python and Tkinter Library**

A PROJECT REPORT

***Submitted By***

HRISHIKESH PANDEY

# *Under the Project Guidance of*

Lakhan Mahato

***At***

***Ardent Computech Pvt. Ltd.***

***At*** 

# BONAFIDE CERTIFICATE

This is to certify that have HRSHIKESH PANDEY successfully completed the project titled “ Voting Management System By Using Python and Tkinter Library ” under my supervision during the period from August 23, 2023 to Oct 23, 2023 which is in fulfillmentof their training in Python Programming and Applications.

**Signature of Mentor**

**Lakhan Mahato**

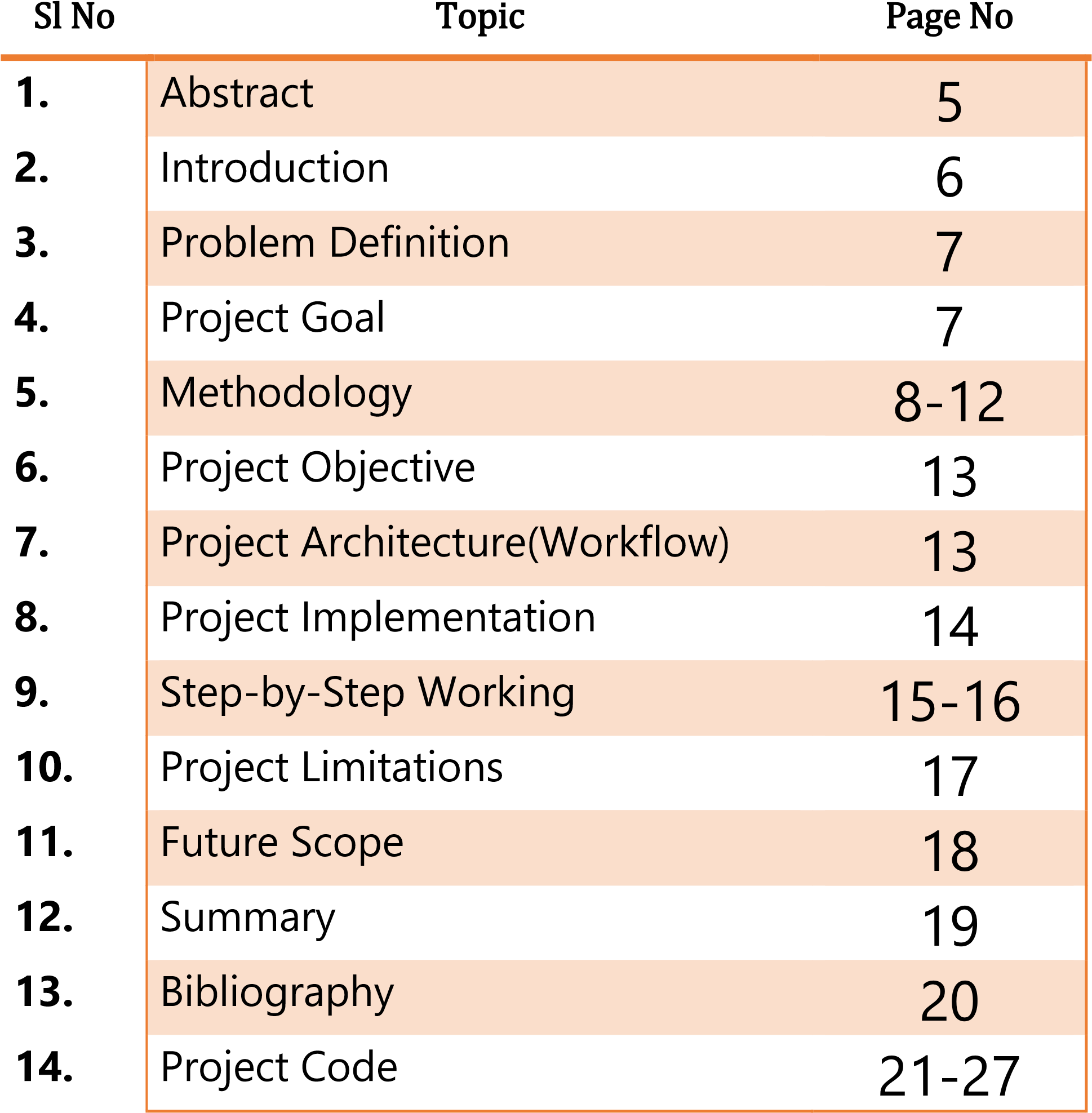
**Ardent ComputechPvt. Ltd. Salt Lake, Kolkata**

# ACKNOWLEDGEMENT

The project “ Voting Management System By Using Python and Tkinter Library ” would not have been possible without the constant guidance of our guide Lakhan Mahato , Ardent ComputechPvt. Ltd., who guided us throughout this process. We are immensely thankful to her for her valuable ideas on improvement of the project.

**PAGE - 1**

**PAGE - 1**



21 - 114

**INDEX**

**ABSTRACT**

**PAGE - 2**

**PAGE - 2**

The Project is developed for the threat free and user oriented Voting Management System. The Voting Management Voting system is made for the people of the country residing around the world and wants to vote for their representative...The automated ballot elections are called the electronic voting. An online voting system for Indian election is proposed for the first time in this paper. The proposed model has a secure authentication for greater security in the sense that voter high security password is confirmed before the vote is accepted in the main database of Election Commission of India. The additional feature of the model is that the voter can confirm if his/her vote has gone to correct candidate/party. In this model a person can also vote from outside of his/her allotted constituency or from his/her preferred location. In the proposed system the tallying of the votes will be done automatically, thus saving a huge time and enabling Election Commissioner of India to announce the result within a very short period. An online voting system for Indian election is proposed for the first time in this paper. The proposed model has a greater security in the sense that voter high security password is confirmed before the vote is accepted in the main database of Election Commission of India. The additional feature of the model is that the voter can confirm if his/her vote has gone to correct candidate/party. In this model a person can also vote from outside of his/her allotted constituency or from his/her preferred location. In the proposed system the tallying of the votes will be done automatically, thus saving a huge time and enabling Election Commissioner of India to announce the result within a very short period. Keywords: Authentication, Voting, Unique key

# INTRODUCTION

Online Election System would have Candidate registration, document verification, auto-generated User ID and pass for candidate and Voters. Admin Login which will be handled by Election Commission .Candidate Login which will be handled By Candidate, Voters will get Unique ID and Password, Using which they can vote for a Candidate only once per Election. The project is beneficial for Election Commission, Voters as the can get to know the candidate background and choose wisely, and even for Candidate. The software system allows the Candidate to login in to their profiles and upload all their details including their previous milestone onto the system. The admin can check each Candidate details and verify the documents, only after verifying Candidate’s ID and Password will be generated, and can remove faulty accounts. The software system allows Voters to view a list of Candidates in their area. The admin has overall rights over the system and can moderate and delete any details not pertaining to Election Rules.

# PROBLEM DEFINITION

# The percentage of polling on the day of elections is not satisfactory as majority of the people are not coming to vote and thinks is just as wastage of time. The manual voting system takes long time as there is a lot of paper work first and then human effort is also there for counting of the votes. Manual voting consumes almost 4-6 hrs. (approx.) of every voter which is surely a headache. The voting will be done online such that there is no need to come at the place on the time of elections and the people can vote from the home or from any other place. A key will be provided to every person, so that on the time of elections they can easily login on the election link and can use his/her vote.

# PROJECT GOAL

# i.Increasing number of voters as individuals will find it easier and more convenient to vote.

# ii. Less effort and less labor intensive, as the primary cost and focus primary on creating, managing, and running a secure web voting portal.

# iii. The system can be used anytime and from anywhere by the Voters.

# iv. No one can cast votes on behalf of others and multiple times.

# v. Saves time and reduces human intervention.

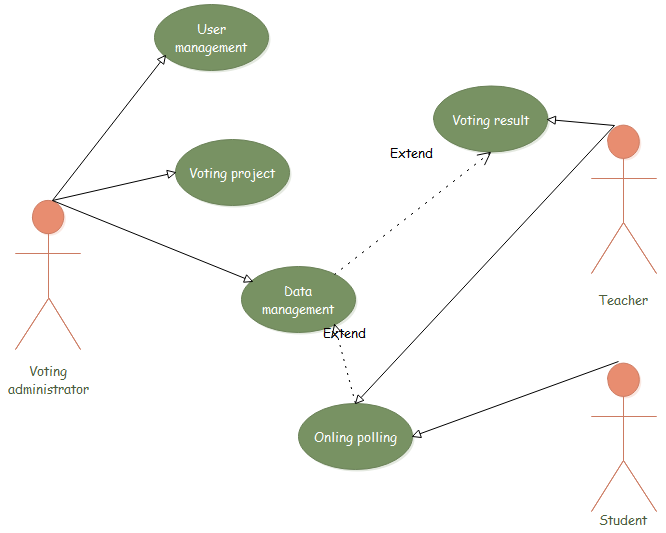
# vi. The system is flexible and secured to be used.

# vii. Unique Identification of voter through Aadhar number.

# viii. Improves voting with friendly Interface.

# ix.No fraud voter can be submitted. METHODOLOGY

This project methodology is needed to make sure the project that consists of software development will be developed systematically in order to acquire a better result. The overall of the project methodology is shown as figure below.

****

**Methodology of Voting Management system**

The methodology is divided to four main components which are literature review, database design, creating a Graphical user interface (GUI) and finally is the software testing.

* **Literature review :-**

The project begins with literature review where the overview of the project needs to be known. The literature review is done by finding out numbers of lectures student and class provided, involved for Faculty of Electronics and Computer Engineering (FKEKK), how many class and how many lecturers contribute to this faculty. All this information need to be known in order to provide information inside the software that is useful in arranging the database.

* **Database design :-**

Database is required to improve the functionality of this project. Therefore, the first stage of software development is to create the database. Database is a body of information made up of related pieces of data organized so that they can be easily been manipulated by the computer. As for the software development, it is important to have database as all the information needed can be saved for future used.

* **Creation of suitable Graphic user interface (GUI) :-**

GUI is a way to interact with a computer using pictures and other visual elements displayed on a computer screen. The pictures and buttons used to control many Internet sites are an example of a GUI. GUI is important to make the user easy to understand what they need to do in order to use the software GUI must be user friendly and easy to be understood. In order to make the GUI function, Microsoft Visual Studio coding must be inserted to make the software work properly.

* **Software testing :-**

Software testing is the process used to measure the quality of developed computer software to determine that the software meets its required results. This testing process of executing a program is intended of finding errors.

* **Requirement plan phase :-**

Before developing the system, either small or large system the proper planning needs to be put in the first place. This is to make sure that all the activities that will execute and overall development are under control. The purpose of planning is to prepare the workflow for project management start from early stages of project development until end of development. The developer need to mention that it is very important to divide the project into subtasks to make sure about time estimation and to identify source requirement to complete this Student Information System project.

* **Requirement specification analysis phase :-**

All the requirement analysis phase will be started after the study case for this system has been completed. Through this phase, developer will determine clearly who will use the system, function of the system, when and where the system will be executed. Developer will begin the study from existing system to define the strength and weakness. This is important as it can identify the opportunities to improve the existing system to a new system. If there are any shortcomings or defects, it can be corrected in the new system. After this research, the concept of new system is developed. Besides, the information is collected by using fact finding techniques such as background reading, interviewing and observation. As a developer, it needs to analyse the feasibility of new system, which is to improve the system as to make sure that the new system will fulfil the user requirement. All the constraints and limitation of the new system also need to analyse.

* **Design phase :-**

Design phase started after the specification and analysis phase completed. In this step, specified technical designs were created to the smallest detail. It is common for several alternative solutions to be identified, but only one must ultimately be chosen as the best of the system and users. Trade-off may become part of the reality for the project, which may include time, scopes and functionality. The purpose of the system designing is to create a blueprint for the new system that will satisfy all documented requirements. There are several things to identify during system designing such as all necessary outputs, inputs, interfaces and processes. Furthermore, for this phase, optimal solution of the designing was developed. The main objects for the system and relationship between object are well defined to make sure that there will be an interaction on it, so that all the objects need to classify on their own classes. Other than that, the attribute of class also was defined.

Also in this phase, there are few designing, which applied into new system such as database design, interface design and coding design.

* **Database design :-**

To design database, it considers and concerns in many aspects. The specification will be determined by what is the data need to be stored in database. In database design, each step of the process needs to be done very carefully for avoiding any unexpected errors. The errors might occur in database, it can give complex problem for the system. These databases are designed by using Microsoft Access 2003. For Student Information System database, there are seven tables involved such as student, lecturer, activity, position, and class, list of student and list of lecturer.

* **Interface design :-**

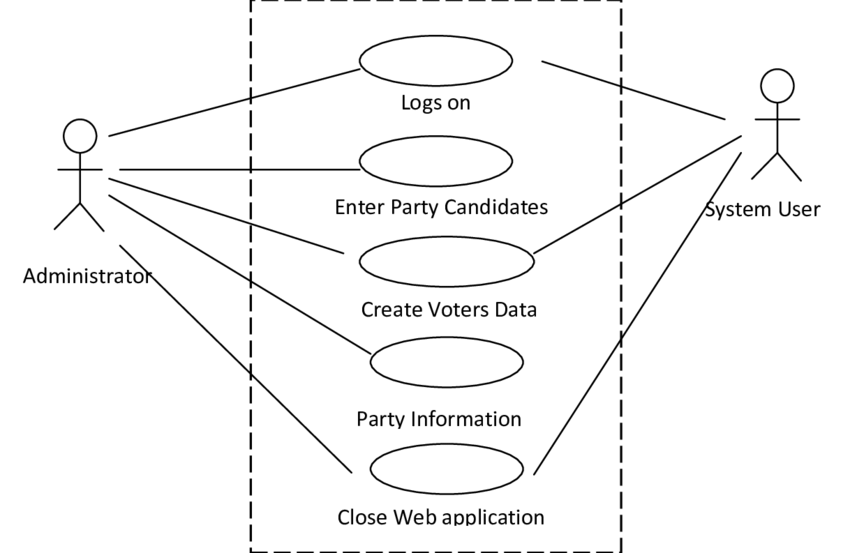
The purpose of interface design is to determine how the layout of the system and to make sure that this layout suitable with user requirement. The good designing can attract the users and supposedly not confused the users with each functions of the system. For instance, the developer also needs to concern about user-friendly interface during designing phase. Through this system, there are some elements, which been used to design the interface.

* **Coding design :-**

This coding designing showed how programming language would be implemented. It also will explain the purpose for each coding development. SQL statement is used in order to make sure that interfaces can be connected with database.

* **RAD (Rapid Application Design) Methodology :-**

This Student Information System development used RAD (Rapid Application Design) methodology. This methodology method follows the System Development Life Cycle (SDLC) that in a sequential and structured away. RAD is a methodology for compressing the analysis, design, build, and test phases into a series of short, iterative development cycles. This has a number of distinct advantages over the traditional sequential development model. Iteration allows for effectiveness and self-correction. Studies have shown that human beings almost never perform a complex task correctly the first time. However, people are extremely good at making an adequate beginning and then making many small refinements and improvements. RAD projects are typically operated with small-integrated teams comprised of developers, end users, and IT technical resources. Small teams, combined with short, iterative development cycles optimize speed, unity of vision and purpose, effective informal communication and simple project management.

****

***Voting Management System use case diagram for admin roles***

# PROJECT OBJECTIVE

The main objective of our project is to successfully complete the Voting Management System and whether Users and Admin are being helped through this should be well noticed.

# PROJECT ARCHITECTURE(WORKFLOW)

This is the detailed work architecture where we are showing the process of Voting Management System.

