ORGAN DONATION MANAGEMENT SYSTEM

SY COMP B(B1)

Ahana Agashe UCE2021502
Pradnya Apte UCE2021505
Akansha Dahikar UCE2021513
Arya Deshmukh UCE2021516

Code file

Database handling : MySQL
Backend : Python
Frontend : Streamlit (Python library) and Python
Connectivity : mysql.connector(Python library) and Python

```
import streamlit as st
from streamlit_option_menu import option_menu
from datetime import date
from datetime import datetime
from PIL import Image
import mysql.connector
import time
import pandas as pd
import random

#establishing the connection
conn = mysql.connector.connect(user='root', password='mysorezoo@2022',
host='localhost', database='DBMS_Project')

#Creating a cursor object using the cursor() method
cursor = conn.cursor()

# Title
st.title("🌎 ORGAN DONATION SYSTEM 🌎")


def authenticate(username,password):
    logins=[]
    try:
        query = ("select Passwrd from Login where Username="+"'%s'"%username)
        # Executing the SQL command
        cursor.execute(query)
        logins = cursor.fetchall()
    except Exception as e:
```

```
        print(e)

    if password == logins[0][0]:
        return True
    else:
        return False


selected = option_menu(
    menu_title=None,  # required
    options=["Home", "Admin", "Donor", "Receiver"],  # required
    icons=["house"],  # optional
    menu_icon="cast",  # optional
    default_index=0,  # optional
    orientation="horizontal",
    styles={
        "container": {"padding": "0!important", "background-color": "#fafafa"},
        "icon": {"color": "orange", "font-size": "20px"},
        "nav-link": {
            "font-size": "18px",
            "text-align": "middle",
            "margin": "5px",
            "--hover-color": "#eee",
        },
        "nav-link-selected": {"background-color": "red"},
    },
)

if selected == "Home":
    st.write()
    st.write()
    image =
Image.open('C://Users//Pradnya//PycharmProjects//DBMS_Miniproject//Organ-Donati
on.jpg')

    st.image(image)
    st.write("**We the creators of this Organ Donation System, believe in the
power of youth to make a difference. We are a team of dedicated  students from
Cummins College (CCOEW, Pune) have come together to create an innovative
platform that aims to save lives through organ donation. Our mission is to
increase awareness about organ donation, facilitate the donation process, and
connect donors with recipients in need.**")


elif (selected == "Admin"):

    # Initialize the session state variables
    if "is_logged_in" not in st.session_state:
        st.session_state.is_logged_in = False
```

```
    if "selectbox_visible" not in st.session_state:
        st.session_state.selectbox_visible = False
    if "logout_button" not in st.session_state:
        st.session_state.logout_button = False

    # If the user is not logged in, display the login form
    if not st.session_state.is_logged_in:
        # Add text inputs for the username and password
        st.header('Login Page')
        username = st.text_input("**Username**")
        password = st.text_input("**Password**", type="password")

        # Add a login button
        if st.button("Login"):
            # Check if the username and password are correct
            if authenticate(username, password):
                # Set the is_logged_in flag to True
                st.session_state.is_logged_in = True
                st.success("**Login Successful**")
                st.session_state.logout_button = True
                # Set the selectbox_visible flag to True
                st.session_state.selectbox_visible = True
            else:
                st.error("Incorrect username or password")
    elif st.session_state.is_logged_in:
        # If the selectbox is visible, display it
        if st.session_state.logout_button:

            col1, col2, col3, col4, col5 = st.columns(5)

            with col1:
                pass
            with col2:
                pass
            with col4:
                pass
            with col3:
                pass
            with col5:
                logout_button = st.button('Logout')
            if (logout_button):
                # Clear the session state variables and display a logout
message
                st.session_state.is_logged_in = False
                st.session_state.selectbox_visible = False
                st.info("You have been logged out.")

