```
import pymysql
print(pymysql._file_)
pymysql.install_as_MySQLdb()
from django.apps import AppConfig
class DepressionappConfig(AppConfig):
 name = 'DepressionApp'
from django.db import models
# Create your models here.
from django.test import TestCase
# Create your tests here.
from django.urls import path
from . import views
urlpatterns = [path("index.html", views.index, name="index"),
 path("Register.html", views.Register, name="Register"),
 path("Signup", views.Signup, name="Signup"),
 path("Login.html", views.Login, name="Login"),
 path("UserLogin", views.UserLogin, name="UserLogin"),
 path("SearchFriends.html", views.SearchFriends, name="SearchFriends"),
 path("UploadPost.html", views.UploadPost, name="UploadPost"),
 path("UploadPostData", views.UploadPostData, name="UploadPostData"),
 path("AdminLogin", views.AdminLogin, name="AdminLogin"),
path("Admin.html", views.Admin, name="Admin"),
 path("ViewUsers.html", views.ViewUsers, name="ViewUsers"),
 path("ViewPosts.html", views.ViewPosts, name="ViewPosts"),
 path("SendMotivatedPost.html", views.SendMotivatedPost,
name="SendMotivatedPost"),
 path("SendMotivatedPostData", views.SendMotivatedPostData,
name="SendMotivatedPostData"),
 path("ViewMotivatedPost.html", views.ViewMotivatedPost,
name="ViewMotivatedPost"),
 path("MotivatedText.html", views.MotivatedText, name="MotivatedText"),
 ]
from django.shortcuts import render
from django.template import RequestContext
from django.contrib import messages
```

```
import pymysql
from django.http import HttpResponse
from django.conf import settings
from django.core.files.storage import FileSystemStorage
import datetime
from sklearn.externals import joblib
import PIL.Image
import pytesseract
import matplotlib.pyplot as plt
import re
import numpy as np
import speech_recognition as sr
# Create your views here.
svm_classifier = joblib.load('svmClassifier.pkl')
def index(request):
 if request.method == 'GET':
 return render(request, 'index.html', {})
def UploadPost(request):
 if request.method == 'GET':
return render(request, 'UploadPost.html', {})
def Register(request):
 if request.method == 'GET':
 return render(request, 'Register.html', {})
def Admin(request):
 if request.method == 'GET':
 return render(request, 'Admin.html', {})
def Login(request):
 if request.method == 'GET':
 return render(request, 'Login.html', {})
def SendMotivatedPost(request):
 if request.method == 'GET':
 return render(request, 'SendMotivatedPost.html', {})
def predict(textdata,classifier):
 text_processed = textdata
 X = [text_processed]
 sentiment = classifier.predict(X)
 return (sentiment[0])
def predictSentiment(textdata):
 result = predict(textdata, svm_classifier)
 predicts = ""
 if result == 0:
 predicts = "Negative"
 if result == 1:
 predicts = "Positive"
 return predicts
def SendMotivatedPostData(request):
 if request.method == 'POST':
username = request.POST.get('t1', False)
 time = request.POST.get('t2', False)
 text = request.POST.get('t3', False)
 db_connection = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root',
database = 'depression',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "update postdata set motivate_post='"+text+"' where
username='"+username+"' and post_time='"+time+"' and motivate_post='Pending'"
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 print(db_cursor.rowcount, "Record Inserted")
 context= {'data':'Your motivated text sent to user '+username}
 return render(request, 'SendMotivatedPost.html', context)


def UploadPostData(request):
 if request.method == 'POST' and request.FILES['t1']:
 output = ''
 myfile = request.FILES['t1']
 fs = FileSystemStorage()
 name = str(myfile)
 if name.lower().endswith(('.txt')):
 name = 'text.txt'
 elif name.lower().endswith(('.png', '.jpg', '.jpeg', 'gif')):
```

```
 name = 'img.jpg'
 filename = fs.save(name, myfile)
 if name.lower().endswith(('.txt')):
 with open("text.txt", "r") as file:
 for line in file:
 line = line.strip('\n')
 output+=line+' '
 elif name.lower().endswith(('.png', '.jpg', '.jpeg', 'gif')):
 output = pytesseract.image_to_string(PIL.Image.open(name))
 output = output.replace('\n',' ')
 elif name.lower().endswith(('.wav')):
 r = sr.Recognizer()
 with sr.WavFile(name) as source:
 audio = r.record(source)
try:
 output = r.recognize_google(audio)
 except:
 pass
 user = ''
 with open("session.txt", "r") as file:
 for line in file:
 user = line.strip('\n')
 now = datetime.datetime.now()
 option = 'Pending'
 output = re.sub('\W+',' ', output)
 current_time = now.strftime("%Y-%m-%d %H:%M:%S")
 sentiment = predictSentiment(output.lower())
 db_connection = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root',
database = 'depression',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "INSERT INTO
postdata(username,post_data,post_time,depression,motivate_post)
VALUES('"+user+"','"+output+"','"+current_time+"','"+sentiment+"','"+option+"')"
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 print(db_cursor.rowcount, "Record Inserted")
 if db_cursor.rowcount == 1:
 context= {'data':'Detected Depression From Uploaded File : '+sentiment}
 return render(request, 'UploadPost.html', context)
 else:
 context= {'data':'Error in signup process'}
 return render(request, 'UploadPost.html', context)

def ViewUsers(request):
 if request.method == 'GET':
 strdata = '<table border=1 align=center