**Voting Management System Page**

**Voting Management System Dashboard**

Voter Login Form

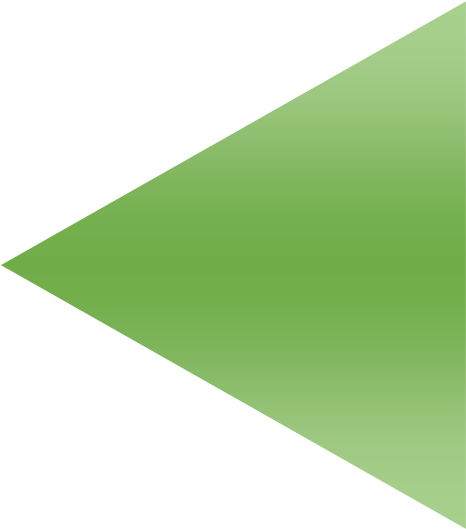
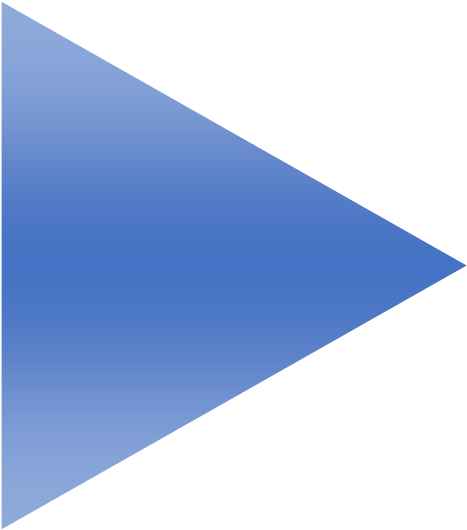
Admin Login Button

Using Voter ID Give Vote

Give Vote Button

New Voter Registration Button

**PROJECT IMPLEMENTATION**



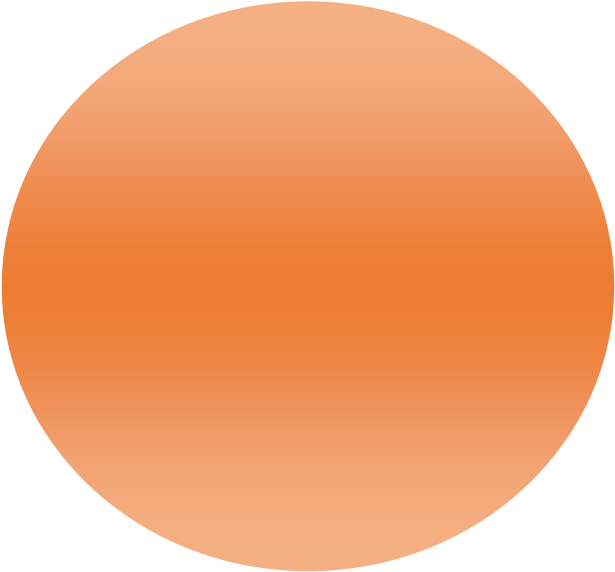
**DATA**

**RESULT**

**INPUT**

**PROCESSING**

**OUTPUT**



**PROCESSING**

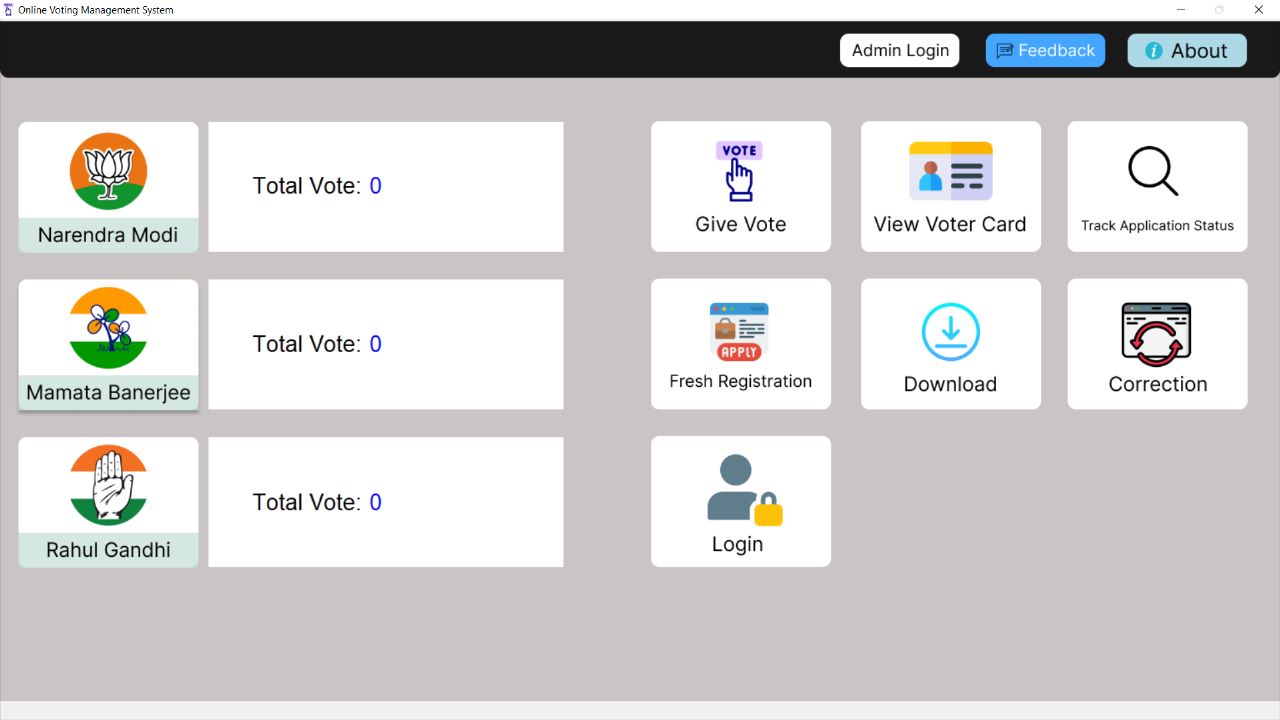
**SYSTEM**

## *Processing system:*

*Processing* systems contain information about important people, places and things within the organization or in the environment surrounding it. Information is a data that have been designed into a form that is meaningful and useful to human beings. Data, in contrast, are streams of raw facts re-presenting events occurring in organizations or the physical environment before they have been organized into a form that people can understand and use.

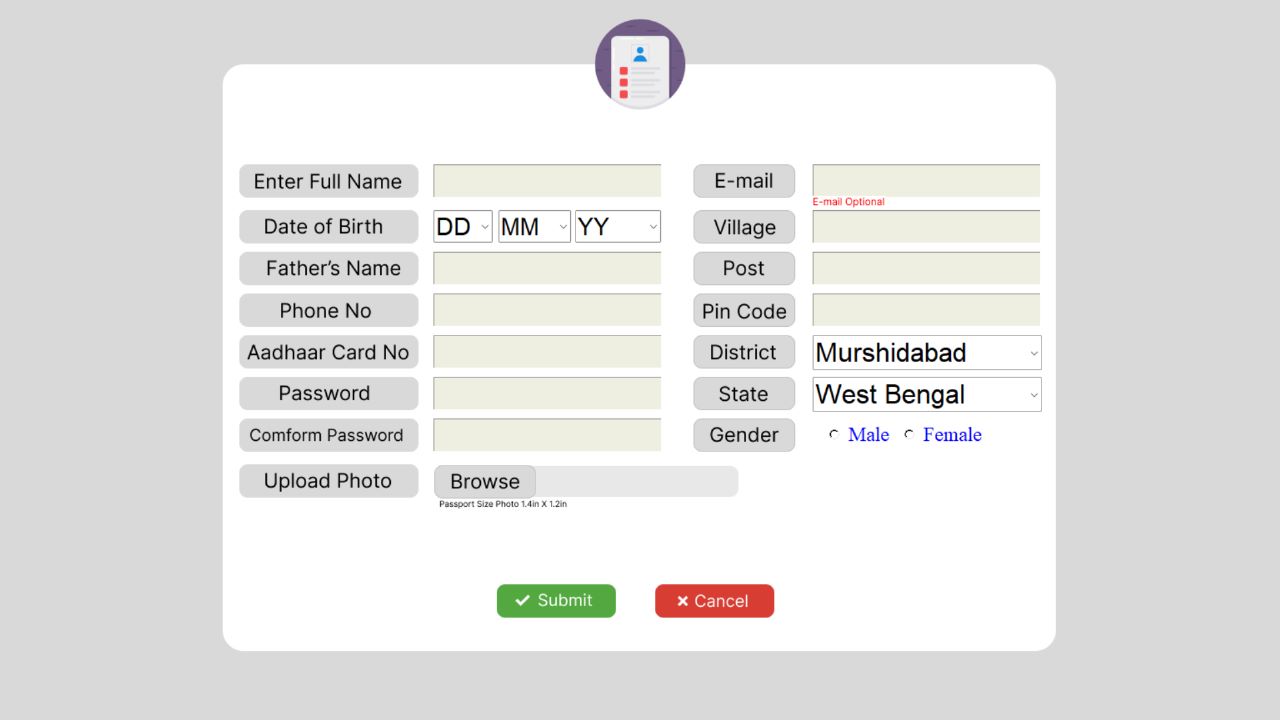
# STEP-BY-STEP WORKING

* **Voting Management System Dashboard :-**



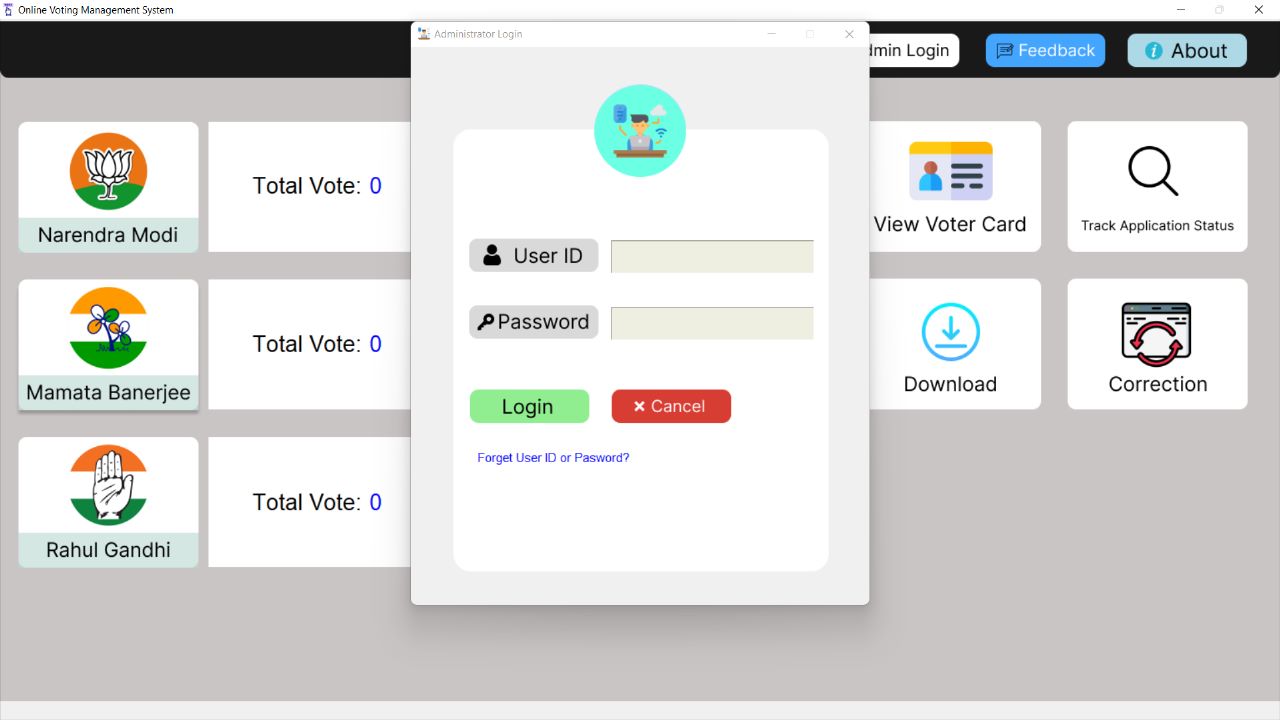
After running the Python program the voting management system dashboard will open and here seven buttons have been created give Vote , View Voter card , Track Application , Fresh Registration, Download ,Correction and Login .

* **Fresh Registration form :-**

****

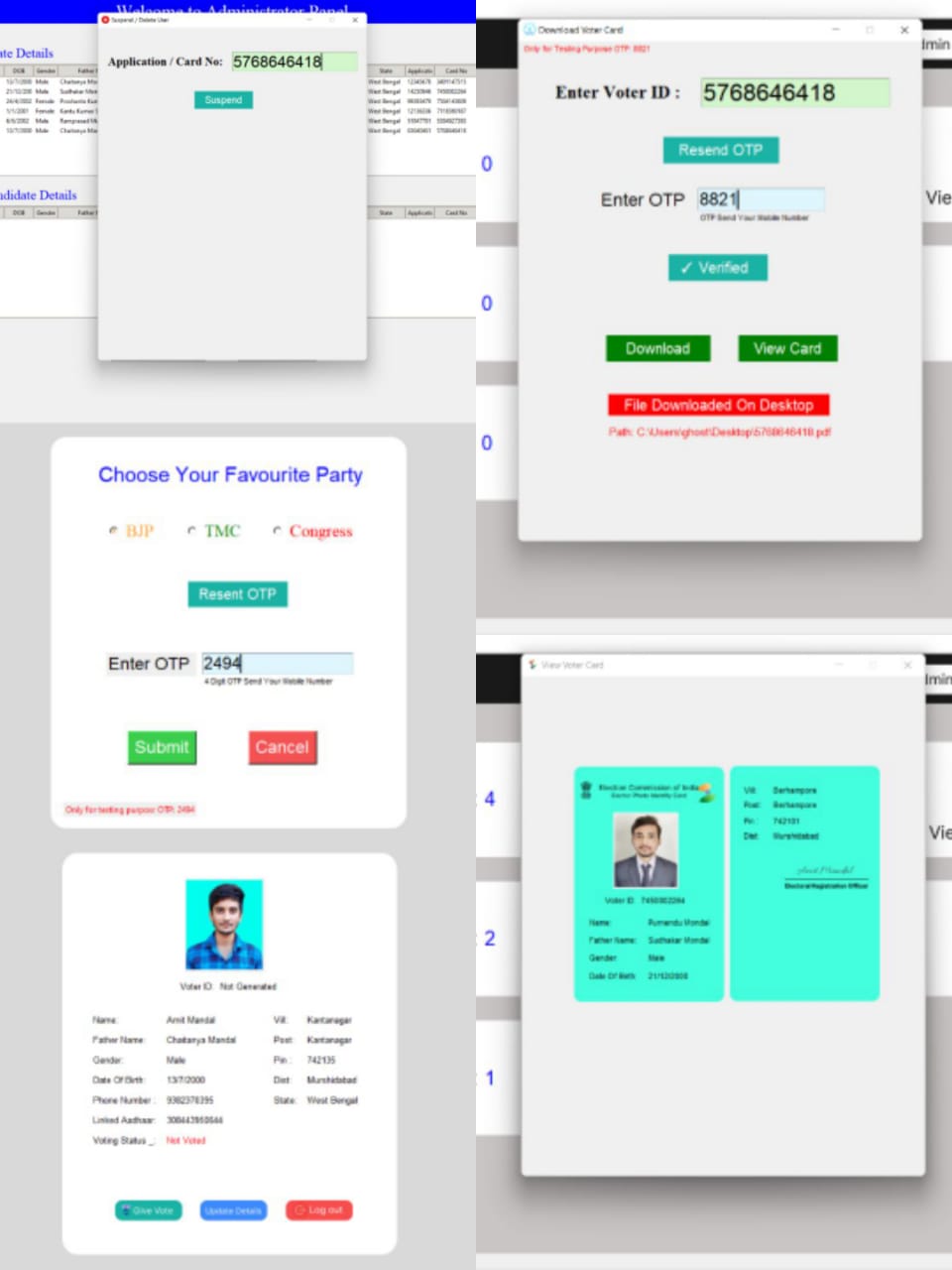
After clicking Fresh Register the registration page will open and there Name, D.O.B, Father Name ,Ph no, Adair no , Password ,conform password , Email , village ,post , pin code ,district ,state, gender after fill up Submit, & Cancel & buttons there.

* **Admin Login Form :**

****

After clicking the Admin login button the Login Form will open and here two buttons have been created User ID and Password.

* **Finding Voter Card , View Voter Card , Give Vote :**

****

# PROJECT LIMITATIONS

* Software is limited to Desktop only.
* System requires python interpreter installed on the system.
* All opinions of student management are not included in current version. Security options provide only low level security against beginner attackers.
* GUI is in English only.

# FUTURE SCOPE

* This software can be made for all OS.
* Higher Security features can be included in this software.
* Program scheduling can also be included in this software.
* This software can be developed to use as tutorial to teach basic concepts of OS to new users.
* This software can be implemented with OS to reduce overhead of installing and running interface of each and every tool at different place.
* Automatic Shutdown through SMS can be implemented in this.

# SUMMARY

This documentation outlines the procedure of V Management System By Using Python and Tkinter Library provided by Ardent

Compute Pvt. Ltd.

This mini project is very good python program which enhances basic skill of learner. People and leaders both can use the application for voting as well for participating in the contest. At first, the application asked any user that they are Voters or a leader of any group. After that it takes respective operation. If the user is a leader, then the application asked the user to store their name, so that voters can nominate their leaders. But if the user is voter, then the application asked for Voter Id. The error handling is also perfect, in every wrong step, the application will show respective message, which makes the application friendlier. It also shows winner name and the number of votes he/she

received.