        if st.session_state.selectbox_visible:
            st.write("**Admin Functionalities**")
```

```
            admin_functions = st.selectbox("", options=["", "View details of
Donor or Receiver","View List of Pending Receivers", "Update Data","View
Expired Organs","View Successful Transactions"], key=21)
            print(admin_functions)

            if (admin_functions == "View details of Donor or Receiver"):
                st.write("")
                st.write("**CHOOSE**")
                choose = st.radio("", key="A", options=['Donor', 'Receiver'])

                if choose == "Donor":
                    id = st.text_input("**Enter Donor ID**")
                    if (st.button("View")):
                        labels = []
                        vals = []
                        try:
                            query = ("select * from Donor_Basic where
DonorID='%s'" % id)
                            cursor.execute(query)
                            result1 = cursor.fetchall()
                            # If DonorID exists in database
                             if (result1 != []):
                                labels = ['DonorID', 'Name', 'Date of Birth',
'Gender', 'Contact Number', 'Address',
                                          'Blood Group', 'Mortality Status']
                                for i in range(8):
                                    vals.append(result1[0][i])

                                query = ("select * from Donor_Kidney where
DonorID='%s'" % id)
                                cursor.execute(query)
                                resultKidney = cursor.fetchall()
                                if (resultKidney != []):
                                    labels.append('Tissue Type')
                                    labels.append('Sugar Level')
                                    vals.append(resultKidney[0][1])
                                    vals.append(resultKidney[0][2])

                                query = ("select * from Donor_Liver where
DonorID='%s'" % id)
                                cursor.execute(query)
                                resultLiver = cursor.fetchall()
                                if (resultLiver != []):
                                    labels.append('Blood Clotting Factor')
                                    vals.append(resultLiver[0][3])

                                query = ("select * from Donor_Pancreas where
DonorID='%s'" % id)
                                cursor.execute(query)
```

```
                                resultPancreas = cursor.fetchall()
                                if (resultPancreas != []):
                                    labels.append('BMI')
                                    vals.append(resultLiver[0][1])

                                query = ("select * from Donor_Cornea where
DonorID='%s'" % id)
                                cursor.execute(query)
                                resultCornea = cursor.fetchall()  # active
diseases parameter

                            else:
                                st.write("DonorID does not Exist")

                        except Exception as e:
                            print(e)

                        result = dict(zip(labels, vals))
                        st.dataframe(result)  # for displaying all details in
table format

                    else:
                        st.write("Donor ID does not Exist, Please enter a valid
Donor ID")



                elif choose == 'Receiver':
                    id = st.text_input("**Enter Receiver ID**")
                    if (st.button("View")):
                        labels = []
                        vals = []
                        try:
                            query = ("select * from Receiver_Basic where
ReceiverID='%s'" % id)
                            cursor.execute(query)
                            result1 = cursor.fetchall()
                            # If DonorID exists in database
                            if (result1 != []):
                                labels = ['DonorID', 'Name', 'Date of Birth',
'Gender', 'Contact Number',
                                          'Blood Group', 'Application date']
                                for i in range(7):
                                    vals.append(result1[0][i])

                                query = ("select * from Receiver_Kidney where
ReceiverID='%s'" % id)
                                cursor.execute(query)
                                resultKidney = cursor.fetchall()
```

```
                                if (resultKidney != []):
                                    labels.append('Tissue Type')
                                    labels.append('Sugar Level')
                                    vals.append(resultKidney[0][1])
                                    vals.append(resultKidney[0][2])

                                query = ("select * from Receiver_Liver where
ReceiverID='%s'" % id)
                                cursor.execute(query)
                                resultLiver = cursor.fetchall()
                                if (resultLiver != []):
                                    labels.append('Blood Clotting Factor')
                                    vals.append(resultLiver[0][3])

                                query = ("select * from Donor_Pancreas where
DonorID='%s'" % id)
                                cursor.execute(query)
                                resultPancreas = cursor.fetchall()
                                if (resultPancreas != []):
                                    labels.append('BMI')
                                    vals.append(resultLiver[0][1])

                                query = ("select * from  Receiver_Cornea where
ReceiverID='%s'" % id)
                                cursor.execute(query)
                                resultCornea = cursor.fetchall()  # active
diseases parameter

                            else:
                                st.write("ReceiverID does not exist")

                        except Exception as e:
                            print(e)

                        result = dict(zip(labels, vals))
                        st.dataframe(result)  # for displaying all details in
table format




            elif (admin_functions == "Update Data"):
                st.write("")
                st.write("**CHOOSE**")
                choose = st.radio("", key="A", options=['Donor', 'Receiver'])

                if (choose == 'Donor'):
                    don_id = st.text_input('**Enter Donor ID**')
```

```
                    r = st.radio("", key="B", options=["Update Basic
Information", "Update Organ Specific Information"])
                    if r == "Update Basic Information":
                        c = st.selectbox("", options=["", "Name", "Contact No",
"Mortality Status", "Address"])

                        if c == "Name":
                            new_val = st.text_input("Name")
                            if (st.button("Update")):
                                # update name
                                 query = ("Update Donor_Basic set DonorName='%s'
where DonorID='%s'" % (new_val, don_id))
                                cursor.execute(query)
                                conn.commit()
                                st.success("Name Updated Successfully")


                        elif c == "Contact No":
                            new_val = st.text_input("Contact No")
                            if (st.button("Update")):
                                # update the conatact no
                                 query = ("Update Donor_Basic set ContactNo=%s
where DonorID='%s'" % (new_val, don_id))
                                cursor.execute(query)
                                conn.commit()
                                st.success("Contact No Updated Successfully")


                        elif c == "Mortality Status":
                            new_val = st.text_input("Mortality Status")
                            if (st.button("Update")):
                                # update the status of mortality
                                 query = ("Update Donor_Basic set
MortalityStatus='%s' where DonorID='%s'" % (
                                new_val, don_id))
                                cursor.execute(query)
                                conn.commit()
                                st.success("Mortality Status Updated")


                        elif c == "Address":
                            new_val = st.text_input("Address")
                            if (st.button("Update")):
                                # update the status of mortality
                                 query = ("Update Donor_Basic set Address='%s'
where DonorID='%s'" % (new_val, don_id))
                                cursor.execute(query)
                                conn.commit()
                                st.success("Address Updated Successfully")
```

```
                elif r == "Update Organ Specific Information":
                    d = st.selectbox("", options=["", "BMI", "Sugar Level"])
                    if d == "BMI":
                        new_bmi = st.text_input("BMI")
                        if (st.button("Update")):
                            # update the status of mortality
                             query = ("Update Donor_Pancreas set BMI=%s where
DonorID='%s'" % (new_bmi, don_id))
                            cursor.execute(query)
                            conn.commit()
                            st.success("BMI Updated Successfully")

                    elif d == "Sugar Level":
                        new_sugar = st.text_input("Sugar Level")
                        if (st.button("Update")):
                            # update the sugar level
                             query = ("Update Donor_Liver set SugarLevel=%s
where DonorID='%s'" % (
                            new_sugar, don_id))
                            cursor.execute(query)
                            conn.commit()
                            query = ("Update Donor_Kidney set SugarLevel=%s
where DonorID='%s'" % (
                            new_sugar, don_id))
                            cursor.execute(query)
                            conn.commit()
                            st.success("Sugar Level Status Updated
Successfully")



            elif (choose == 'Receiver'):
                rec_id = st.text_input('**Enter Receiver ID**')
                r = st.radio("", key="B", options=["Update Basic
Information", "Update Organ Specific Information"])

                if r == "Update Basic Information":

                    c = st.selectbox("", options=["", "Name", "Contact No",
"Address"])

                    if c == "Name":
                        new_name = st.text_input("Updated Name")
                        if (st.button("Update")):
                            query1 = 'Update Receiver_Basic set Name= "%s"'
% new_name + 'where Receiver_ID = "%s"' % rec_id
```

```
                                cursor.execute(query1)
                                conn.commit()
                                st.success("Name Updated Successfully.")