width=100%><tr><th>Username</th><th>Password</th><th>Contact No</th><th>Email
ID</th><th>Address</th></tr><tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database =
'depression',charset='utf8')
 with con:
cur = con.cursor()
 cur.execute("select * FROM users")
 rows = cur.fetchall()
 for row in rows:

strdata+='<td>'+row[0]+'</td><td>'+row[1]+'</td><td>'+row[2]+'</td><td>'+str(row[3])+'</td><td
>'+str(row[4])+'</td></tr>'
 context= {'data':strdata}
 return render(request, 'ViewUsers.html', context)
def ViewPosts(request):
 if request.method == 'GET':
 positive = 0
 negative = 0
 strdata = '<table border=1 align=center width=100%><tr><th>Username</th><th>Post
Data</th><th>Post Time</th><th>Depression</th><th>Motivated Post</th></tr><tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database =
'depression',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select * FROM postdata")
 rows = cur.fetchall()
 for row in rows:
 if row[3] == 'Negative':
 negative = negative + 1
 else:
 positive = positive + 1

strdata+='<td>'+row[0]+'</td><td>'+row[1]+'</td><td>'+str(row[2])+'</td><td>'+str(row[3])+'</td
><td>'+str(row[4])+'</td></tr>'
 height = [positive,negative]
 bars = ('Depression Posts', 'Non Depression Post')
 y_pos = np.arange(len(bars))
 plt.bar(y_pos, height)
 plt.xticks(y_pos, bars)
 plt.show()
 context= {'data':strdata}
 return render(request, 'ViewPosts.html', context)
def MotivatedText(request):
 if request.method == 'GET':
 user = ''
 with open("session.txt", "r") as file:
 for line in file:
 user = line.strip('\n')
 strdata = '<table border=1 align=center width=100%><tr><th>Username</th><th>Post
Data</th><th>Post Time</th><th>Depression</th><th>Motivated Post</th></tr><tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database =
'depression',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select * FROM postdata")
 rows = cur.fetchall()
 for row in rows:
 if row[0] == user:
```

```
strdata+='<td>'+row[0]+'</td><td>'+row[1]+'</td><td>'+str(row[2])+'</td><td>'+str(row[3])+'</td><td>'+str(row[4])+'</td></tr>'
 context= {'data':strdata}
 return render(request, 'MotivatedText.html', context)
def ViewMotivatedPost(request):
 if request.method == 'GET':
 strdata = '<table border=1 align=center width=100%><tr><th>Username</th><th>Post Data</th><th>Post Time</th><th>Depression</th><th>Motivated Post</th></tr><tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database = 'depression',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select * FROM postdata")
 rows = cur.fetchall()
 for row in rows:
 if row[4] != 'Pending':

strdata+='<td>'+row[0]+'</td><td>'+row[1]+'</td><td>'+str(row[2])+'</td><td>'+str(row[3])+'</td><td>'+str(row[4])+'</td></tr>'
 context= {'data':strdata}
return render(request, 'ViewMotivatedPost.html', context)
def SearchFriends(request):
 if request.method == 'GET':
 user = ''
 with open("session.txt", "r") as file:
 for line in file:
 user = line.strip('\n')
 strdata = '<table border=1 align=center width=100%><tr><th>Username</th><th>Contact No</th><th>Email ID</th><th>Address</th></tr><tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database = 'depression',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select * FROM users")
 rows = cur.fetchall()
 for row in rows:
 if row[0] != user:

strdata+='<td>'+row[0]+'</td><td>'+row[2]+'</td><td>'+row[3]+'</td><td>'+str(row[4])+'</td></tr
>'
 context= {'data':strdata}
 return render(request, 'SearchFriends.html', context)
def UserLogin(request):
 if request.method == 'POST':
 username = request.POST.get('t1', False)
 password = request.POST.get('t2', False)
 index = 0
 con = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database =
'depression',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select * FROM users")
 rows = cur.fetchall()
 for row in rows:
 if row[0] == username and password == row[1]:
index = 1
 break
 if index == 1:
 file = open('session.txt','w')
 file.write(username)
 file.close()
 context= {'data':'welcome '+username}
 return render(request, 'UserScreen.html', context)
 else:
 context= {'data':'login failed'}
 return render(request, 'Login.html', context)
def Signup(request):
 if request.method == 'POST':
 username = request.POST.get('t1', False)
 password = request.POST.get('t2', False)
 contact = request.POST.get('t3', False)
 email = request.POST.get('t4', False)
 address = request.POST.get('t5', False)
 db_connection = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root',
database = 'depression',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "INSERT INTO users(username,password,contact_no,email,address)
VALUES('"+username+"','"+password+"','"+contact+"','"+email+"','"+address+"')"
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 print(db_cursor.rowcount, "Record Inserted")
 if db_cursor.rowcount == 1:
 context= {'data':'Signup Process Completed'}
 return render(request, 'Register.html', context)
 else:
 context= {'data':'Error in signup process'}
 return render(request, 'Register.html', context)
def AdminLogin(request):
 if request.method == 'POST':
username = request.POST.get('t1', False)
 password = request.POST.get('t2', False)
 if username == 'admin' and password == 'admin':
 context= {'data':'welcome '+username}
 return render(request, 'AdminScreen.html', context)
 else:
 context= {'data':'login failed'}
 return render(request, 'Admin.html', context)
```