**Conclusion:-**

**Please give conclusion**

**BIBLIOGRAPHY**

* **To make this Project, I have to take some sources from the :**

## https://technocrash.online/python-projects/online-voting-system-project-in-python

## https://github.com/andrew-geeks/tkinter-voting-system

## https://www.freeprojectz.com/python-django-project/voting-management-system

## https://www.kashipara.com/program/python/6394/voting-system-using-python-tkinter

## https://www.geeksforgeeks.org/voting-system-project-using-django-framework

## PROJECT CODE

From chi import test

From tab nanny import check

from tinder import\*

from msilib.schema import ComboBox

from operator import concat

from tkinter import ttk

from tkinter import messagebox

from venv import create

from sqlalchemy import null

from tkinter import filedialog

import sqlite3

import re

import random

from twilio.rest import Client

from fpdf import FPDF

import os

from PIL import Image, ImageTk

import io

from win10toast import ToastNotifier

import time

try:

db=sqlite3.connect('voting management system.db')

cr=db.cursor()

print('Database is connected with Dashboard')

db.commit()

except:

print('Database is connected with Dashboard')

root=Tk()

root.title('Online Voting Management System')

root.geometry('600x600')

root.minsize(600,400)

root.wm\_iconbitmap('icon/vote-sign.ico')

root.resizable(False,False)#maximize option disable

#root.attributes('-fullscreen',True)

root.state('zoomed')#default open fullscreen

#-------------------------------------- Main Body ---------------------------------------- #

'''------------------------------ Count Total Real-Time Vote-------------------------------'''

cr.execute("select vot\_status from registration")

vt\_data=cr.fetchall()

tmc=0

bjp=0

cong=0

for v\_c in vt\_data:

total\_v=v\_c[0]

if total\_v=='TMC':

tmc += 1

if total\_v== 'BJP':

bjp +=1

if total\_v== 'Congress':

cong +=1

# print('tmc=',tmc)

# print('bjp=',bjp)

# print('conj=',cong)

'''-----------------------------------------------------------------------------------------'''

#-----------------------------------<<< About >>>------------------------------------ #

def about():

messagebox.showinfo('Details','Develop by ' ' ----> 'Hrishikesh Pandey ' \nApplication Verson 1.0')

def feedback():

feedwin=Toplevel()

feedwin.title('Feedback or Report')

feedwin.geometry('550x680+480+25')

feedwin.wm\_iconbitmap('icon/feedback.ico')

feedwin.resizable(False,False)

feedwin.config(bg='#D9D9D9')

feedwin.config(bg='#C9C5C5')

#aboutwin.attributes('-fullscreen',True)

try:

db=sqlite3.connect('voting management system.db')

cr=db.cursor()

cr.execute('create table feedback(email text,feedback\_receive text)')

except:

print('feedback table is connected')

#send feedback to database............

feedemail=StringVar()

def senddata():

#define variable

feed\_email=feedemail.get()

feed\_data=feed\_e.get('1.0','end-1c')

regex = r'\b[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b'

# Define a function for

# for validating an Email

def check(feed\_email):

# pass the regular expression

# and the string into the fullmatch() method

if(re.fullmatch(regex, feed\_email)):

#print("Valid Email")

#insert data..........

cr.execute('insert into feedback(email,feedback\_receive) values(?,?)',(feed\_email,feed\_data))

db.commit()

messagebox.showinfo('Feedback','Thank You for Submit Feedback\n we will notify you after review feedback')

feedwin.destroy()

else:

#print("Invalid Email")

invalid=Label(feedwin,text='Please Enter Valid E-mail',font=('Regular',10),bg='white',fg='red')

invalid.place(x=175,y=210)

check(feed\_email)

#bg thems..............

feedback\_img=PhotoImage(file='themes/feedback\_b.png')

feed\_l=Label(feedwin,image=feedback\_img,bg='#C9C5C5',width=500,height=600)

feed\_l.place(x=25,y=10)

#emai...........

em\_img=PhotoImage(file='button/feedem.png')

feed\_l=Label(feedwin,image=em\_img,bg='white')

feed\_l.place(x=60,y=166)

#entry.........

feed\_e=Entry(feedwin,font=('Regular',24),width=18,bd=0,bg='#D4F6CC',textvar=feedemail)

feed\_e.place(x=175,y=169)

#report........

report\_img=PhotoImage(file='button/feedreport.png')

report\_l=Label(feedwin,image=report\_img,bg='white')

report\_l.place(x=60,y=250)

#entry........

feed\_e=Text(feedwin,font=('Regular',16),height=8,width=36,bd=0,bg='#D4F6CC')

feed\_e.place(x=62,y=280)

#send button img import

send\_bg=PhotoImage(file='button/fedsend.png')

send=Button(feedwin,image=send\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=senddata)

send.place(x=125,y=530)

#cancel button img import

can\_bg=PhotoImage(file='button/cancel.png')

login=Button(feedwin,image=can\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:feedwin.destroy())

login.place(x=290,y=530)

feedwin.mainloop()

#--------------------------------<<< Registration Form >>>-------------------------------- #

def RegistrationForm():

root.destroy()

import registration\_form

#-----------------------------------<<< Login Form >>>------------------------------------ #

def LoginForm():

logwin=Toplevel()

logwin.title('Login Form')

logwin.geometry('550x680')#+510+50')

logwin.wm\_iconbitmap('icon/login.ico')

#logwin.resizable(False,False)

logwin.config(bg='#D9D9D9')

logwin.attributes('-fullscreen',True)

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> Database Connect <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

null=0

uid=StringVar()

upass=StringVar()

def login():

userid=uid.get()

upassword=upass.get()

#userid.................

userid\_length=len(userid)

if null == userid\_length:

#validation.....

validation=Label(logwin,text='Please Enter User ID',font=('Regular',10),fg='red',bg='white')

validation.place(x=730,y=280)

else:

#password..............

password\_length=len(upassword)

if null == password\_length:

p='Please Enter Password'

validation=Label(logwin,text=p,font=('Regular',10),fg='red',bg='white')

validation.place(x=730,y=360)

else:

cr.execute("select phone,password from registration")

data=cr.fetchall()

check=0

for i in data:

a=i[0]

b=i[1]

if userid==a and upassword==b:

check=1

break

else:

check=0

if check == 1:

#successfully login

afterloginwin=Toplevel()

afterloginwin.title('My Profile')

afterloginwin.geometry('550x680+498+30')

afterloginwin.wm\_iconbitmap('icon/myprofile.ico')

afterloginwin.resizable(False,False)

afterloginwin.config(bg='#D9D9D9')

afterloginwin.overrideredirect(1)

cr.execute(f'select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,card\_no,voting\_status,photo from registration where phone={a}')

data=cr.fetchall()

for i in data:

n=i[0]

d=i[1]

g=i[2]

f=i[3]

p=i[4]

aah=i[5]

vill=i[6]

po=i[7]

pin=i[8]

dist=i[9]

state=i[10]

cd=i[11]

vs=i[12]

myphoto=i[13]

'''--------------------------- Give Vote -------------------------'''

def give():

votewin=Toplevel()

votewin.title('Give Vote')

votewin.geometry('550x680+480+25')

votewin.wm\_iconbitmap('icon/feedback.ico')

votewin.resizable(False,False)

votewin.config(bg='#D9D9D9')

#votewin.attributes('-fullscreen',True)

votewin.overrideredirect(1)

# db=sqlite3.connect('voting management system.db')

# cr=db.cursor()

# c='9382370394'

# cr.execute(f'select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,card\_no,voting\_status from registration where phone={c}')

# data=cr.fetchall()

# for i in data:

# n=i[0]

# d=i[1]

# g=i[2]

# f=i[3]

# p=i[4]

# aah=i[5]

# vill=i[6]

# po=i[7]

# pin=i[8]

# dist=i[9]

# state=i[10]

# cd=i[11]

# vs=i[12]

if cd=='Not Generated':

nota='Sorry Your not Eligible for Vote\n Beacuse your voter number is not gengrated'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(votewin,image=gv\_img,bg='#D9D9D9',width=500,height=600)

gv\_l.place(x=30,y=10)

not\_l=Label(votewin,text=nota,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=35,y=200)

#cancel button img import

canc\_bg=PhotoImage(file='button/close.png')

canc=Button(votewin,image=canc\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:votewin.destroy())

canc.place(x=225,y=350)

else:

null=0

v1=IntVar()

def give():

v2=v1.get()

if v2==null:

#choose party.......

cpp\_l=Label(votewin,text='Please Choose Party',font=('Regular',8),fg='red',bg='white')

cpp\_l.place(x=223,y=210)

v3='please choose party'

elif v2==1:

v3='BJP'

elif v2==2:

v3='TMC'

else:

v3='Congress'

if v3=='BJP' or v3=='TMC' or v3=='Congress':

rsotp=StringVar()

cppp\_l=Label(votewin,font=('Regular',10),bg='white',width=30)

cppp\_l.place(x=223,y=210)

#resend otp

#resend\_bg=PhotoImage(file='button/resend.png')

resend=Button(votewin,text='Resent OTP',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',command=give,padx=10)

resend.place(x=225,y=240)

'''------------------------------------------------------'''

#generate otp.........

motp=str(random.randint(1111,9999))

#print(motp)

#

'''-------------------TESTING OTP-------------------------'''

ottp='Only for testing purposr OTP: ' +motp

test\_l=Label(votewin,text=ottp,fg='red')

test\_l.place(x=50,y=550)

'''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+p

# msg='Online Voting System\nDon't Share OTP\nyour Give vote OTP is: '+motp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

cr.execute(f'update registration set otp={motp} where card\_no={cd}')

db.commit()

already='Already Voted'

def submitotp():

reseveotp=rsotp.get()

cr.execute(f'select voting\_status,otp from registration where card\_no={cd}')

otpdata=cr.fetchall()

for ot in otpdata:

vts=ot[0]

sotp=ot[1]

if sotp==reseveotp:

if vts=='Not Voted':

cr.execute('update registration set voting\_status=?,vot\_status=? where card\_no=?',(already,v3,cd))

db.commit()

#Thanks for vote...................

successwin=Toplevel()

successwin.title('success full vote')

successwin.geometry('550x680+480+25')

successwin.wm\_iconbitmap('icon/feedback.ico')

successwin.resizable(False,False)

successwin.config(bg='#D9D9D9')

#successwin.attributes('-fullscreen',True)

successwin.overrideredirect(1)

note='Thank You For Give Vote...'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(successwin,image=gv\_img,bg='#D9D9D9',width=500,height=600)

gv\_l.place(x=33,y=10)

warning\_img=PhotoImage(file='logo/check.png')

war\_l=Label(successwin,image=warning\_img,width=200,height=100,bg='white')

war\_l.place(x=180,y=70)

not\_l=Label(successwin,text=note,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=150,y=200)

def closesuccess():

votewin.destroy()

successwin.destroy()

#cancel button img import

tgv\_bg=PhotoImage(file='button/close.png')

tgv=Button(successwin,image=tgv\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=closesuccess)

tgv.place(x=230,y=370)

successwin.mainloop()

else:

#already voted...................

alwin=Toplevel()

alwin.title('Already Voted')

alwin.geometry('550x680+480+25')

alwin.wm\_iconbitmap('icon/feedback.ico')

alwin.resizable(False,False)

alwin.config(bg='#D9D9D9')

#alwin.attributes('-fullscreen',True)

alwin.overrideredirect(1)

note='One user can give vote at once time\nYou Have Already Voted...\nSorry you can\'t give vote again\n Because Our System Checks You...'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(alwin,image=gv\_img,bg='#D9D9D9',width=500,height=600)

gv\_l.place(x=30,y=10)

warning\_img=PhotoImage(file='logo/warning.png')

war\_l=Label(alwin,image=warning\_img,width=200,height=100,bg='white')

war\_l.place(x=180,y=70)

not\_l=Label(alwin,text=note,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=85,y=200)

#cancel button img import

canc\_bg=PhotoImage(file='button/close.png')

canc=Button(alwin,image=canc\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:alwin.destroy())

canc.place(x=225,y=370)

alwin.mainloop()

else:

#print('wrong otp')

w\_l=Label(votewin,text='Please Enter Valid OTP',bg='White',fg='red',width=50)

w\_l.place(x=130,y=372)

#otp..........

otp\_l=Label(votewin,text='Enter OTP ',font=('Regular',18))

otp\_l.place(x=110,y=340)

otp\_s=Label(votewin,text='4 Digit OTP Send Your Mobile Number',font=('Regular',8),bg='white')

otp\_s.place(x=245,y=370)

otp\_e=Entry(votewin,font=('Regular',18),bg='#DFF6FF',width=16,textvar=rsotp)

otp\_e.place(x=245,y=340)

#login button img import

#sub1\_img=PhotoImage(file='button/u\_login.png')

sub1=Button(votewin,text='Submit',font=('Regular',18),fg='white',bg='#3CCF4E',cursor='hand2',command=submitotp)

sub1.place(x=140,y=450)

#cancel button img import

#canc\_bg=PhotoImage(file='button/cancel.png')

canc=Button(votewin,text='Cancel',font=('Regular',18),fg='white',bg='#F55353',cursor='hand2',command=lambda:votewin.destroy())

canc.place(x=310,y=450)

else:

print()

#please select party

# stream = io.BytesIO(n)

# img=Image.open(stream)

# img =PhotoImage(img)

#bg thems..............

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(votewin,image=gv\_img,bg='#D9D9D9',width=500,height=600)

gv\_l.place(x=30,y=10)

#choose party.......

cp\_l=Label(votewin,text='Choose Your Favourite Party',font=('Regular',22),bg='white',fg='blue')

cp\_l.place(x=96,y=70)

cp1\_e=Radiobutton(votewin,text='BJP',font=('times new roman',18),fg='#FF9F29',bg='white',activebackground='white',value=1,cursor='hand2',variable=v1)

cp1\_e.place(x=110,y=150)

cp2\_e=Radiobutton(votewin,text='TMC',font=('times new roman',18),fg='green',bg='white',activebackground='white',value=2,cursor='hand2',variable=v1)

cp2\_e.place(x=220,y=150)

cp3\_e=Radiobutton(votewin,text='Congress',font=('times new roman',18),fg='red',bg='white',activebackground='white',value=3,cursor='hand2',variable=v1)

cp3\_e.place(x=340,y=150)

#give vote button img import

#gvote\_bg=PhotoImage(file='button/verify.png')

gvote=Button(votewin,text='✓ Verify ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',command=give,padx=10)

gvote.place(x=225,y=240)

votewin.mainloop()

'''---------------------- Profile Area----------------------'''

afterlogin\_img=PhotoImage(file='themes/afterlogin.png')

afterlogin\_l=Label(afterloginwin,image=afterlogin\_img,bg='#D9D9D9',width=500,height=600)

afterlogin\_l.place(x=25,y=10)

u\_img=PhotoImage(file='button/uphoto.png')

userphoto=Label(afterloginwin,image=u\_img)

userphoto.place(x=210,y=50)

#////////////////////////////////

fp = io.BytesIO(myphoto)

# load the image

image = Image.open(fp)

res=image.resize((114,134))

# drawing image to top window

userimg = ImageTk.PhotoImage(res)

userphoto=Label(afterloginwin,image=userimg)

userphoto.place(x=210,y=50)

#\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

#------------------------------------

#card no

ucard\_no=Label(afterloginwin,text='Voter ID: ',font=('Regular',10),bg='white')

ucard\_no.place(x=200,y=200)

ucard\_p=Label(afterloginwin,text=cd,font=('Regular',10),bg='white')

ucard\_p.place(x=260,y=200)

#>>>>>>>>>>>>>>>>>>>>>>>

#name

uname=Label(afterloginwin,text='Name: ',font=('Regular',10),bg='white')

uname.place(x=70,y=250)

uname\_p=Label(afterloginwin,text=n,font=('Regular',10),bg='white')

uname\_p.place(x=180,y=250)

#father name

fname=Label(afterloginwin,text='Father Name: ',font=('Regular',10),bg='white')

fname.place(x=70,y=280)

fname\_p=Label(afterloginwin,text=f,font=('Regular',10),bg='white')

fname\_p.place(x=180,y=280)

#gender

ugender=Label(afterloginwin,text='Gender: ',font=('Regular',10),bg='white')

ugender.place(x=70,y=310)

ugender\_p=Label(afterloginwin,text=g,font=('Regular',10),bg='white')

ugender\_p.place(x=180,y=310)

#date of birth

udob=Label(afterloginwin,text='Date Of Birth: ',font=('Regular',10),bg='white')

udob.place(x=70,y=340)

udob\_p=Label(afterloginwin,text=d,font=('Regular',10),bg='white')

udob\_p.place(x=180,y=340)

#phone no

uphone=Label(afterloginwin,text='Phone Number : ',font=('Regular',10),bg='white')

uphone.place(x=70,y=370)

uphone\_p=Label(afterloginwin,text=p,font=('Regular',10),bg='white')

uphone\_p.place(x=180,y=370)