                            elif c == "Contact No":
                                new_contnum = st.text_input("Updated Contact No")
                                if (st.button("Update")):
                                    query2 = 'Update Receiver_Basic set ContactNo=
"%s"' % new_contnum + 'where Receiver_ID = "%s"' % rec_id
                                cursor.execute(query2)
                                conn.commit()
                                st.success("Contact No Updated Successfully")


                            elif c == "Address":
                                new_add = st.text_input("Updated Address")
                                if (st.button("Update")):
                                    query3 = 'Update Receiver_Basic set Address =
"%s"' % new_add + 'where Receiver_ID = "%s"' % rec_id
                                cursor.execute(query3)
                                conn.commit()
                                st.success("Address Updated Successfully")


                        elif r == "Update Organ Specific Information":
                            d = st.selectbox("", options=["", "BMI", "Sugar Level"])
                            if d == "BMI":
                                new_bmi = st.text_input("Updated BMI")
                                if (st.button("Update")):
                                    query4 = 'Update Receiver_Pancreas set BMI=
"%s"' % new_bmi + 'where Receiver_ID = "%s"' % rec_id
                                cursor.execute(query4)
                                conn.commit()
                                st.success("BMI Status Updated Successfully.")

                            elif d == "Sugar Level":
                                new_sugar = st.text_input("Updated Sugar Level")
                                if (st.button("Update")):
                                    query5 = 'Update Receiver_Liver set Sugar_Level=
"%s"' % new_sugar + 'where Receiver_ID = "%s"' % rec_id
                                cursor.execute(query5)
                                conn.commit()
                                query6 = 'Update Receiver_Kidney set
Sugar_Level= "%s"' % new_sugar + 'where Receiver_ID = "%s"' % rec_id
                                cursor.execute(query6)
                                conn.commit()
                                st.success("Sugar Level Status Updated
Successfully.")
```

```
            elif (admin_functions == "View List of Pending Receivers"):
                query = 'Select * from Receiver_Basic'
                cursor.execute(query)
                pending = cursor.fetchall()
                df = pd.DataFrame(pending, columns=[desc[0] for desc in
cursor.description])
                # Display data as table
                st.table(df)

                st.write("**Match Pending Receiver**")
                old_recID = st.text_input("Receiver ID: ")

                if (st.button('Kidney')):
                    query1 = "Select Donor_Basic.DonorID from Donor_Basic inner
join Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % old_recID + " inner join Donor_Kidney on
Donor_Basic.DonorID = Donor_Kidney.DonorID inner join Receiver_Kidney on
Receiver_Basic.ReceiverID = Receiver_Kidney.ReceiverID where
(Donor_Kidney.TissueType = Receiver_Kidney.TissueType and
Donor_Kidney.SugarLevel < 200)"
                    cursor.execute(query1)
                    match_donor_id = cursor.fetchall()

                    if (len(match_donor_id) > 0):
                        st.subheader("MATCH FOUND")
                        st.write("You have been matched with DonorID " +
str(match_donor_id[0][0]))

                        querytemp = 'Select count(*) from
Successful_Transactions'
                        cursor.execute(querytemp)
                        count = cursor.fetchall()

                        query1 = 'Insert into Successful_Transactions
values(%s,%s,%s,%s,%s)'
                        values = ('OD2023' + str(count[0][0]), date.today(),
match_donor_id[0][0], old_recID,"SUCCESS")
                        cursor.execute(query1, values)

                        query2 = 'Delete from Donor_Kidney where DonorID = "%s"'
% match_donor_id[0][0]
                        cursor.execute(query2)
                        conn.commit()
```

```
                        query3 = "DELETE from Receiver_Kidney where ReceiverID =
'%s'" % old_recID
                        cursor.execute(query3)
                        conn.commit()
                        st.write("Database updated. ")
                        st.balloons()
                    else:
                        st.error("Currently no match found.")