#aadhaar no

uaadhaar=Label(afterloginwin,text='Linked Aadhaar: ',font=('Regular',10),bg='white')

uaadhaar.place(x=70,y=400)

uaadhaar\_p=Label(afterloginwin,text=aah,font=('Regular',10),bg='white')

uaadhaar\_p.place(x=180,y=400)

uvstatus=Label(afterloginwin,text='Voting Status \_: ',font=('Regular',10),bg='white')

uvstatus.place(x=70,y=430)

uvstatus\_p=Label(afterloginwin,text=vs,font=('Regular',10),bg='white',fg='red')

uvstatus\_p.place(x=180,y=430)

#>>>>>>>>>>>>

#village

uvill=Label(afterloginwin,text='Vill: ',font=('Regular',10),bg='white')

uvill.place(x=340,y=250)

uvill\_p=Label(afterloginwin,text=vill,font=('Regular',10),bg='white')

uvill\_p.place(x=390,y=250)

#post

upost=Label(afterloginwin,text='Post: ',font=('Regular',10),bg='white')

upost.place(x=340,y=280)

upost=Label(afterloginwin,text=po,font=('Regular',10),bg='white')

upost.place(x=390,y=280)

#Pin

upin=Label(afterloginwin,text='Pin : ',font=('Regular',10),bg='white')

upin.place(x=340,y=310)

upin=Label(afterloginwin,text=pin,font=('Regular',10),bg='white')

upin.place(x=390,y=310)

#district

udist=Label(afterloginwin,text='Dist: ',font=('Regular',10),bg='white')

udist.place(x=340,y=340)

udist=Label(afterloginwin,text=dist,font=('Regular',10),bg='white')

udist.place(x=390,y=340)

#stste

ustate=Label(afterloginwin,text='State: ',font=('Regular',10),bg='white')

ustate.place(x=340,y=370)

ustate=Label(afterloginwin,text=state,font=('Regular',10),bg='white')

ustate.place(x=390,y=370)

#button area >>>>>>>>>>>>>>>>>>>>>>

#give vote button img import

gvote\_bg=PhotoImage(file='button/gvote.png')

gvote=Button(afterloginwin,image=gvote\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=give)

gvote.place(x=105,y=530)

def upd():

lw=Label(afterloginwin,text='Sorry You Can Not Update Any Detaile, It\' Possiable After Upcomming Upgrade',fg='red',bg='white' )

lw.place(x=75,y=495)

#Update button

upd\_bg=PhotoImage(file='button/upd.png')

upd=Button(afterloginwin,image=upd\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=upd)

upd.place(x=232,y=530)

def ulogout():

afterloginwin.destroy()

logwin.destroy()

#logout button img import

logout\_bg=PhotoImage(file='button/ulogout.png')

logout=Button(afterloginwin,image=logout\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=ulogout)

logout.place(x=360,y=530)

afterloginwin.mainloop()

else:

#cancel button img import

Warning\_u\_p=Label(logwin,text='Wrong User ID or Password',bg='white',fg='red')

Warning\_u\_p.place(x=690,y=180)

def forgetpass():

try:

db=sqlite3.connect('voting management system.db')

cr=db.cursor()

except:

print('Forgt db is running...')

forgetwin=Toplevel()

forgetwin.title('Forget Password')

forgetwin.geometry('550x680+485+25')

forgetwin.wm\_iconbitmap('icon/login.ico')

forgetwin.resizable(False,False)

entphone=StringVar()

#send OTP

def sendotpp():

fphone\_num=entphone.get()

cr.execute('select phone from registration')

fp\_data=cr.fetchall()

fcheck=0

for ff in fp\_data:

fp=ff[0]

if fphone\_num==fp:

fcheck=1

break

else:

fcheck=0

if fcheck==1:

# print('phone nummber matched')

#hide invalid phone number error

invalid\_phone=Label(forgetwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_phone.place(x=250,y=94)

#resend button

sendotp=Button(forgetwin,text=' Resend OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendotpp)

sendotp.place(x=198,y=130)

'''------------------------------------------------------'''

#generate otp.........

fotp=str(random.randint(1111,9999))

#print('otp is: ',fotp)

#

'''-------------------TESTING OTP-------------------------'''

'''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+fp

# msg='Online Voting System\nForget Password OTP: '+fotp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

#TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

'''--------------------- Testing Otp ----------------------'''

otpnote='Only for Testing Purpose OTP: '+fotp

testing=Label(forgetwin,text=otpnote,font=('Regular',8),fg='red')

testing.place(x=5,y=650)

#update otp..............

cr.execute('update registration set otp=? where phone=?',(fotp,fp))

db.commit()

#print('changed otp')

rotp=StringVar()

def verifyfotp():

rsotp=rotp.get()

cr.execute(f'select otp from registration where phone={fp}')

fetchotp=cr.fetchall()

for fe in fetchotp:

dotp=fe[0]

#if matched otp............

if dotp==rsotp:

#hide invalid otp error

invalid\_OTP=Label(forgetwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verified otp button

Verifyotp=Button(forgetwin,text='✓ Verified ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verifyfotp)

Verifyotp.place(x=205,y=290)

pass1=StringVar()

pass2=StringVar()

def changepass():

pass11=pass1.get()

pass22=pass2.get()

if pass11==pass22:

cr.execute('update registration set password=?,conform\_password=? where phone=?',(pass11,pass22,fp))

db.commit()

# print('pass changed')

#close forgetwindow............

forgetwin.destroy()

changedwin=Toplevel()

changedwin.title('Successfully changed Password')

changedwin.geometry('550x680+485+25')

changedwin.wm\_iconbitmap('icon/login.ico')

changedwin.resizable(False,False)

changedwin.overrideredirect()

ch\_img=PhotoImage(file='logo/check.png')

ch\_l=Label(changedwin,image=ch\_img)

ch\_l.place(x=215,y=100)

ph\_l=Label(changedwin,text='Your Password Successfully changed',font=('times new roman',20),fg='blue')

ph\_l.place(x=70,y=230)

changedwin.mainloop()

else:

# print('conform pass not match')

#password not match

invalid\_pass=Label(forgetwin,text='Conform Password Not Matched',font=('times new roman',8),fg='red')

invalid\_pass.place(x=250,y=462)

#import password image

pass\_f=Label(forgetwin,text='New Password',font=('times new roman',18),bg='white')

pass\_f.place(x=80,y=380)

#entry password

pass\_fe=Entry(forgetwin,font=('timew new roman',18),bd=1,bg='#EFEFE0',width=15,textvar=pass1)

pass\_fe.place(x=250,y=380)

#import conform password image

con\_pass\_e=Label(forgetwin,text='Re... Password',font=('times new roman',18),bg='white')

con\_pass\_e.place(x=80,y=430)

#entry conform password

conpass\_fe=Entry(forgetwin,font=('timew new roman',18),bd=1,bg='#EFEFE0',width=15,show='\*',textvar=pass2)

conpass\_fe.place(x=250,y=430)

#login button img import

#sub1\_img=PhotoImage(file='button/u\_login.png')

sub11=Button(forgetwin,text='Submit',font=('Regular',18),fg='white',bg='#3CCF4E',cursor='hand2',command=changepass)

sub11.place(x=145,y=550)

#cancel button img import

#canc\_bg=PhotoImage(file='button/cancel.png')

canc1=Button(forgetwin,text='Cancel',font=('Regular',18),fg='white',bg='#F55353',cursor='hand2',command=lambda:forgetwin.destroy())

canc1.place(x=295,y=550)

else:

# print('Please Enter Valid OTP')

invalid\_OTP=Label(forgetwin,text='Enter Valid OTP',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verify otp..........

otp\_l1=Label(forgetwin,text='Enter OTP ',font=('Regular',18))

otp\_l1.place(x=110,y=200)

otp\_s1=Label(forgetwin,text='OTP Send Your Mobile Number',font=('Regular',8))

otp\_s1.place(x=245,y=230)

otp\_e1=Entry(forgetwin,font=('Regular',18),bg='#DFF6FF',width=13,textvar=rotp)

otp\_e1.place(x=245,y=200)

Verifyotp=Button(forgetwin,text='Verify OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verifyfotp)

Verifyotp.place(x=205,y=290)

else:

#print('Phone number not match')

invalid\_phone=Label(forgetwin,text='Please Enter Registered Phone Number',font=('times new roman',8),fg='red')

invalid\_phone.place(x=250,y=90)

#label of phone number

fphone=Label(forgetwin,text='Phone Number :',font=('times new roman',18,'bold'),fg='black',bd=0)

fphone.place(x=50,y=50)

#enter phone number

fphone\_e=Entry(forgetwin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entphone)

fphone\_e.place(x=250,y=50)

sendotp=Button(forgetwin,text='✓ Send OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendotpp)

sendotp.place(x=198,y=130)

forgetwin.mainloop()

#frame bg image import

fbg=PhotoImage(file='themes/login form.png')

login\_background=Label(logwin,image=fbg,bg='#D9D9D9',width=518,height=654)

login\_background.place(x=510,y=10)

#uid label img import

uid\_bg=PhotoImage(file='button/uid.png')

uid\_l=Label(logwin,image=uid\_bg,bd=0,bg='white')

uid\_l.place(x=565,y=240)

#uid input ----\*\*\* Entry Box \*\*\*----

uid\_e=Entry(logwin,font=('times new roman',24),bd=1,width=15,bg='#EFEFE0',textvar=uid)

uid\_e.place(x=730,y=242)

#password label img import

pass\_bg=PhotoImage(file='button/password.png')

password=Label(logwin,image=pass\_bg,bd=0,bg='white')

password.place(x=565,y=320)

#password input ----\*\*\* Entry Box \*\*\*----

pass\_e=Entry(logwin,font=('times new roman',24),bd=1,width=15,bg='#EFEFE0',show='\*',textvar=upass)

pass\_e.place(x=730,y=322)

#login button img import

log\_bg=PhotoImage(file='button/u\_login.png')

login=Button(logwin,image=log\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=login)

login.place(x=565,y=410)

#cancel button img import

cancel\_bg=PhotoImage(file='button/cancel.png')

login=Button(logwin,image=cancel\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:logwin.destroy())

login.place(x=730,y=410)

#forget button img import

#forget\_bg=PhotoImage(file='button/forget.png')

forget=Button(logwin,text='Forget Pasword?',font=('Regular',11),bd=0,bg='white',fg='blue',activebackground='white',cursor='hand2',command=forgetpass)

forget.place(x=570,y=512)

#slash / button img import

slash\_bg=PhotoImage(file='button/slash.png')

slash=Button(logwin,image=slash\_bg,bd=0,bg='white',activebackground='white')

slash.place(x=705,y=512)

#Create new account button img import

#create\_new\_ac\_bg=PhotoImage(file='button/create\_new\_ac.png')

create=Button(logwin,text='Apply For New Candidate',font=('Regular',11),bd=0,bg='white',fg='blue',activebackground='white',cursor='hand2',command=RegistrationForm)

create.place(x=730,y=512)

logwin.mainloop()

def givevote\_now():

gv2=Toplevel()

gv2.title('Give Vote')

gv2.geometry('550x680+480+25')

gv2.wm\_iconbitmap('icon/vote-sign.ico')

gv2.resizable(False,False)

cr=db.cursor()

entcard=StringVar()

def sub\_vot\_otp():

inputcard\_no=entcard.get()

cr.execute(f'select card\_no from registration')

data22=cr.fetchall()

vnc=0

for ii in data22:

vn=ii[0]

if inputcard\_no == vn:

vnc=1

break

else:

vnc=0

if vnc==1:

# print('match')

#print('Phone number not match')

invalid\_card=Label(gv2,text=' ',font=('times new roman',8),fg='red',width=30)

invalid\_card.place(x=250,y=90)

cr.execute(f'select phone,voting\_status from registration where card\_no={vn}')

fvts=cr.fetchall()

for jj in fvts:

vtphone=jj[0]

votingstatus=jj[1]

# print(votingstatus)

# print(vtphone)

if votingstatus=='Not Voted':

null=0

v1=IntVar()

def give():

v2=v1.get()

if v2==null:

#choose party.......

cpp\_l=Label(gv2,text='Please Choose Party',font=('Regular',8),fg='red')

cpp\_l.place(x=213,y=310)

v3='please choose party'

elif v2==1:

v3='BJP'

elif v2==2:

v3='TMC'

else:

v3='Congress'

if v3=='BJP' or v3=='TMC' or v3=='Congress':

# print(v3)

cpp\_l=Label(gv2,text=' ',font=('Regular',8),fg='red',width=30)

cpp\_l.place(x=213,y=310)

#gvote\_bg=PhotoImage(file='button/verify.png')

gvote=Button(gv2,text=' Verified ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=give)

gvote.place(x=210,y=330)

'''------------------------------------------------------'''

#generate otp.........

motpp=str(random.randint(1111,9999))

# print(motpp)

#

'''-------------------TESTING OTP-------------------------'''

ottp='Only for testing purposr OTP: ' +motpp

test\_l=Label(gv2,text=ottp,fg='red')

test\_l.place(x=10,y=630)

# '''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+vtphone

# msg='Online Voting System\nyou can give vote\nyour OTP is: '+motpp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

cr.execute(f'update registration set otp={motpp} where card\_no={vn}')

db.commit()

rsotp=StringVar()

def urs\_otp():

reseveotp=rsotp.get()

cr.execute(f'select voting\_status,otp from registration where card\_no={vn}')

otpdata=cr.fetchall()

for ot in otpdata:

vts=ot[0]

sotp=ot[1]

already='Already Voted'

if sotp==reseveotp:

if vts=='Not Voted':

cr.execute('update registration set voting\_status=?,vot\_status=? where card\_no=?',(already,v3,vn))

db.commit()

#Thanks for vote...................

successwin=Toplevel()

successwin.title('success full vote')

successwin.geometry('550x680+487+25')

successwin.wm\_iconbitmap('icon/feedback.ico')

successwin.resizable(False,False)

#successwin.attributes('-fullscreen',True)

successwin.overrideredirect(1)

note='Thank You For Give Vote...'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(successwin,image=gv\_img,width=500,height=600)

gv\_l.place(x=33,y=10)

warning\_img=PhotoImage(file='logo/check.png')

war\_l=Label(successwin,image=warning\_img,width=200,height=100,bg='white')

war\_l.place(x=180,y=70)

not\_l=Label(successwin,text=note,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=150,y=200)

def closesuccess():

successwin.destroy()

gv2.destroy()

#cancel button img import

tgv\_bg=PhotoImage(file='button/close.png')

tgv=Button(successwin,image=tgv\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=closesuccess)

tgv.place(x=230,y=370)

successwin.mainloop()

else:

#already voted...................

alwin=Toplevel()

alwin.title('Already Voted')

alwin.geometry('550x680+487+25')

alwin.wm\_iconbitmap('icon/feedback.ico')

alwin.resizable(False,False)

#alwin.attributes('-fullscreen',True)

alwin.overrideredirect(1)

note='One user can give vote at once time\nYou Have Already Voted...\nSorry you can\'t give vote again\n Because Our System Checks You...'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(alwin,image=gv\_img,width=500,height=600)

gv\_l.place(x=30,y=10)

warning\_img=PhotoImage(file='logo/warning.png')

war\_l=Label(alwin,image=warning\_img,width=200,height=100,bg='white')

war\_l.place(x=180,y=70)

not\_l=Label(alwin,text=note,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=85,y=200)

#cancel button img import

canc\_bg=PhotoImage(file='button/close.png')

canc=Button(alwin,image=canc\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:alwin.destroy())

canc.place(x=225,y=370)

alwin.mainloop()

else:

#print('wrong otp')

w\_l=Label(gv2,text='Please Enter Valid OTP',fg='red',width=50)

w\_l.place(x=130,y=430)

#otp..........

otp\_l=Label(gv2,text='Enter OTP ',font=('Regular',18))

otp\_l.place(x=110,y=400)

otp\_s=Label(gv2,text='4 Digit OTP Send Your Mobile Number',font=('Regular',8))

otp\_s.place(x=245,y=430)

otp\_e=Entry(gv2,font=('Regular',18),bg='#DFF6FF',width=16,textvar=rsotp)

otp\_e.place(x=245,y=400)

#login button img import

#sub1\_img=PhotoImage(file='button/u\_login.png')

sub1=Button(gv2,text='Submit',font=('Regular',18),fg='white',bg='#3CCF4E',cursor='hand2',command=urs\_otp)

sub1.place(x=140,y=490)

#cancel button img import

#canc\_bg=PhotoImage(file='button/cancel.png')

canc=Button(gv2,text='Cancel',font=('Regular',18),fg='white',bg='#F55353',cursor='hand2',command=lambda:gv2.destroy())

canc.place(x=310,y=490)

#choose party.......

cp\_l=Label(gv2,text='Choose Your Favourite Party',font=('Regular',22),fg='blue')

cp\_l.place(x=96,y=200)

cp1\_e=Radiobutton(gv2,text='BJP',font=('times new roman',18),fg='#FF9F29',activebackground='white',value=1,cursor='hand2',variable=v1)

cp1\_e.place(x=110,y=265)

cp2\_e=Radiobutton(gv2,text='TMC',font=('times new roman',18),fg='green',activebackground='white',value=2,cursor='hand2',variable=v1)

cp2\_e.place(x=220,y=265)

cp3\_e=Radiobutton(gv2,text='Congress',font=('times new roman',18),fg='red',activebackground='white',value=3,cursor='hand2',variable=v1)

cp3\_e.place(x=340,y=265)

#gvote\_bg=PhotoImage(file='button/verify.png')

gvote=Button(gv2,text='✓ Verify ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=give)

gvote.place(x=210,y=330)

else:

# print('cant not take vote')

#already voted...................

alwin=Toplevel()

alwin.title('Already Voted')

alwin.geometry('550x680+487+25')

alwin.wm\_iconbitmap('icon/feedback.ico')

alwin.resizable(False,False)

#alwin.attributes('-fullscreen',True)

alwin.overrideredirect(1)

note='One user can give vote at once time\nYou Have Already Voted...\nSorry you can\'t give vote again\n Because Our System Checks You...'

#background...

gv\_img=PhotoImage(file='themes/voteback.png')

gv\_l=Label(alwin,image=gv\_img,width=500,height=600)

gv\_l.place(x=30,y=10)

warning\_img=PhotoImage(file='logo/warning.png')

war\_l=Label(alwin,image=warning\_img,width=200,height=100,bg='white')

war\_l.place(x=180,y=70)

not\_l=Label(alwin,text=note,font=('Regular',18),bg='white',fg='blue')

not\_l.place(x=85,y=200)

#cancel button img import

canc\_bg=PhotoImage(file='button/close.png')

canc=Button(alwin,image=canc\_bg,font=('Regular',18),bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:alwin.destroy())

canc.place(x=225,y=370)

alwin.mainloop()

else:

#print('Phone number not match')

invalid\_card=Label(gv2,text='Please Enter Valid Voter ID',font=('times new roman',8),fg='red')

invalid\_card.place(x=250,y=90)

#label of phone number

card\_num=Label(gv2,text='Enter Voter ID :',font=('times new roman',18,'bold'),fg='black',bd=0)

card\_num.place(x=50,y=55)

#enter phone number

card\_num\_e=Entry(gv2,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entcard)

card\_num\_e.place(x=250,y=50)

send\_card\_otp=Button(gv2,text=' Submit ', fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sub\_vot\_otp)

send\_card\_otp.place(x=210,y=130)

gv2.mainloop()

#required to login......

def required\_to\_login():

log\_result=messagebox.askyesno('Login Required','You Can\'t Change/Update any Details Now\n It\'s Possiable After UpComming Update\n Our Team Workin On This System for better Upgrad...\n \nAre You Want To Login Now ?')

if log\_result == True:

LoginForm()

#Track Application..................................>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

def trackapplication():

trackwin=Toplevel()

trackwin.geometry('550x680+490+50')

trackwin.wm\_iconbitmap('icon/track.ico')

trackwin.title('Track Application Status')

trackwin.resizable(False,False)

trackwin.config(background='#EAF6F6')

db=sqlite3.connect('voting management system.db')

cr=db.cursor()

tid=StringVar()

def aptrack():

apno=tid.get()

#print(apno)

cr.execute('select application\_no from registration')

data=cr.fetchall()

db.commit()

for i in data:

a=i[0]

check=0

if apno==a:

check=1

break

else:

check=0

if check==1:

cr.execute(f'select card\_no from registration where application\_no={a}')

card\_no=cr.fetchall()

for k in card\_no:

card=k[0]

w='You are Voter Id is ' +card

track = Label(trackwin, text=w, font=('Regular', 16), fg='blue',width=45,bg='#EAF6F6')

track.place(x=0, y=270)

else:

track1 = Label(trackwin, text='Please Enter Valid Application No', font=('Regular', 8), fg='red',bg='#EAF6F6')

track1.place(x=240, y=135)

L2=Label(trackwin,text='Application No :',font=('times new roman',20,'bold'),bg='#EAF6F6',fg='black',bd=0)

L2.place(x=25,y=100)

E1=Entry(trackwin,font=('times new roman',20,'bold'),textvar=tid)

E1.place(x=240,y=100)

B1\_img=PhotoImage(file='button/appsearch.png')

B1=Button(trackwin,image=B1\_img,bd=0,bg='#EAF6F6',fg='black',command=aptrack,cursor='hand2')

B1.place(x=200,y=185)

trackwin.mainloop()

def adminlogin\_from():

adlogwin=Toplevel()

adlogwin.title('Administrator Login')

adlogwin.geometry('550x670+485+25')

adlogwin.wm\_iconbitmap('icon/admin.ico')

adlogwin.resizable(False,False)

#frame bg image import

fbg=PhotoImage(file='themes/adlog.png')

login\_background=Label(adlogwin,image=fbg,width=518,height=654)

login\_background.place(x=15,y=8)

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> Database Connect <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

null=0

uid=StringVar()

upass=StringVar()

def adlogin():

userid=uid.get()

upassword=upass.get()

#userid.................

userid\_length=0

userid\_length=len(userid)

if null == userid\_length:

#validation.....

validation=Label(adlogwin,text='Please Enter User ID',font=('Regular',10),fg='red',bg='white')

validation.place(x=240,y=270)

else:

#password..............

password\_length=len(upassword)

if null == password\_length:

p='Please Enter Password'

validation=Label(adlogwin,text=p,font=('Regular',10),fg='red',bg='white')

validation.place(x=240,y=360)

else:

cr.execute("select uid,password from admin")

data=cr.fetchall()

check=0

for i in data:

a=i[0]

b=i[1]

if userid==a and upassword==b:

check=1

break

else:

check=0

if check == 1:

#//////////////////////////////////////<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

# admin login successfull

adminwin=Toplevel()

adminwin.title('Admin Panel')

adminwin.geometry('600x600')

adminwin.minsize(600,400)

adminwin.wm\_iconbitmap('icon/admin.ico')

adminwin.resizable(False,False)#maximize option disable

adminwin.attributes('-fullscreen',True)

adminwin.state('zoomed')#default open fullscreen

#-------------------------------------- Main Body ---------------------------------------- #

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

#approved candidate

s=ttk.Style()

s.theme\_use('clam')

s.configure('Treeview', rowheight=20)

tv=ttk.Treeview(adminwin)

tv['columns']=('1','2','3','4','5','6','7','8','9','10','11','12','13','14','15','16')

tv.column('#0',width='0')

tv.column('1',width='40')

tv.column('2',width='150')

tv.column('3',width='60')

tv.column('4',width='50')

tv.column('5',width='150')

tv.column('6',width='70')

tv.column('7',width='80')

tv.column('8',width='100')

tv.column('9',width='100')

tv.column('10',width='50')

tv.column('11',width='80')

tv.column('12',width='80')

tv.column('13',width='60')

tv.column('14',width='88')

tv.column('15',width='85')

tv.column('16',width='80')

# name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,application\_no,card\_no,voting\_status,Voted by

tv.heading('#0',text='+')

tv.heading('1',text='Sl No')

tv.heading('2',text='Name')

tv.heading('3',text='DOB')

tv.heading('4',text='Gender')

tv.heading('5',text='Father Name')

tv.heading('6',text='Phone No')

tv.heading('7',text='Aadhaar No')

tv.heading('8',text='Village')

tv.heading('9',text='Post')

tv.heading('10',text='Pin')

tv.heading('11',text='District')

tv.heading('12',text='State')

tv.heading('13',text='Application No')

tv.heading('14',text='Card No')

tv.heading('15',text='Voting Status')

tv.heading('16',text='Voted By')

cr.execute('select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,application\_no,card\_no,voting\_status,vot\_status from registration')

adata=cr.fetchall()

fdata=0

a=0

for i in adata:

f=i[12]

if f!='Not Generated':

fdata += 1

a += 1

tv.insert(parent="", index="end", iid=i, text="", values=((a,i[0],i[1],i[2],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[11],i[12],i[13],i[14])))

tv.place(x=90,y=130)

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# not approve candidate..........

tv=ttk.Treeview(adminwin)

tv['columns']=('1','2','3','4','5','6','7','8','9','10','11','12','13','14','15','16')

tv.column('#0',width='0')

tv.column('1',width='40')

tv.column('2',width='150')

tv.column('3',width='60')

tv.column('4',width='50')

tv.column('5',width='150')

tv.column('6',width='70')

tv.column('7',width='80')

tv.column('8',width='100')

tv.column('9',width='100')

tv.column('10',width='50')

tv.column('11',width='80')

tv.column('12',width='80')

tv.column('13',width='60')

tv.column('14',width='88')

tv.column('15',width='85')

tv.column('16',width='80')

# name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,application\_no,card\_no,voting\_status,Voted by

tv.heading('#0',text='+')

tv.heading('1',text='Sl No')

tv.heading('2',text='Name')

tv.heading('3',text='DOB')

tv.heading('4',text='Gender')

tv.heading('5',text='Father Name')

tv.heading('6',text='Phone No')

tv.heading('7',text='Aadhaar No')

tv.heading('8',text='Village')

tv.heading('9',text='Post')

tv.heading('10',text='Pin')

tv.heading('11',text='District')

tv.heading('12',text='State')

tv.heading('13',text='Application No')

tv.heading('14',text='Card No')

tv.heading('15',text='Voting Status')

tv.heading('16',text='Voted By')

cr.execute('select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,application\_no,card\_no,voting\_status,vot\_status from registration')

adata=cr.fetchall()

fdata=0

a=0

for i in adata:

f=i[12]

if f=='Not Generated':

fdata += 1

a += 1

tv.insert(parent="", index="end", iid=i, text="", values=((a,i[0],i[1],i[2],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[11],i[12],i[13],i[14])))

tv.place(x=90,y=420)

#>......>>>>>>>>>>>>>>............>>>>>>>>>>>>>>>...................>>>>>>>>>>>>>>>>..............>>>>>>>>>>>>>>>>.

topnav=Label(adminwin,text='Welcome to Administrator Panel',font=('times new roman',28),fg='white',bg='blue')

topnav.pack(fill=X)

candidate=Label(adminwin,text='Approved Candidate Details',font=('times new roman',20),fg='blue')

candidate.place(x=90,y=90)

exit=Button(adminwin,text='Exit / Logout', font=('times new roman',14),fg='white',bg='#FF8B8B',command=lambda:adminwin.destroy(),cursor='hand2')

exit.place(x=1270,y=90)

notapcandidate=Label(adminwin,text='Not Approved Candidate Details',font=('times new roman',20),fg='blue')

notapcandidate.place(x=90,y=380)

def approved():

approvewin=Toplevel()

approvewin.title('Generate Card Number')

approvewin.geometry('550x680+485+25')

approvewin.wm\_iconbitmap('icon/aprove.ico')

approvewin.resizable(False,False)

entcard=StringVar()

#send OTP

def sendcardotp():

cardn=entcard.get()

cr.execute('select application\_no,card\_no from registration')

sc\_data=cr.fetchall()

cfetch=0

for sc in sc\_data:

sc1=sc[0]

crd=sc[1]

if cardn==sc1:

cfetch=1

break

else:

cfetch=0

if cfetch==1:

invalid\_phone=Label(approvewin,text='',font=('times new roman',8),width=50)

invalid\_phone.place(x=265,y=90)

if crd=='Not Generated':

c1=random.randint(1111111111,9999999999)

cr.execute('update registration set card\_no=? where application\_no=?',(c1,sc1))

db.commit()

approvewin.destroy()

apsuccesswin=Toplevel()

apsuccesswin.title('Generate Card Number')

apsuccesswin.geometry('550x680+485+25')

apsuccesswin.wm\_iconbitmap('icon/aprove.ico')

apsuccesswin.resizable(False,False)

warning\_img=PhotoImage(file='logo/check.png')

war\_l=Label(apsuccesswin,image=warning\_img,width=200,height=100)

war\_l.place(x=165,y=70)

ge=Label(apsuccesswin,text="Successfully Generated Candidate Card Number...",font=('times new roman',14),fg='blue')

ge.place(x=80,y=200)

apsuccesswin.mainloop()

else:

apsuccesswin=Toplevel()

apsuccesswin.title('Already Generate Card Number')

apsuccesswin.geometry('550x680+485+25')

apsuccesswin.wm\_iconbitmap('icon/notap.ico')

apsuccesswin.resizable(False,False)

warning\_img=PhotoImage(file='logo/warning.png')

war\_l=Label(apsuccesswin,image=warning\_img,width=200,height=100)

war\_l.place(x=165,y=70)

ge=Label(apsuccesswin,text="Already Generated Card Number...",font=('times new roman',14),fg='blue')

ge.place(x=135,y=200)

apsuccesswin.mainloop()

else:

invalid\_phone=Label(approvewin,text='Please Enter Valid Application Number',font=('times new roman',8),fg='red')

invalid\_phone.place(x=265,y=90)

invalid=Label(approvewin,text='Invalid Application Number',font=('times new roman',18),fg='red')

invalid.place(x=135,y=300)

#label of phone number

card\_num=Label(approvewin,text='Enter Application No: ',font=('times new roman',18,'bold'),fg='black',bd=0)

card\_num.place(x=30,y=55)

#enter phone number

card\_num\_e=Entry(approvewin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entcard)

card\_num\_e.place(x=265,y=50)

send\_card\_otp=Button(approvewin,text=' Generated ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=205,y=130)

approvewin.mainloop()

def suspend():

approvewin=Toplevel()

approvewin.title('Suspend / Delete User')

approvewin.geometry('550x680+485+25')

approvewin.wm\_iconbitmap('icon/suspend.ico')

approvewin.resizable(False,False)

entcard=StringVar()

#send OTP

def sendcardotp():

cardn=entcard.get()

cr.execute('select application\_no,card\_no,phone from registration')

sc\_data=cr.fetchall()

cfetch=0

for sc in sc\_data:

sc1=sc[0]

crd=sc[1]

ph=sc[2]

if cardn==sc1 or cardn==crd:

cfetch=1

break

else:

cfetch=0

if cfetch==1:

invalid\_phone=Label(approvewin,text='',font=('times new roman',8),width=50)

invalid\_phone.place(x=265,y=90)

cr.execute('delete from registration where phone=?',(ph,))

db.commit()

apsuccesswin=Toplevel()

apsuccesswin.title('Generate Card Number')

apsuccesswin.geometry('550x680+485+25')

apsuccesswin.wm\_iconbitmap('icon/suspend.ico')

apsuccesswin.resizable(False,False)

warning\_img=PhotoImage(file='logo/check.png')

war\_l=Label(apsuccesswin,image=warning\_img,width=200,height=100)

war\_l.place(x=165,y=70)

ge=Label(apsuccesswin,text="Candidate Card Number Permanently Deleted...",font=('times new roman',14),fg='red')

ge.place(x=80,y=200)

apsuccesswin.mainloop()

else:

invalid\_phone=Label(approvewin,text='Enter Valid Application / Card Number',font=('times new roman',8),fg='red')

invalid\_phone.place(x=265,y=90)

invalid=Label(approvewin,text='Enter Application / Card Number Already Suspend \nor\n Not Registered Our System',font=('times new roman',18),fg='red')

invalid.place(x=25,y=300)

#label of phone number

card\_num=Label(approvewin,text='Application / Card No: ',font=('times new roman',18,'bold'),fg='black',bd=0)

card\_num.place(x=20,y=55)

#enter phone number

card\_num\_e=Entry(approvewin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entcard)

card\_num\_e.place(x=275,y=50)

send\_card\_otp=Button(approvewin,text=' Suspend ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=198,y=130)

approvewin.mainloop()

approve=Button(adminwin,text=' Generate Card Number', font=('times new roman',14),bg='#00FFDD',cursor='hand2',command=approved)

approve.place(x=520,y=700)

suspend=Button(adminwin,text='Suspend Candidate', font=('times new roman',14),bg='#FC4F4F',cursor='hand2',command=suspend)

suspend.place(x=780,y=700)

adminwin.mainloop()

#//////////////////////////////////////<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

else:

#cancel button img import

wp=Label(adlogwin,text='Wrong User ID or Password',bg='white',fg='red')

wp.place(x=195,y=180)

def forgetadminpass():

forgetwin=Toplevel()

forgetwin.title('Forget User ID or Password')

forgetwin.geometry('550x680+485+25')

forgetwin.wm\_iconbitmap('icon/admin.ico')

forgetwin.resizable(False,False)

entphone=StringVar()

#send OTP

def sendotpp():

fphone\_num=entphone.get()

cr.execute('select phone from admin')

fp\_data=cr.fetchall()

fcheck=0

for ff in fp\_data:

fp=ff[0]

if fphone\_num==fp:

fcheck=1

break

else:

fcheck=0

if fcheck==1:

# print('phone nummber matched')

#hide invalid phone number error

invalid\_phone=Label(forgetwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_phone.place(x=250,y=94)

#resend button

sendotp=Button(forgetwin,text=' Resend OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendotpp)

sendotp.place(x=198,y=130)

'''------------------------------------------------------'''

#generate otp.........

fotp=str(random.randint(1111,9999))

#print('otp is: ',fotp)

#

'''-------------------TESTING OTP-------------------------'''

'''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+fp

# msg='Online Voting System\nForget Admin Password OTP: '+fotp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

#TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

'''--------------------- Testing Otp ----------------------'''

otpnote='Only for Testing Purpose OTP: '+fotp

testing=Label(forgetwin,text=otpnote,font=('Regular',8),fg='red')

testing.place(x=5,y=650)

#update otp..............

cr.execute('update admin set otp=? where phone=?',(fotp,fp))

db.commit()

#print('changed otp')

rotp=StringVar()

def verifyfotp():

rsotp=rotp.get()

cr.execute(f'select uid,otp from admin where phone={fp}')

fetchotp=cr.fetchall()

for fe in fetchotp:

ud=fe[0]

dotp=fe[1]

#if matched otp............

if dotp==rsotp:

#hide invalid otp error

invalid\_OTP=Label(forgetwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verified otp button

Verifyotp=Button(forgetwin,text='✓ Verified ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verifyfotp)

Verifyotp.place(x=205,y=290)

pass1=StringVar()

pass2=StringVar()

def changepass():

pass11=pass1.get()

pass22=pass2.get()

if pass11==pass22:

cr.execute('update admin set password=? where phone=?',(pass11,fp))

db.commit()

# print('pass changed')

#close forgetwindow............

forgetwin.destroy()

changedwin=Toplevel()

changedwin.title('Successfully changed Password')

changedwin.geometry('550x680+485+25')

changedwin.wm\_iconbitmap('icon/login.ico')

changedwin.resizable(False,False)

changedwin.overrideredirect()

ch\_img=PhotoImage(file='logo/check.png')

ch\_l=Label(changedwin,image=ch\_img)

ch\_l.place(x=215,y=100)

ph\_l=Label(changedwin,text='Your Password Successfully changed',font=('times new roman',20),fg='blue')

ph\_l.place(x=70,y=230)

changedwin.mainloop()

else:

# print('conform pass not match')

#password not match

invalid\_pass=Label(forgetwin,text='Conform Password Not Matched',font=('times new roman',8),fg='red')

invalid\_pass.place(x=250,y=462)

#import password image

pass\_f=Label(forgetwin,text='New Password',font=('times new roman',18),bg='white')

pass\_f.place(x=80,y=400)

#entry password

pass\_fe=Entry(forgetwin,font=('timew new roman',18),bd=1,bg='#EFEFE0',width=15,textvar=pass1)

pass\_fe.place(x=250,y=400)

#import conform password image

con\_pass\_e=Label(forgetwin,text='Re... Password',font=('times new roman',18),bg='white')

con\_pass\_e.place(x=80,y=450)

#entry conform password

conpass\_fe=Entry(forgetwin,font=('timew new roman',18),bd=1,bg='#EFEFE0',width=15,show='\*',textvar=pass2)

conpass\_fe.place(x=250,y=450)

#import uid

uuid='Your User ID: '+ud

puid\_f=Label(forgetwin,text=uuid,font=('times new roman',14),fg='red')

puid\_f.place(x=180,y=345)

#login button img import

#sub1\_img=PhotoImage(file='button/u\_login.png')

sub11=Button(forgetwin,text='Submit',font=('Regular',18),fg='white',bg='#3CCF4E',cursor='hand2',command=changepass)

sub11.place(x=145,y=550)

#cancel button img import

#canc\_bg=PhotoImage(file='button/cancel.png')

canc1=Button(forgetwin,text='Cancel',font=('Regular',18),fg='white',bg='#F55353',cursor='hand2',command=lambda:forgetwin.destroy())

canc1.place(x=295,y=550)

else:

# print('Please Enter Valid OTP')

invalid\_OTP=Label(forgetwin,text='Enter Valid OTP',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verify otp..........

otp\_l1=Label(forgetwin,text='Enter OTP ',font=('Regular',18))

otp\_l1.place(x=110,y=200)

otp\_s1=Label(forgetwin,text='OTP Send Your Mobile Number',font=('Regular',8))

otp\_s1.place(x=245,y=230)

otp\_e1=Entry(forgetwin,font=('Regular',18),bg='#DFF6FF',width=13,textvar=rotp)

otp\_e1.place(x=245,y=200)

Verifyotp=Button(forgetwin,text='Verify OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verifyfotp)

Verifyotp.place(x=205,y=290)

else:

#print('Phone number not match')

invalid\_phone=Label(forgetwin,text='Please Enter Registered Phone Number',font=('times new roman',8),fg='red')

invalid\_phone.place(x=250,y=90)

#label of phone number

fphone=Label(forgetwin,text='Phone Number :',font=('times new roman',18,'bold'),fg='black',bd=0)

fphone.place(x=50,y=50)

#enter phone number

fphone\_e=Entry(forgetwin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entphone)

fphone\_e.place(x=250,y=50)

sendotp=Button(forgetwin,text='✓ Send OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendotpp)

sendotp.place(x=198,y=130)

forgetwin.mainloop()

#uid label img import

uid\_bg=PhotoImage(file='button/uid.png')

uid\_l=Label(adlogwin,image=uid\_bg,bd=0,bg='white')

uid\_l.place(x=70,y=230)

#uid input ----\*\*\* Entry Box \*\*\*----

uid\_e=Entry(adlogwin,font=('times new roman',24),bd=1,width=15,bg='#EFEFE0',textvar=uid)

uid\_e.place(x=240,y=232)

#password label img import

pass\_bg=PhotoImage(file='button/password.png')

password=Label(adlogwin,image=pass\_bg,bd=0,bg='white')

password.place(x=70,y=310)

#password input ----\*\*\* Entry Box \*\*\*----

pass\_e=Entry(adlogwin,font=('times new roman',24),bd=1,width=15,bg='#EFEFE0',show='\*',textvar=upass)

pass\_e.place(x=240,y=312)

#login button img import

log\_bg=PhotoImage(file='button/login.png')

login=Button(adlogwin,image=log\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=adlogin)

login.place(x=70,y=410)

#cancel button img import

cancel\_bg=PhotoImage(file='button/cancel.png')

cancel=Button(adlogwin,image=cancel\_bg,bd=0,bg='white',activebackground='white',cursor='hand2',command=lambda:adlogwin.destroy())

cancel.place(x=240,y=410)

#forget button img import

#forget\_bg=PhotoImage(file='button/forget.png')

forget=Button(adlogwin,text='Forget User ID or Pasword?',font=('Regular',11),bd=0,bg='white',fg='blue',activebackground='white',cursor='hand2',command=forgetadminpass)

forget.place(x=75,y=480)

adlogwin.mainloop()

#------------------------------------ Top Nav Area --------------------------------------- #

#Background Themes

theme\_img=PhotoImage(file='themes/background\_theme.png')

theme=Label(root,image=theme\_img)

theme.pack()

#Top Nav button------------->

#About Button

about\_logo=PhotoImage(file='button/about.png')

about\_button=Button(theme,image=about\_logo,bd=0,bg='#1A1A1A',activebackground='#1A1A1A',cursor='hand2',command=about)

about\_button.place(x=1350,y=14)

#feedback.........

feedback\_logo=PhotoImage(file='button/feedback.png')

feedback\_button=Button(theme,image=feedback\_logo,bd=0,bg='#1A1A1A',activebackground='#1A1A1A',cursor='hand2',command=feedback)

feedback\_button.place(x=1180,y=14)

#Admin Button

admin\_logo=PhotoImage(file='button/adminlogin.png')

admin\_button=Button(theme,image=admin\_logo,bd=0,bg='#1A1A1A',cursor='hand2',activebackground='#1A1A1A',command=adminlogin\_from)

admin\_button.place(x=1005,y=14)

#------------------------------------ Top Nav End --------------------------------------- #

#------------------------------------- Main Body End -------------------------------------- #

#################################### Voting Count Area ######################################

# l1=Label(theme,height=8,width=20,bg='red')

# l1.place(x=50,y=120)

# l2=Label(theme,height=8,width=20,bg='blue')

# l2.place(x=230,y=120)

# l3=Label(theme,height=8,width=20,bg='blue')

# l3.place(x=410,y=120)

# l3=Label(theme,height=8,width=20,bg='blue')

# l3.place(x=590,y=120)

def viewvoter\_card():

viewcardwin=Toplevel()

viewcardwin.title('View Voter Card')

viewcardwin.geometry('550x680+485+25')

viewcardwin.wm\_iconbitmap('icon/ind.ico')

viewcardwin.resizable(False,False)

entcard=StringVar()

#send OTP

def sendcardotp():

cardn=entcard.get()

cr.execute('select card\_no from registration')

sc\_data=cr.fetchall()

cfetch=0

for sc in sc\_data:

sc1=sc[0]

if cardn==sc1:

cfetch=1

break

else:

cfetch=0

if cfetch==1:

# print('valid card number')

# print(sc1)

# print('phone nummber matched')

#hide invalid phone number error

invalid\_card=Label(viewcardwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_card.place(x=250,y=94)

#resend button

send\_card\_otp=Button(viewcardwin,text=' Resend OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=198,y=130)

cr.execute(f'select phone from registration where card\_no={sc1}')

fetchcard\_phone=cr.fetchall()

for fpp in fetchcard\_phone:

cpn=fpp[0]

# print(cpn)

#generate otp.........

viewotp=str(random.randint(1111,9999))

# print('otp is: ',viewotp)

#

'''-------------------TESTING OTP-------------------------'''

'''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+cpn

# msg='Online Voting System\nYour card verification OTP is: '+viewotp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

#TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

'''--------------------- Testing Otp ----------------------'''

otpnote\_view='Only for Testing Purpose OTP: '+viewotp

testing=Label(viewcardwin,text=otpnote\_view,font=('Regular',8),fg='red')

testing.place(x=5,y=0)

#update otp..............

cr.execute('update registration set otp=? where phone=?',(viewotp,cpn))

db.commit()

# print('changed/update otp')

ue\_otp=StringVar()

def verify\_ue\_otp():

rs\_otp=ue\_otp.get()

cr.execute(f'select otp from registration where phone={cpn}')

f\_otpdata=cr.fetchall()

for dotpp in f\_otpdata:

ot=dotpp[0]

if ot==rs\_otp:

# print('otp match')

Verifyotpp=Button(viewcardwin,text='✓ Verified ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verify\_ue\_otp)

Verifyotpp.place(x=205,y=290)

def view():

#successfully login

cv=Toplevel()

cv.title('View Voter Card')

cv.geometry('550x680+485+25')

cv.wm\_iconbitmap('icon/ind.ico')

cv.resizable(False,False)

#successfully login

cr.execute(f'select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,card\_no,voting\_status,photo from registration where phone={cpn}')

data=cr.fetchall()

for i in data:

n=i[0]

d=i[1]

g=i[2]

f=i[3]

p=i[4]

aah=i[5]

vill=i[6]

po=i[7]

pin=i[8]

dist=i[9]

state=i[10]

cd=i[11]

vs=i[12]

myphoto=i[13]

'''---------------------- Profile Area----------------------'''

#bg thems..............

vc\_p=PhotoImage(file='themes/cardimg.png')

vc\_l=Label(cv,image=vc\_p)

vc\_l.place(x=70,y=120)

fp = io.BytesIO(myphoto)

# load the image

image = Image.open(fp)

res=image.resize((85,100))

# drawing image to top window

userimg = ImageTk.PhotoImage(res)

userphoto=Label(cv,image=userimg)

userphoto.place(x=125,y=185)

#------------------------------------

#card no

ucard\_no=Label(cv,text='Voter ID: ',font=('Regular',8),bg='#44FFDD')

ucard\_no.place(x=111,y=295)

ucard\_p=Label(cv,text=cd,font=('Regular',8),bg='#44FFDD')

ucard\_p.place(x=160,y=295)

#>>>>>>>>>>>>>>>>>>>>>>>

#name

uname=Label(cv,text='Name: ',font=('Regular',8),bg='#44FFDD')

uname.place(x=90,y=325)

uname\_p=Label(cv,text=n,font=('Regular',8),bg='#44FFDD')

uname\_p.place(x=170,y=325)

#father name

fname=Label(cv,text='Father Name: ',font=('Regular',8),bg='#44FFDD')

fname.place(x=90,y=349)

fname\_p=Label(cv,text=f,font=('Regular',8),bg='#44FFDD')

fname\_p.place(x=170,y=349)

#gender

ugender=Label(cv,text='Gender: ',font=('Regular',8),bg='#44FFDD')

ugender.place(x=90,y=373)

ugender\_p=Label(cv,text=g,font=('Regular',8),bg='#44FFDD')

ugender\_p.place(x=170,y=373)

# #date of birth

udob=Label(cv,text='Date Of Birth: ',font=('Regular',8),bg='#44FFDD')

udob.place(x=90,y=398)

udob\_p=Label(cv,text=d,font=('Regular',8),bg='#44FFDD')

udob\_p.place(x=170,y=398)

# #phone no

# uphone=Label(cv,text='Phone Number : ',font=('Regular',8),bg='#44FFDD')

# uphone.place(x=70,y=370)

# uphone\_p=Label(cv,text=p,font=('Regular',8),bg='#44FFDD')

# uphone\_p.place(x=170,y=370)

# #aadhaar no

# uaadhaar=Label(cv,text='Linked Aadhaar: ',font=('Regular',8),bg='#44FFDD')

# uaadhaar.place(x=70,y=400)

# uaadhaar\_p=Label(cv,text=aah,font=('Regular',8),bg='#44FFDD')

# uaadhaar\_p.place(x=170,y=400)

# uvstatus=Label(cv,text='Voting Status \_: ',font=('Regular',8),bg='#44FFDD')

# uvstatus.place(x=70,y=430)

# uvstatus\_p=Label(cv,text=vs,font=('Regular',8),bg='#44FFDD',fg='red')

# uvstatus\_p.place(x=170,y=430)

# #>>>>>>>>>>>>

#village

uvill=Label(cv,text='Vill: ',font=('Regular',8),bg='#44FFDD')

uvill.place(x=300,y=145)

uvill\_p=Label(cv,text=vill,font=('Regular',8),bg='#44FFDD')

uvill\_p.place(x=340,y=145)

#post

upost=Label(cv,text='Post: ',font=('Regular',8),bg='#44FFDD')

upost.place(x=300,y=165)

upost=Label(cv,text=po,font=('Regular',8),bg='#44FFDD')

upost.place(x=340,y=165)

#Pin

upin=Label(cv,text='Pin : ',font=('Regular',8),bg='#44FFDD')

upin.place(x=300,y=187)

upin=Label(cv,text=pin,font=('Regular',8),bg='#44FFDD')

upin.place(x=340,y=187)

#district

udist=Label(cv,text='Dist: ',font=('Regular',8),bg='#44FFDD')

udist.place(x=300,y=207)

udist=Label(cv,text=dist,font=('Regular',8),bg='#44FFDD')

udist.place(x=340,y=207)

#signature........

sign\_img=PhotoImage(file='img/sig.png')

sig\_l=Label(cv,image=sign\_img,bg='#44FFDD')

sig\_l.place(x=355,y=235)

cv.mainloop()

view()

# view\_l=Button(viewcardwin,text=' View Card ',fg='white',font=('Regular',15),bd=0,bg='green',cursor='hand2',padx=10,command=view)

# view\_l.place(x=205,y=380)

else:

# print('Please Enter Valid OTP')

invalid\_OTP=Label(viewcardwin,text='Enter Valid OTP',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verify otp..........

otp\_t1=Label(viewcardwin,text='Enter OTP ',font=('Regular',18))

otp\_t1.place(x=110,y=200)

otp\_w1=Label(viewcardwin,text='OTP Send Your Mobile Number',font=('Regular',8))

otp\_w1.place(x=245,y=230)

otp\_ee=Entry(viewcardwin,font=('Regular',18),bg='#DFF6FF',width=13,textvar=ue\_otp)

otp\_ee.place(x=245,y=200)

Verifyotpp=Button(viewcardwin,text='Verify OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verify\_ue\_otp)

Verifyotpp.place(x=205,y=290)

else:

#print('Phone number not match')

invalid\_card=Label(viewcardwin,text='Please Enter Voter ID',font=('times new roman',8),fg='red')

invalid\_card.place(x=250,y=90)

#label of phone number

card\_num=Label(viewcardwin,text='Enter Voter ID :',font=('times new roman',18,'bold'),fg='black',bd=0)

card\_num.place(x=50,y=55)