                if (st.button('Liver')):
                    query = "Select Donor_Basic.DonorID from Donor_Basic inner
join Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % old_recID + " inner join Donor_Liver on
Donor_Basic.DonorID = Donor_Liver.DonorID inner join Receiver_Liver on
Receiver_Basic.ReceiverID = Receiver_Liver.ReceiverID where
(Donor_Liver.TissueType = Receiver_Liver.TissueType and Donor_Liver.BCF =
Receiver_Liver.BCF)"
                    cursor.execute(query)
                    match_donorID = cursor.fetchall()

                    if (len(match_donorID) > 0):
                        st.subheader("MATCH FOUND")
                        st.write("You have been matched with DonorID " +
str(match_donorID[0][0]))
                        querytemp = 'Select count(*) from
Successful_Transactions'
                        cursor.execute(querytemp)
                        count = cursor.fetchall()

                        query1 = 'Insert into Successful_Transactions
values(%s,%s,%s,%s,%s)'
                        values = ('OD2023' + str(count[0][0]), date.today(),
match_donorID[0][0], old_recID, "SUCCESS")
                        cursor.execute(query1, values)

                        query2 = 'Delete from Donor_Liver where DonorID = "%s"'
% match_donorID[0][0]
                        cursor.execute(query2)
                        conn.commit()

                        query3 = "DELETE from Receiver_Liver where ReceiverID =
'%s'" % old_recID
                        cursor.execute(query3)
                        conn.commit()
                        st.balloons()
                        st.write("Database updated. ")
                    else:
                        st.error("Currently no match found. Application
queued.")
```

```
            if (st.button('Pancreas')):
                query = 'Select Donor_Basic.DonorID from Donor_Basic inner
join Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup inner
join Donor_Pancreas on Donor_Basic.DonorID = Donor_Pancreas.DonorID where
Donor_Pancreas.BMI<30 and datediff(curdate(), Donor_Basic.DonationDate) < 36'
                cursor.execute(query)
                match_donorID = cursor.fetchall()

                if (len(match_donorID) > 0):
                    st.subheader("MATCH FOUND")
                    st.write("You have been matched with DonorID: " +
str(match_donorID[0][0]))
                    querytemp = 'Select count(*) from
Successful_Transactions'
                    cursor.execute(querytemp)
                    count = cursor.fetchall()

                    query1 = 'Insert into Successful_Transactions
values(%s,%s,%s,%s,%s)'
                    values = ('OD2023' + str(count[0][0]), date.today(),
match_donorID[0][0], old_recID, "SUCCESS")
                    cursor.execute(query1, values)

                    query2 = 'Delete from Donor_Pancreas where DonorID =
"%s"' % match_donorID[0][0]
                    cursor.execute(query2)
                    conn.commit()

                    query3 = "DELETE from Receiver_Pancreas where ReceiverID
= '%s'" % old_recID
                    cursor.execute(query3)
                    conn.commit()
                    st.write("Database updated. ")
                    st.balloons()
                else:
                    st.error("Currently no match found. Application
queued.")

            if (st.button('Cornea')):
                query = "Select Donor_Basic.DonorID from Donor_Basic inner
join Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % old_recID + " inner join Donor_Cornea on
Donor_Basic.DonorID = Donor_Cornea.DonorID inner join Receiver_Cornea on
Receiver_Basic.ReceiverID = Receiver_Cornea.ReceiverID"
                cursor.execute(query)
                match_donor_id = cursor.fetchall()
                if (len(match_donor_id) > 0):
                    st.subheader("MATCH FOUND")
```

```
                        st.write("You have been matched with DonorID: " +
str(match_donor_id[0][0]))
                        querytemp = 'Select count(*) from
Successful_Transactions'
                        cursor.execute(querytemp)
                        count = cursor.fetchall()

                        query1 = 'Insert into Successful_Transactions
values(%s,%s,%s,%s,%s)'
                        values = ('OD2023' + str(count[0][0]), date.today(),
match_donor_id[0][0], old_recID,"SUCCESS")
                        cursor.execute(query1, values)

                        query2 = 'Delete from Donor_Cornea where DonorID = "%s"'
% match_donor_id[0][0]
                        cursor.execute(query2)
                        conn.commit()

                        query3 = "DELETE from Receiver_Cornea where ReceiverID =
'%s'" % old_recID
                        cursor.execute(query3)
                        conn.commit()
                        st.write("Database updated. ")
                        st.balloons()
                    else:
                        st.error("Currently no match found.")