#enter phone number

card\_num\_e=Entry(viewcardwin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entcard)

card\_num\_e.place(x=250,y=50)

send\_card\_otp=Button(viewcardwin,text='✓ Send OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=198,y=130)

viewcardwin.mainloop()

def download():

viewcardwin=Toplevel()

viewcardwin.title('Download Voter Card')

viewcardwin.geometry('550x680+485+25')

viewcardwin.wm\_iconbitmap('icon/download.ico')

viewcardwin.resizable(False,False)

entcard=StringVar()

#send OTP

def sendcardotp():

cardn=entcard.get()

cr.execute('select card\_no from registration')

sc\_data=cr.fetchall()

cfetch=0

for sc in sc\_data:

sc1=sc[0]

if cardn==sc1:

cfetch=1

break

else:

cfetch=0

if cfetch==1:

# print('valid card number')

# print(sc1)

# print('phone nummber matched')

#hide invalid phone number error

invalid\_card=Label(viewcardwin,text=' ',font=('times new roman',8),fg='red',width=50)

invalid\_card.place(x=250,y=94)

#resend button

send\_card\_otp=Button(viewcardwin,text=' Resend OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=198,y=130)

cr.execute(f'select phone from registration where card\_no={sc1}')

fetchcard\_phone=cr.fetchall()

for fpp in fetchcard\_phone:

cpn=fpp[0]

# print(cpn)

#generate otp.........

viewotp=str(random.randint(1111,9999))

# print('otp is: ',viewotp)

#

'''-------------------TESTING OTP-------------------------'''

'''------------------Twillo OTP Service------------------'''

# account\_sid = 'AC714959399352b0bebabaac962eb62449'

# auth\_token = '62b5f2426eb5ce5ac021ded99f10c6ab'

# client = Client(account\_sid, auth\_token)

# too='+91'+cpn

# msg='Online Voting System\nFor card verification\nyour OTP is: '+viewotp

# client.messages.create(body=msg,from\_='+19704576923',to=too)

'''-------------------------------------------------------'''

#TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

'''--------------------- Testing Otp ----------------------'''

otpnote\_view='Only for Testing Purpose OTP: '+viewotp

testing=Label(viewcardwin,text=otpnote\_view,font=('Regular',8),fg='red')

testing.place(x=5,y=0)

#update otp..............

cr.execute('update registration set otp=? where phone=?',(viewotp,cpn))

db.commit()

# print('changed/update otp')

view\_l=Button(viewcardwin,text=' ',fg='white',font=('Regular',15),bd=0,cursor='hand2',padx=10,width=70)

view\_l.place(x=100,y=400)

ue\_otp=StringVar()

def verify\_ue\_otp():

rs\_otp=ue\_otp.get()

cr.execute(f'select otp from registration where phone={cpn}')

f\_otpdata=cr.fetchall()

for dotpp in f\_otpdata:

ot=dotpp[0]

if ot==rs\_otp:

# print('otp match')

Verifyotpp=Button(viewcardwin,text='✓ Verified ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verify\_ue\_otp)

Verifyotpp.place(x=205,y=290)

def view():

#successfully login

cv=Toplevel()

cv.title('View Voter Card')

cv.geometry('550x680+485+25')

cv.wm\_iconbitmap('icon/ind.ico')

cv.resizable(False,False)

#successfully login

cr.execute(f'select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,card\_no,voting\_status,photo from registration where phone={cpn}')

data=cr.fetchall()

for i in data:

n=i[0]

d=i[1]

g=i[2]

f=i[3]

p=i[4]

aah=i[5]

vill=i[6]

po=i[7]

pin=i[8]

dist=i[9]

state=i[10]

cd=i[11]

vs=i[12]

myphoto=i[13]

'''---------------------- Profile Area----------------------'''

#bg thems..............

vc\_p=PhotoImage(file='themes/cardimg.png')

vc\_l=Label(cv,image=vc\_p)

vc\_l.place(x=70,y=120)

#////////////////////////////////

fp = io.BytesIO(myphoto)

# load the image

image = Image.open(fp)

res=image.resize((85,100))

# drawing image to top window

userimg = ImageTk.PhotoImage(res)

userphoto=Label(cv,image=userimg)

userphoto.place(x=125,y=185)

#\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

#------------------------------------

#card no

ucard\_no=Label(cv,text='Voter ID: ',font=('Regular',8),bg='#44FFDD')

ucard\_no.place(x=111,y=295)

ucard\_p=Label(cv,text=cd,font=('Regular',8),bg='#44FFDD')

ucard\_p.place(x=160,y=295)

#>>>>>>>>>>>>>>>>>>>>>>>

#name

uname=Label(cv,text='Name: ',font=('Regular',8),bg='#44FFDD')

uname.place(x=90,y=325)

uname\_p=Label(cv,text=n,font=('Regular',8),bg='#44FFDD')

uname\_p.place(x=170,y=325)

#father name

fname=Label(cv,text='Father Name: ',font=('Regular',8),bg='#44FFDD')

fname.place(x=90,y=349)

fname\_p=Label(cv,text=f,font=('Regular',8),bg='#44FFDD')

fname\_p.place(x=170,y=349)

#gender

ugender=Label(cv,text='Gender: ',font=('Regular',8),bg='#44FFDD')

ugender.place(x=90,y=373)

ugender\_p=Label(cv,text=g,font=('Regular',8),bg='#44FFDD')

ugender\_p.place(x=170,y=373)

# #date of birth

udob=Label(cv,text='Date Of Birth: ',font=('Regular',8),bg='#44FFDD')

udob.place(x=90,y=398)

udob\_p=Label(cv,text=d,font=('Regular',8),bg='#44FFDD')

udob\_p.place(x=170,y=398)

# #phone no

# uphone=Label(cv,text='Phone Number : ',font=('Regular',8),bg='#44FFDD')

# uphone.place(x=70,y=370)

# uphone\_p=Label(cv,text=p,font=('Regular',8),bg='#44FFDD')

# uphone\_p.place(x=170,y=370)

# #aadhaar no

# uaadhaar=Label(cv,text='Linked Aadhaar: ',font=('Regular',8),bg='#44FFDD')

# uaadhaar.place(x=70,y=400)

# uaadhaar\_p=Label(cv,text=aah,font=('Regular',8),bg='#44FFDD')

# uaadhaar\_p.place(x=170,y=400)

# uvstatus=Label(cv,text='Voting Status \_: ',font=('Regular',8),bg='#44FFDD')

# uvstatus.place(x=70,y=430)

# uvstatus\_p=Label(cv,text=vs,font=('Regular',8),bg='#44FFDD',fg='red')

# uvstatus\_p.place(x=170,y=430)

# #>>>>>>>>>>>>

#village

uvill=Label(cv,text='Vill: ',font=('Regular',8),bg='#44FFDD')

uvill.place(x=300,y=145)

uvill\_p=Label(cv,text=vill,font=('Regular',8),bg='#44FFDD')

uvill\_p.place(x=340,y=145)

#post

upost=Label(cv,text='Post: ',font=('Regular',8),bg='#44FFDD')

upost.place(x=300,y=165)

upost=Label(cv,text=po,font=('Regular',8),bg='#44FFDD')

upost.place(x=340,y=165)

#Pin

upin=Label(cv,text='Pin : ',font=('Regular',8),bg='#44FFDD')

upin.place(x=300,y=187)

upin=Label(cv,text=pin,font=('Regular',8),bg='#44FFDD')

upin.place(x=340,y=187)

#district

udist=Label(cv,text='Dist: ',font=('Regular',8),bg='#44FFDD')

udist.place(x=300,y=207)

udist=Label(cv,text=dist,font=('Regular',8),bg='#44FFDD')

udist.place(x=340,y=207)

#signature........

sign\_img=PhotoImage(file='img/sig.png')

sig\_l=Label(cv,image=sign\_img,bg='#44FFDD')

sig\_l.place(x=355,y=235)

cv.mainloop()

def createpdf():

# viewcardwin.title('View Voter Card')

# viewcardwin.geometry('550x680+485+25')

# viewcardwin.wm\_iconbitmap('icon/ind.ico')

# viewcardwin.resizable(False,False)

#successfully login

cr.execute(f'select name,dob,gender,father\_name,phone,aadhaar,village,post,pin,dist,state,card\_no,voting\_status from registration where phone={cpn}')

data=cr.fetchall()

for i in data:

n=i[0]

d=i[1]

g=i[2]

f=i[3]

p=i[4]

aah=i[5]

vill=i[6]

po=i[7]

pin=i[8]

dist=i[9]

state=i[10]

cd=i[11]

vs=i[12]

'''---------------------- Profile Area----------------------'''

#------------------------------------

#card no

cardnum='Voter ID: '

cd #card number

carduser\_name='Name: '

n #holder name

father='Father Name: '

f #father name

gender='Gender: '

g #gender

dob='Date Of Birth: '

d #dob..

phone='Phone Number : '

p #phone

village='Vill: '

vill

post='Post'

po

pn='Pin : '

pin

dst='Dist: '

dist

st='State: '

state

pdf=FPDF()

pdf.add\_page()

pdf.image('themes/cardimg.png',x=33,y=20)

pdf.image('img/sig.png',x=135,y=70)

pdf.set\_font("Arial",size=10) #set font size 10

pdf.set\_text\_color(0,0,0)

#>>>>>>>>>>>>>>>>>> font <<<<<<<<<<<<<<<<<<<<<<<

pdf.text(52,80,txt=cardnum)

pdf.text(67,80,txt=cd)

pdf.set\_font("Arial",size=10) #set font size 8

pdf.text(38,90,txt=carduser\_name)

pdf.text(64,90,txt=n)

pdf.text(38,98,txt=father)

pdf.text(64,98,txt=f)

pdf.text(38,106,txt=gender)

pdf.text(64,106,txt=g)

pdf.text(38,114,txt=dob)

pdf.text(64,114,txt=d)

pdf.text(38,122,txt=phone)

pdf.text(64,122,txt=p)

#>>>>>>>>>>>>>>>>>> Back <<<<<<<<<<<<<<<<<<<<<<<

pdf.text(114,32,txt=village)

pdf.text(125,32,txt=vill)

pdf.text(114,40,txt=post)

pdf.text(125,40,txt=po)

pdf.text(114,48,txt=pn)

pdf.text(125,48,txt=pin)

pdf.text(114,56,txt=dst)

pdf.text(125,56,txt=dist)

pdf.text(114,64,txt=st)

pdf.text(125,64,txt=state)

path=os.environ['USERPROFILE']

savefile=path+'\\'+'Desktop\\'+cd+'.pdf'

pdf.output(savefile)

showpath='Path: '+savefile

df\_d1=Label(viewcardwin,text=showpath,fg='red',font=('Regular',11),padx=20)

df\_d1.place(x=102,y=520)

df\_d=Label(viewcardwin,text='File Downloaded On Desktop',fg='white',font=('Regular',15),bg='red',padx=20)

df\_d.place(x=123,y=480)

down\_l=Button(viewcardwin,text=' Download ',fg='white',font=('Regular',15),bd=0,bg='green',cursor='hand2',padx=10,command=createpdf)

down\_l.place(x=120,y=400)

view\_l=Button(viewcardwin,text=' View Card ',fg='white',font=('Regular',15),bd=0,bg='green',cursor='hand2',padx=10,command=view)

view\_l.place(x=300,y=400)

#>>>>>>>>>>>>

else:

# print('Please Enter Valid OTP')

invalid\_OTP=Label(viewcardwin,text='Enter Valid OTP',font=('times new roman',8),fg='red',width=50)

invalid\_OTP.place(x=134,y=231)

#verify otp..........

otp\_t1=Label(viewcardwin,text='Enter OTP ',font=('Regular',18))

otp\_t1.place(x=110,y=200)

otp\_w1=Label(viewcardwin,text='OTP Send Your Mobile Number',font=('Regular',8))

otp\_w1.place(x=245,y=230)

otp\_ee=Entry(viewcardwin,font=('Regular',18),bg='#DFF6FF',width=13,textvar=ue\_otp)

otp\_ee.place(x=245,y=200)

Verifyotpp=Button(viewcardwin,text='Verify OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=verify\_ue\_otp)

Verifyotpp.place(x=205,y=290)

else:

#print('Phone number not match')

invalid\_card=Label(viewcardwin,text='Please Enter Voter ID',font=('times new roman',8),fg='red')

invalid\_card.place(x=250,y=90)

#label of phone number

card\_num=Label(viewcardwin,text='Enter Voter ID :',font=('times new roman',18,'bold'),fg='black',bd=0)

card\_num.place(x=50,y=55)

#enter phone number

card\_num\_e=Entry(viewcardwin,font=('Regular',24),width=14,bg='#D4F6CC',textvar=entcard)

card\_num\_e.place(x=250,y=50)

send\_card\_otp=Button(viewcardwin,text='✓ Send OTP ',fg='white',font=('Regular',15),bd=0,bg='#1BB3A5',cursor='hand2',padx=10,command=sendcardotp)

send\_card\_otp.place(x=198,y=130)

viewcardwin.mainloop()

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

#################################### User Panel / Area #######################################

#row ---------------> 1

#give vote

give\_img=PhotoImage(file='button/givevote.png')

give\_vote\_b=Button(root,image=give\_img,bd=0,bg='#C9C5C5',activebackground='#C9C5C5',cursor='hand2',command=givevote\_now)

give\_vote\_b.place(x=780,y=121)

#view card

view\_img=PhotoImage(file='button/viewcard.png')

view\_card\_b=Button(root,image=view\_img,bg='#C9C5C5',bd=0,activebackground='#C9C5C5',cursor='hand2',command=viewvoter\_card)

view\_card\_b.place(x=1032,y=121)

#track application

search\_img=PhotoImage(file='button/searchstatus.png')

search\_app\_status\_b=Button(root,image=search\_img,bd=0,bg='#C9C5C5',activebackground='#C9C5C5',cursor='hand2',command=trackapplication)

search\_app\_status\_b.place(x=1280,y=121)

#row ---------------> 2

#registration

new\_img=PhotoImage(file='button/regc.png')

new\_b=Button(root,image=new\_img,bd=0,bg='#C9C5C5',activebackground='#C9C5C5',cursor='hand2',command=RegistrationForm)

new\_b.place(x=780,y=310)

#download

download\_img=PhotoImage(file='button/download.png')

download\_b=Button(root,image=download\_img,bg='#C9C5C5',bd=0,activebackground='#C9C5C5',cursor='hand2',command=download)

download\_b.place(x=1032,y=310)

#update

update\_img=PhotoImage(file='button/update.png')

update\_card\_b=Button(root,image=update\_img,bg='#C9C5C5',bd=0,activebackground='#C9C5C5',cursor='hand2',command=required\_to\_login)

update\_card\_b.place(x=1280,y=310)

#row ---------------> 3

log\_img=PhotoImage(file='button/loginc.png')

log\_b=Button(root,image=log\_img,bd=0,bg='#C9C5C5',activebackground='#C9C5C5',cursor='hand2',command=LoginForm)

log\_b.place(x=780,y=499)

#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> left side <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

#row ---------------> 3

bjp\_img=PhotoImage(file='logo/bjp logo.png')

bjp\_b=Label(root,image=bjp\_img ,bg='#C9C5C5')

bjp\_b.place(x=20,y=121)

#count......

bjp1\_b=Label(root,bg='white',width=60,height=10)

bjp1\_b.place(x=250,y=123)

bjp\_l1=Label(root,text='Total Vote: ',font=('Regular',20),bg='white')

bjp\_l1.place(x=300,y=180)

#show count

bjp\_s=Label(root,text=bjp,font=('Regular',20),bg='white',fg='blue')

bjp\_s.place(x=440,y=180)

#tmc..........

tmc\_img=PhotoImage(file='logo/tmc logo.png')

tmc\_b=Label(root,image=tmc\_img ,bg='#C9C5C5')

tmc\_b.place(x=16,y=310)

#count........

tmc1\_b=Label(root,bg='white',width=60,height=10)

tmc1\_b.place(x=250,y=312)

tmc\_l1=Label(root,text='Total Vote: ',font=('Regular',20),bg='white')

tmc\_l1.place(x=300,y=370)

#show count

tmc\_s=Label(root,text=tmc,font=('Regular',20),bg='white',fg='blue')

tmc\_s.place(x=440,y=370)

#conj......

conj\_img=PhotoImage(file='logo/cong logo.png')

conj\_b=Label(root,image=conj\_img ,bg='#C9C5C5')

conj\_b.place(x=20,y=499)

#count.....

conj\_b=Label(root,bg='white',width=60,height=10)

conj\_b.place(x=250,y=501)

conj\_l1=Label(root,text='Total Vote: ',font=('Regular',20),bg='white')

conj\_l1.place(x=300,y=560)

#show count

conj\_s=Label(root,text=cong,font=('Regular',20),bg='white',fg='blue')

conj\_s.place(x=440,y=560)

root.mainloop()