            elif (admin_functions == "View Expired Organs"):
                st.write("Expired Organs")
                donorId = []
                category = []
                kidney_list = []
                cornea_list = []
                liver_list = []
                pancreas_list = []

                try:
                    query = ("select Donor_Kidney.DonorID from Donor_Kidney
inner join Donor_Basic on Donor_Kidney.DonorID=Donor_Basic.DonorID and
datediff(curdate(),Donor_Basic.DonationDate)>= 7")
                    cursor.execute(query)
                    kidney = cursor.fetchall()
                    print(kidney)
                    for x in kidney:
                        donorId.append(x[0])
                        kidney_list.append((x[0]))
                        category.append("Kidney")
```

```
                    query = ("select Donor_Liver.DonorID from Donor_Liver inner
join Donor_Basic on Donor_Liver.DonorID=Donor_Basic.DonorID and
datediff(curdate(),Donor_Basic.DonationDate)>= 7")
                    cursor.execute(query)
                    liver = cursor.fetchall()
                    for x in liver:
                        donorId.append(x[0])
                        liver_list.append((x[0]))
                        category.append("Liver")

                    query = ("select Donor_Pancreas.DonorID from Donor_Pancreas
inner join Donor_Basic on Donor_Pancreas.DonorID=Donor_Basic.DonorID and
datediff(curdate(),Donor_Basic.DonationDate) >= 7")
                    cursor.execute(query)
                    pancreas = cursor.fetchall()
                    for x in pancreas:
                        donorId.append(x[0])
                        pancreas_list.append((x[0]))
                        category.append("Pancreas")

                    query = ("select Donor_Cornea.DonorID from Donor_Cornea
inner join Donor_Basic on Donor_Cornea.DonorID=Donor_Basic.DonorID and
datediff(curdate(),Donor_Basic.DonationDate)>= 7")
                    cursor.execute(query)
                    cornea = cursor.fetchall()
                    for x in cornea:
                        donorId.append(x[0])
                        cornea_list.append((x[0]))
                        category.append("Cornea")

                except Exception as e:
                    print(e)


                expired = dict(zip(donorId, category))
                st.dataframe(expired)


                values = tuple(donorId)


                if(st.button("DELETE")):
                    st.write("Records Deleted Successfully")
                if delete:
                    query = ("delete from Donor_Kidney where DonorID in" +
str(tuple(kidney_list)))
                    cursor.execute(query)
                    conn.commit()
```

```
                    query = ("delete from Donor_Liver where DonorID in" +
str(tuple(liver_list)))
                    cursor.execute(query)
                    conn.commit()
                    query = ("delete from Donor_Pancreas where DonorID in" +
str(tuple(pancreas_list)))
                    cursor.execute(query)
                    conn.commit()
                    query = ("delete from Donor_ where DonorID in" +
str(tuple(cornea_list)))
                    cursor.execute(query)
                    conn.commit()

                    st.write("Records Deleted Successfully")



            elif (admin_functions == "View Successful Transactions"):
                st.write("**Successful Transactions**")
                query = 'Select * from Successful_Transactions'
                cursor.execute(query)
                successful = cursor.fetchall()
                df = pd.DataFrame(successful, columns=[desc[0] for desc in
cursor.description])
                # Display data as table
                st.table(df)
                st.write("**Filter By**")
                col1, col2, col3, col4, col5 = st.columns(5)

                with col1:
                    pass

                with col2:
                    options1 = ["", "January", "February", "March", "April",
"May", "June", "July", "August","September", "October", "November", "December"]
                    month_option = st.selectbox('Month', options1)

                with col4:
                    options2 = ["", 2015, 2016, 2017, 2018, 2019, 2020, 2021,
2022, 2023]
                    year_option = st.selectbox('Year', options2)

                with col3:
                    pass

                if(options1 == "All"):
                    if(st.button(label="Apply")):
                        query = ("Select * from Successful_Transactions where
Year(Transaction_date) = %s " %year_option)
```

```
                        cursor.execute(query)
                        successful = cursor.fetchall()
                        df = pd.DataFrame(successful, columns=[desc[0] for desc
in cursor.description])
                        # Display data as table
                         st.table(df)

                else:
                    if (st.button(label="Apply")):
                        query = ("Select * from Successful_Transactions where
Year(Transaction_date) = '%s' and MonthName(Transaction_date) = '%s'" %
(year_option,month_option))
                        cursor.execute(query)
                        successful = cursor.fetchall()
                        df = pd.DataFrame(successful, columns=[desc[0] for desc
in cursor.description])
                        # Display data as table
                         st.table(df)



    else:
        st.error('Invalid username or password. Please try again.')



elif (selected == "Donor"):
    st.title("Register Donor")

    def validate_input(input_string):
        if len(input_string.strip()) == 0:
            return False
        return True

    @st.cache_data
    def generate_variable():
        today = date.today()
        year = today.year
        # To check if Donor ID exists in Table
        Donor_ID = ''
        query_stat = False
        while (query_stat == False):
            Donor_ID = ("D" + str(year) + str(random.randint(15, 99)))  #
generate Donor ID
            query = ("select * from Donor_Basic where DonorID='%s'" % Donor_ID)
            cursor.execute(query)
            if (cursor.fetchall() == []):
                query_stat = True
```

```
        return Donor_ID


    new_Donor_ID = generate_variable()


    my_form = st.form(key="form0")
    name = my_form.text_input(label="Name")
    start_date = datetime(1900, 1, 1)
    today_date = datetime.today()
    dob = my_form.date_input("Date of Birth", today_date, start_date)
    gender = my_form.radio("Gender", key="visibility", options=['', 'Male',
'Female', 'Other'])
    contact_no = my_form.text_input(label="Contact No")
    address = my_form.text_input(label="Address")
    blood_group = my_form.selectbox('Blood Group',
                                    ['', 'A+ve', 'B+ve', 'O+ve', 'AB+ve',
'A-ve', 'B-ve', 'O-ve', 'AB-ve'], key=1)
    mortality = my_form.selectbox('Mortality Status', ['', 'Not Deceased',
'Deceased'], key=2)


    if mortality == "'Deceased'":
        donationDate = datetime.today()
    else:
        donationDate = 'default'


    if my_form.form_submit_button(label="Submit"):
        if validate_input(name) and dob and gender and
validate_input(contact_no) and blood_group:
            # If DonorID not in table, insert values
            values = ("('%s','%s','%s','%s','%s','%s','%s','%s',%s)" % (
            new_Donor_ID, name, dob, gender, contact_no, address, blood_group,
mortality, donationDate))

            try:
                query = ("insert into Donor_Basic values" + values)
                # Executing the SQL command
                cursor.execute(query)
                conn.commit()
                st.success("Form Submitted Successfully")
                st.write("Your Donor ID is %s" % new_Donor_ID)

            except Exception as e:
                print(e)


        else:
```

```
            st.error("Please fill out all required fields.")