# toaster = ToastNotifier()

# toaster.show\_toast("Online Voting Management System", "Thak you for using Our Application/Software...",icon\_path='icon/vote-sign.ico',duration=10)

**Only Registration Page Coding**

from msilib.schema import ComboBox

from operator import concat

from tkinter import \*

from tkinter import ttk

from tkinter import messagebox

from sqlalchemy import null

from tkinter import filedialog

import sqlite3

import random

try:

db=sqlite3.connect('voting management system.db')

cr=db.cursor()

cr.execute('create table registration(name text,dob text,gender text,father\_name text,phone text,email text,aadhaar text,village text,post text,pin text,dist text,state text,password text,conform\_password text,photo blob,ststus text,application\_no text,card\_no text,voting\_status text,otp text)')

db.commit()

except:

print('Database is running...')

regwin=Tk()

regwin.geometry('1367x700')

regwin.attributes('-fullscreen',True)

regwin.wm\_iconbitmap('icon/reg.ico')

regwin.title('Registration Form')

regwin.state('zoomed')

regwin.config(bg='#D9D9D9')

#import reg\_background image

reg\_back=PhotoImage(file='themes/registration form.png')

reg\_them=Label(regwin,bg='#D9D9D9',image=reg\_back,width=1050,height=780)

reg\_them.place(x=240,y=10)

#x-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x

name=StringVar()

dd=StringVar()

mm=StringVar()

yy=StringVar()

fathername=StringVar()

phone=StringVar()

aadhaar=StringVar()

password=StringVar()

conformpassword=StringVar()

email=StringVar()

village=StringVar()

post=StringVar()

pincode=StringVar()

dist=StringVar()

state=StringVar()

g=IntVar()

def cancel():

regwin.destroy()

import OnlineVotingManagementSystem

get\_image=0

#File dialog to select files

def filedialogs():

global get\_image

get\_image=0

get\_image = filedialog.askopenfilenames(title="SELECT IMAGE", filetypes=(("Allfile", "\*.\*"), ("png", "\*.png"), ("jpg" , "\*.jpg")))

#photo name...............

photoname=Label(regwin,text=get\_image,font=('Regular',10),fg='green',bg='white')

photoname.place(x=650,y=565)

#Image need to be conver into binary before insert into database

def conver\_image\_into\_binary(filename):

with open(filename, 'rb') as file:

photo\_image = file.read()

return photo\_image

def submit():

null = 0 #assign null value

name1=name.get()

dd1=dd.get()

mm1=mm.get()

yy1=yy.get()

fathername1=fathername.get()

phone1=phone.get()

aadhaar1=aadhaar.get()

password1=password.get()

password2=conformpassword.get()

email1=email.get()

village1=village.get()

post1=post.get()

pincode1=pincode.get()

dist1=dist.get()

state1=state.get()

gender=g.get()

# print(name1)

# print(dob)

# print(fathername1)

# print(phone1)

# print(aadhaar1)

# print(password1)

# print(password2)

# print(email1)

# print(village1)

# print(post1)

# print(pincode1)

# print(dist1)

# print(state1)

# print(g)

#name.................

length\_of\_name=len(name1)#set leangth 0

if null == length\_of\_name:

messagebox.showerror('Worning','Please Enter Your Name')

else:

# print(name1)

#date of birth..................

if dd1 == 'DD' or mm1 == 'MM' or yy1 == 'YY':

messagebox.showerror('Worning','Please Choose Correct Date of Birth')

else:

day=int(dd1)

month=int(mm1)

year=int(yy1)

if day <= 31 and month <= 12 and year <= 2050:

d1=str(day)

m1=str(month)

y1=str(year)

dob=d1 + '/'+ m1 +'/'+y1

# print(dob)

#father name..........................

length\_of\_father=len(fathername1)

if null == length\_of\_father:

messagebox.showerror('Worning','Please Enter Father Name')

else:

# print(fathername1)

#phone...........................

length\_of\_phone=len(phone1)

if null == length\_of\_phone:

messagebox.showerror('Worning','Please Enter Phone Number')

elif phone1.isdigit()==False:

messagebox.showwarning('Worning','Please Enter only Phone digit')

elif length\_of\_phone != 10:

messagebox.showwarning('Worning','Please Enter 10 digit Phone Number')

else:

# print(phone1)

#aadhaar card...................

alength\_of\_aadhaar=len(aadhaar1)

if null == alength\_of\_aadhaar:

messagebox.showerror('Worning','Please Enter aadhaar card number')

elif aadhaar1.isdigit()==False:

messagebox.showwarning('Worning','Please Enter only Aadhaar digit')

elif alength\_of\_aadhaar !=12:

messagebox.showwarning('Worning','Please Enter 12 digit Aadhaar Number')

else:

# print(aadhaar1)

#village.......................

length\_of\_village=len(village1)

if null == length\_of\_village:

messagebox.showerror('Worning','Please Enter village Name')

else:

# print(village1)

#post.........................

length\_of\_post=len(post1)

if null == length\_of\_post:

messagebox.showerror('Worning','Please Enter post office name')

else:

# print(post1)

#pin..........................

length\_of\_pin=len(pincode1)

if null == length\_of\_pin:

messagebox.showerror('Worning','Please Enter pincode')

elif pincode1.isdigit()==False:

messagebox.showwarning('Worning','Please Enter only pin digit ')

else:

# print(pincode1)

#district...............

length\_of\_dist=len(dist1)

if null == length\_of\_dist:

messagebox.showerror('Worning','Please Enter district Name')

else:

# print(dist1)

#stste........................

length\_of\_state=len(state1)

if null == length\_of\_state:

messagebox.showerror('Worning','Please Enter state Name')

else:

# print(state1)

#password..................

length\_of\_pass=len(password1)

if null == length\_of\_pass:

messagebox.showerror('Worning','Please Enter Password')

else:

# print(password1)

#password match............

if password1 != password2:

messagebox.showerror('Worning','Conform Password Not Match')

else:

#generate application no.................

app1=random.randint(10000000,99999999)

appid=str(app1)

notemessage='Application No: '+ appid+' \nPhone Number is Your Login Id...'

voterid='Not Generated'

votstatus='Not Voted'

#gender........................

if gender == null:

messagebox.showerror('Worning','Please Select Gender')

elif gender == 1:

gender='Male'

#photo...........

def insert\_data():

# image\_database = sqlite3.connect("Image\_data.db")

# data = image\_database.cursor()

if get\_image == 0:

messagebox.showerror('Worning','Please Select Image')

else:

cr.execute('select phone from registration')

data=cr.fetchall()

check=0

for i in data:

p1=i[0]

#a1=i[1] #for aadhaar card validation

if phone1==p1:

check=1

break

else:

check=0

if check == 1:

messagebox.showwarning('Exist User','Your Phone Number is Already Registered...')

else:

result=messagebox.askyesno('Confirmation','Are You Sure All Details is Correct?')

if result == True:

for image in get\_image:

insert\_photo = conver\_image\_into\_binary(image)

cr.execute('insert into registration(name,dob,gender,father\_name,phone,email,aadhaar,village,post,pin,dist,state,password,conform\_password,photo,application\_no,card\_no,voting\_status) values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)',(name1,dob,gender,fathername1,phone1,email1,aadhaar1,village1,post1,pincode1,dist1,state1,password1,password2,insert\_photo,appid,voterid,votstatus))

db.commit()

db.close()

messagebox.showinfo('','Application submit Successfull')

messagebox.showinfo('Please Note Down Your Application No',notemessage)

regwin.destroy()

import OnlineVotingManagementSystem

# data.execute("INSERT INTO Image Values(:image)", {'image': insert\_photo })

#print('success')

# image\_database.commit()

# image\_database.close()

insert\_data()

else:

gender='Female'

#photo...........

def insert\_image():

# image\_database = sqlite3.connect("Image\_data.db")

# data = image\_database.cursor()

if get\_image == 0:

messagebox.showerror('Worning','Please Select Image')

else:

cr.execute('select phone from registration')

data=cr.fetchall()

check=0

for i in data:

p1=i[0]

#a1=i[1] #for aadhaar card validation

if phone1==p1:

check=1

break

else:

check=0

if check == 1:

messagebox.showwarning('Exist User','Your Phone Number is Already Registered...')

else:

result=messagebox.askyesno('Confirmation','Are You Sure All Details is Correct?')

if result == True:

for image in get\_image:

insert\_photo = conver\_image\_into\_binary(image)

cr.execute('insert into registration(name,dob,gender,father\_name,phone,email,aadhaar,village,post,pin,dist,state,password,conform\_password,photo,application\_no,card\_no,voting\_status) values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)',(name1,dob,gender,fathername1,phone1,email1,aadhaar1,village1,post1,pincode1,dist1,state1,password1,password2,insert\_photo,appid,voterid,votstatus))

db.commit()

db.close()

messagebox.showinfo('','Application submit Successfull')

messagebox.showinfo('Please Note Down Your Application No',notemessage)

regwin.destroy()

import OnlineVotingManagementSystem

# data.execute("INSERT INTO Image Values(:image)", {'image': insert\_photo })

#print('success')

# image\_database.commit()

# image\_database.close()

insert\_image()

else:

messagebox.showwarning('Worning','Choose Correct Date of Birth')

#-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x-x--x-x-x-x-x-x-x-x-x-x

'''------------------------------------left side----------------------------------------'''

#import name image

name\_img=PhotoImage(file='button/name.png')

name\_l=Label(regwin,image=name\_img,bg='white')

name\_l.place(x=285,y=195)

#entry name

name\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=name)

name\_e.place(x=520,y=197)

#import dob image

dob\_img=PhotoImage(file='button/dob.png')

dob\_l=Label(regwin,image=dob\_img,bg='white')

dob\_l.place(x=285,y=250)

#entry date of birth

dob\_d=ttk.Combobox(regwin,font=('timew new roman',22),width=3,cursor='hand2',textvar=dd)

dob\_d['value']=['DD',1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27,28,29,30,31]

dob\_d.current(0)

dob\_d.place(x=520,y=252)

dob\_m=ttk.Combobox(regwin,font=('timew new roman',22),width=4,cursor='hand2',textvar=mm)

dob\_m['value']=['MM',1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

dob\_m.current(0)

dob\_m.place(x=598,y=252)

dob\_y=ttk.Combobox(regwin,font=('timew new roman',22),width=5,cursor='hand2',textvar=yy)

dob\_y['value']=['YY',1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023]

dob\_y.current(0)

dob\_y.place(x=690,y=252)

#importfather name image

father\_img=PhotoImage(file='button/fathers\_name.png')

fathername\_l=Label(regwin,image=father\_img,bg='white')

fathername\_l.place(x=285,y=300)

#entry father name

father\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=fathername)

father\_e.place(x=520,y=302)

#import phone image

phone\_img=PhotoImage(file='button/phone.png')

phone\_l=Label(regwin,image=phone\_img,bg='white')

phone\_l.place(x=285,y=350)

#entry phone

phone\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=phone)

phone\_e.place(x=520,y=352)

#import Aadhaar image

aadhaar\_img=PhotoImage(file='button/aadhaar.png')

aadhaar\_l=Label(regwin,image=aadhaar\_img,bg='white')

aadhaar\_l.place(x=285,y=400)

#entry aadhar

aadhaar\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=aadhaar)

aadhaar\_e.place(x=520,y=402)

#import password image

pass\_img=PhotoImage(file='button/pass.png')

password\_l=Label(regwin,image=pass\_img,bg='white')

password\_l.place(x=285,y=450)

#entry password

pass\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=password)

pass\_e.place(x=520,y=452)

#import conform password image

con\_img=PhotoImage(file='button/con\_pass.png')

con\_pass=Label(regwin,image=con\_img,bg='white')

con\_pass.place(x=285,y=500)

#entry conform password

conpass\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,show='\*',textvar=conformpassword)

conpass\_e.place(x=520,y=502)

#import upload image

upload\_img=PhotoImage(file='button/upload.png')

upload\_l=Label(regwin,image=upload\_img,bg='white')

upload\_l.place(x=285,y=555)

#import browse button image

browse\_img=PhotoImage(file='button/browse.png')

browse\_l=Button(regwin,image=browse\_img,bg='white',activebackground='white',bd=0,command=filedialogs)

browse\_l.place(x=520,y=557)

'''------------------------------------right side----------------------------------------'''

#import email image

email\_img=PhotoImage(file='button/email.png')

email\_l=Label(regwin,image=email\_img,bg='white')

email\_l.place(x=830,y=195)

#entry email

email\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=email)

email\_e.place(x=975,y=197)

#email optional..............

email\_op\_img=PhotoImage(file='button/E-mail Optional.png')

email\_op=Label(regwin,image=email\_op\_img,font=('Regular',8),fg='red',bg='white')

email\_op.place(x=974,y=235)

#import village image

vill\_img=PhotoImage(file='button/vill.png')

vill\_l=Label(regwin,image=vill\_img,bg='white')

vill\_l.place(x=830,y=250)

#entry village

vill\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=village)

vill\_e.place(x=975,y=252)

#import post image

post\_img=PhotoImage(file='button/post.png')

post\_l=Label(regwin,image=post\_img,bg='white')

post\_l.place(x=830,y=300)

#entry post

post\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=post)

post\_e.place(x=975,y=302)

#import pin code image

pin\_img=PhotoImage(file='button/pincode.png')

pin\_l=Label(regwin,image=pin\_img,bg='white')

pin\_l.place(x=830,y=350)

#entry pincode

pin\_e=Entry(regwin,font=('timew new roman',24),bd=1,bg='#EFEFE0',width=15,textvar=pincode)

pin\_e.place(x=975,y=352)

#import district image

dist\_img=PhotoImage(file='button/district.png')

dist\_l=Label(regwin,image=dist\_img,bg='white')

dist\_l.place(x=830,y=400)

#entry district

dist\_e=ttk.Combobox(regwin,font=('timew new roman',24),width=14,cursor='hand2',textvar=dist)

dist\_e['value']=["Alipurduar","Bankura","Birbhum","Cooch Behar","Darjeeling","Hooghly","Howra","Jalpaiguri","Jhargram","Kalimpong","Kolkata","Malda","Murshidabad","Nadia","North 24 Parganas","Purulia","Araria","Arwal","Aurangabad","Banka","Begusarai","Bhagalpur","Bhojpur","Buxar","Darbhanga","Gaya","Gopalganj","North Goa","South Goa",'Ahmedabad','Amreli','Anand','Aravalli','Bhavnagar','Dahod',"Jalandhar","Kapurthala","Ludhiana","Mansa","Moga","Muktsar","Pathankot",'Agra','Aligarh','Allahabad','Azamgarh','Baghpat','Bahraich','Ballia','Balrampur','Banda','Barabanki','Bokaro','Chatra','Deoghar','Dhanbad','Dumka','East', 'Singhbhum','Garhwa','Giridih','Srikakulam','Vizianagaram','Anakapalli','Kakinada','Eluru','NTR','Palnadu','Guntur']

dist\_e.current(12)

dist\_e.place(x=975,y=402)

#import state image

state\_img=PhotoImage(file='button/state.png')

state\_l=Label(regwin,image=state\_img,bg='white')

state\_l.place(x=830,y=450)

#entry state

state\_e=ttk.Combobox(regwin,font=('timew new roman',24),width=14,cursor='hand2',textvar=state)

state\_e['value']=["Andhra Pradesh", "Bihar" ,"Goa" ,"Gujarat" , "Jharkhand" ,"Punjab","Uttar Pradesh" ,"West Bengal"]

state\_e.current(7)

state\_e.place(x=975,y=452)

#import gender image

gender\_img=PhotoImage(file='button/gender.png')

gender\_l=Label(regwin,image=gender\_img,bg='white')

gender\_l.place(x=830,y=500)

#entry gender

gender1\_e=Radiobutton(regwin,text='Male',font=('times new roman',18),fg='blue',bg='white',activebackground='white',value=1,cursor='hand2',variable=g)

gender1\_e.place(x=990,y=502)

gender2\_e=Radiobutton(regwin,text='Female',font=('times new roman',18),fg='blue',bg='white',activebackground='white',value=2,cursor='hand2',variable=g)

gender2\_e.place(x=1080,y=502)

'''----------------------------------------- End ------------'---------------------------'''

'''-----------------------------------submit / cancel-------------------------------------'''

#import submit image

submit\_img=PhotoImage(file='button/u\_login.png')

submit=Button(regwin,image=submit\_img,bg='white',activebackground='white',bd=0,cursor='hand2',command=submit)

submit.place(x=595,y=700)

#import cancel image

cancel\_img=PhotoImage(file='button/cancel.png')

cancel=Button(regwin,image=cancel\_img,bg='white',activebackground='white',bd=0,cursor='hand2',command=cancel)

cancel.place(x=785,y=700)

regwin.mainloop()

***Thank You***