    # to verify match criteria
    Donor_Status = mortality

    # Organ Specific Forms
    st.subheader("Select Organ")
    organ = st.selectbox('Select Organ', ['', 'Kidney', 'Cornea', 'Liver',
'Pancreas'], label_visibility='hidden')

    if (organ == 'Kidney'):
        st.subheader("Enter Organ Specific attributes for kidney ")
        print("Inside Kidney form")
        f1 = st.form(key="form1")
        tissue_type = f1.selectbox('Tissue Type', ['', 'HLA-A', 'HLA-B',
'HLA-C', 'HLA-DP', 'HLA-DQ', 'HLA-DR'],label_visibility='hidden')
        sugar_level = str(f1.slider('Sugar Level', 50, 300, 600))
        values = ("('%s','%s','%s')" % (new_Donor_ID, tissue_type, sugar_level))
        print("After values submitted")
        if (f1.form_submit_button(label="Submit")):
            print("Inside Kidney Submit Button")
            if validate_input(tissue_type) and validate_input(sugar_level):
                print("Validated")
                try:
                    print(new_Donor_ID)
                    query = ("insert into Donor_Kidney values" + values)
                    # Executing the SQL command
                    cursor.execute(query)
                    conn.commit()
                    st.success("Form Submitted Successfully")

                except Exception as e:
                    print(e)

            else:
                st.error("Please fill out all required fields.")

        if Donor_Status == 'Deceased':

            if (st.button("Match")):
                query = "select * from Receiver_Kidney inner join Donor_Kidney
on Receiver_Kidney.TissueType=Donor_Kidney.TissueType and ReceiverID in (select
ReceiverID from Receiver_Basic inner join Donor_Basic on
Receiver_Basic.BloodGroup = Donor_Basic.BloodGroup and
Donor_Basic.DonorID='%s')" % new_Donor_ID
                cursor.execute(query)
                receiverid = cursor.fetchall()
                if (receiverid != None):
                    st.subheader("Match Found!")
```

```
                    st.write("Donor Matched with Receiver %s"%receiverid[0][0])
                    st.balloons()
                else:
                    st.error("Currently no match found")


    elif (organ == 'Cornea'):
        st.subheader("Enter Organ Specific attributes for Cornea ")
        f2 = st.form(key="form2")
        active_diseases = f2.text_input(label="Active diseases")
        values = ("('%s','%s')" % (new_Donor_ID, active_diseases))
        if (f2.form_submit_button(label="Submit")):
            if validate_input(active_diseases):
                try:
                    query = ("insert into Donor_Cornea values" + values)
                    # Executing the SQL command
                     cursor.execute(query)
                    conn.commit()
                    st.success("Form Submitted Successfully")
                except Exception as e:
                    print(e)

            else:
                st.error("Please fill out all required fields.")

        if Donor_Status == 'Deceased':
            if (st.button("Match")):
                query = "select ReceiverID from Receiver_Cornea inner join
Donor_Cornea on ReceiverID in (select ReceiverID from Receiver_Basic inner join
Donor_Basic on Receiver_Basic.BloodGroup = Donor_Basic.BloodGroup and
Donor_Basic.DonorID='%s')" % new_Donor_ID
                cursor.execute(query)
                receiverid = cursor.fetchall()
                if (receiverid != None):
                    st.subheader("Match Found!!")
                    st.write("Donor Matched with Receiver %s" %
receiverid[0][0])
                    st.balloons()
                else:
                    st.error("Currently no match found")



    elif (organ == 'Liver'):
        st.subheader("Enter Organ Specific attributes for Liver ")
        f3 = st.form(key="form3")
        tissue_type = f3.selectbox('Tissue Type', ['', 'HLA-A', 'HLA-B',
'HLA-C', 'HLA-DP', 'HLA-DQ', 'HLA-DR'],
                                   label_visibility='hidden')
```

```
        blood_clotting_factor = f3.text_input(label="Blood Clotting Factor")
        sugar_level = str(f3.slider('Sugar Level', 50, 300, 600))
        values = ("('%s','%s','%s','%s')" % (new_Donor_ID, sugar_level,
tissue_type, blood_clotting_factor))
        if (f3.form_submit_button(label="Submit")):
            if validate_input(tissue_type) and
validate_input(blood_clotting_factor) and validate_input(sugar_level):
                try:
                    query = ("insert into Donor_Liver values" + values)
                    # Executing the SQL command
                    cursor.execute(query)
                    conn.commit()
                    st.success("Form Submitted Successfully")
                except Exception as e:
                    print(e)

            else:
                st.error("Please fill out all required fields.")

        if Donor_Status == 'Deceased':
            if (st.button("Match")):
                query = "select ReceiverID from Receiver_Liver inner join
Donor_Liver on Receiver_Liver.TissueType=Donor_Liver.TissueType and
Receiver_Liver.BCF=Donor_Liver.BCF and
Receiver_Liver.SugarLevel<(Donor_Liver.SugarLevel + 10) and
Receiver_Liver.SugarLevel>(Donor_Liver.SugarLevel - 10) and ReceiverID in
(select ReceiverID from Receiver_Basic inner join Donor_Basic on
Receiver_Basic.BloodGroup = Donor_Basic.BloodGroup and
Donor_Basic.DonorID='%s')" % new_Donor_ID
                cursor.execute(query)
                receiverid = cursor.fetchall()
                if (receiverid != None):
                    st.subheader("Match Found!!")
                    st.write("Donor Matched with Receiver %s" %
receiverid[0][0])
                    st.balloons()
                else:
                    st.error("Currently no match found")



    elif (organ == 'Pancreas'):
        st.subheader("Enter Organ Specific attributes for Pancreas ")
        f4 = st.form(key="form4")
        bmi = str(f4.number_input(label="BMI"))
        values = ("('%s','%s')" % (new_Donor_ID, bmi))
        submit = f4.form_submit_button(label="Submit")
        if (submit):
            if validate_input(bmi):
```

```
                try:
                    query = ("insert into Donor_Pancreas values" + values)
                    # Executing the SQL command
                    cursor.execute(query)
                    conn.commit()
                    st.success("Form Submitted Successfully")
                except Exception as e:
                    print(e)
            else:
                st.error("Please fill out all required fields.")

        if Donor_Status == 'Deceased':
            if (st.button("Match")):
                query = "select ReceiverID from Receiver_Pancreas inner join
Donor_Pancreas on ReceiverID in (select ReceiverID from Receiver_Basic inner
join Donor_Basic on Receiver_Basic.BloodGroup = Donor_Basic.BloodGroup and
Donor_Basic.DonorID='%s')" % new_Donor_ID
                cursor.execute(query)
                receiverid = cursor.fetchall()
                if (receiverid != None):
                    st.subheader("Match Found!!")
                    st.write("Donor Matched with Receiver %s" %
receiverid[0][0])
                    st.balloons()
                else:
                    st.error("Currently no match found")


elif (selected == "Receiver"):
   st.title("Register Receiver")

   def validate_input(input_string):  # validating if entered string is null
        if len(input_string.strip()) == 0:
            return False
        return True


   @st.cache_data
   def generate_variable():
        today = date.today()
        year = today.year
        new_rec_ID = ''
        query_stat = False
        while (query_stat == False):
            new_rec_ID = ("R" + str(year) + str(random.randint(15, 99)))  #
generate Receiver ID
            query = ("select * from Receiver_Basic where ReceiverID='%s'" %
new_rec_ID)
            cursor.execute(query)
```

```
            if (cursor.fetchall() == []):
                query_stat = True
        return new_rec_ID



    rec_ID = generate_variable()

    my_form = st.form(key="form0")  # form0 for creating new reciever
application
    name = my_form.text_input(label="Name")
    start_date = datetime(1900, 1, 1)
    today_date = datetime.today()
    dob = my_form.date_input("Date of Birth", today_date, start_date)
    gender = my_form.radio("Gender", key="visibility", options=['Male',
'Female', 'Other'])
    contact_no = my_form.text_input(label="Contact No")
    address=my_form.text_input(label="Address")
    blood_group = my_form.selectbox('Blood Group', ['A+ve', 'B+ve', 'O+ve',
'AB+ve', 'A-ve', 'B-ve', 'O-ve', 'AB-ve'], key=1)


    if (my_form.form_submit_button(label="Submit")):  # button for submitting
new application into reciever_basic
        if validate_input(name)  and dob and gender and
validate_input(contact_no) and blood_group:
            query = "INSERT INTO Receiver_Basic VALUES (%s, %s, %s, %s, %s, %s,
%s,%s)"
            values = (rec_ID, name, dob, gender, contact_no,
address,blood_group, date.today())
            cursor.execute(query, values)
            conn.commit()
            st.success("Form Submitted Successfully.")
            st.write("Your Receiver ID is " + rec_ID)

        else:
            st.error("Please fill out all required fields.")

    st.subheader("Select Organ")  # selecting specific organ

    organ = st.selectbox('Select Organ', ['', 'Kidney', 'Cornea', 'Liver',
'Pancreas'], label_visibility='hidden')

    if (organ == 'Kidney'):  # entering organ specific attributes for kidney
        st.subheader("Enter Organ Specific attributes for kidney ")
        f1 = st.form(key="form1")
        tissue_type = f1.selectbox('**Tissue Type**', ['', 'HLA-A', 'HLA-B',
'HLA-C', 'HLA-DP', 'HLA-DQ', 'HLA-DR'],
                                   label_visibility='hidden')
```

```
        sugar_level = f1.slider('**Sugar Level**', 50, 120, 450)

        if (f1.form_submit_button(label="Submit")):
            if validate_input(tissue_type) and sugar_level:
                try:  # inserting into reciever_kidney
                     query = "Insert into Receiver_Kidney values(%s,%s,%s)"
                    values = (rec_ID, tissue_type, sugar_level)
                    cursor.execute(query, values)
                    conn.commit()
                    st.success("Application Submitted Successfully.")

                except Exception as e:
                    print(e)

            else:
                st.error("Please fill out all required fields.")

        if (st.button("Match")):
            print(rec_ID)
            query = "Select Donor_Basic.DonorID from Donor_Basic inner join
Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % rec_ID + " inner join Donor_Kidney on
Donor_Basic.DonorID = Donor_Kidney.DonorID inner join Receiver_Kidney on
Receiver_Basic.ReceiverID = Receiver_Kidney.ReceiverID where
(Donor_Kidney.TissueType = Receiver_Kidney.TissueType and
Donor_Kidney.SugarLevel < 200)"
            cursor.execute(query)
            match_donor_id = cursor.fetchall()

            if (len(match_donor_id) > 0):
                st.subheader("MATCH FOUND")
                st.write("You have been matched with DonorID " +
str(match_donor_id[0][0]))

                querytemp = 'Select count(*) from Successful_Transactions'
                cursor.execute(querytemp)
                count = cursor.fetchall()

                query1 = 'Insert into Successful_Transactions(Transaction_ID,
Transaction_date, DonorID, ReceiverID, Status) values(%s,%s,%s,%s,%s)'
                values = ('OD2023' + str(count[0][0]), date.today(),
match_donor_id[0][0],rec_ID, "SUCCESS")
                cursor.execute(query1, values)

                query2 = 'Delete from Donor_Kidney where DonorID = "%s"' %
match_donor_id[0][0]
                cursor.execute(query2)
                conn.commit()
```

```
                query3 = "DELETE from Receiver_Kidney where ReceiverID = '%s'" %
rec_ID
                cursor.execute(query3)
                conn.commit()
                st.write("Database updated. ")
                st.balloons()
            else:
                st.error("Currently no match found.")


    elif (organ == 'Cornea'):
        organ_rec_ID = rec_ID[:len(rec_ID) - 1] + str(int(rec_ID[len(rec_ID) -
1]) - 1)
        st.subheader("Enter Organ Specific attributes for Cornea ")
        f2 = st.form(key="form2")
        active_diseases = f2.text_input(label="Active diseases")
        if (f2.form_submit_button(label="Submit")):
            if validate_input(active_diseases):
                try:
                    query = 'insert into Receiver_Cornea(ReceiverID,
ActiveDiseases) values(%s, %s)'
                    values = (organ_rec_ID, active_diseases)
                    cursor.execute(query, values)
                    conn.commit()
                    st.success("Application Submitted Successfully.")

                except Exception as e:
                    print(e)
            else:
                st.error("Please fill out all required fields.")

        if (st.button("Match")):
            query = "Select Donor_Basic.DonorID from Donor_Basic inner join
Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % organ_rec_ID + " inner join Donor_Cornea on
Donor_Basic.DonorID = Donor_Cornea.DonorID inner join Receiver_Cornea on
Receiver_Basic.ReceiverID = Receiver_Cornea.ReceiverID"
            cursor.execute(query)
            match_donor_id = cursor.fetchall()
            if (len(match_donor_id) > 0):
                st.subheader("CONGRATULATIONS !!! ITS A MATCH")
                st.write("You have been matched with DonorID: " +
str(match_donor_id[0][0]))
                querytemp = 'Select count(*) from Successful_Transactions'
                cursor.execute(querytemp)
                count = cursor.fetchall()

                query1 = 'Insert into Successful_Transactions(Transaction_ID,
Transaction_date, Donor_ID, Receiver_ID, Status) values(%s,%s,%s,%s,%s)'
```

```
                values = ('OD2023' + str(count[0][0]), date.today(),
match_donor_id[0][0], organ_rec_ID, "SUCCESS")
                cursor.execute(query1, values)

                query2 = 'Delete from Donor_Cornea where DonorID = "%s"' %
match_donor_id[0][0]
                cursor.execute(query2)
                conn.commit()

                query3 = "DELETE from Receiver_Cornea where ReceiverID = '%s'" %
organ_rec_ID
                cursor.execute(query3)
                conn.commit()
                st.write("Database updated. ")
                st.balloons()
            else:
                st.error("Currently no match found.")



    elif (organ == 'Liver'):
        organ_rec_ID = rec_ID[:len(rec_ID) - 1] + str(int(rec_ID[len(rec_ID) -
1]) - 1)
        st.subheader("Enter Organ Specific attributes for Liver ")
        f3 = st.form(key="form3")
        tissue_type = f3.selectbox('Tissue Type', ['_', 'HLA-A', 'HLA-B',
'HLA-C', 'HLA-DP', 'HLA-DQ', 'HLA-DR'],
                                   label_visibility='hidden')
        blood_counting_factor = f3.selectbox('Blood Counting Factor',
                                             ['_', 'Factor 1', 'Factor 2',
'Factor 3', 'Factor 4'])
        sugar_level = f3.slider('Sugar Level', 50, 120, 450)

        if (f3.form_submit_button(label="Submit")):
            if validate_input(tissue_type) and blood_counting_factor and
sugar_level:
                try:
                    query = 'insert into Receiver_Liver values(%s,%s,%s,%s)'
                    values = (organ_rec_ID, sugar_level, tissue_type,
blood_counting_factor)
                    cursor.execute(query, values)
                    conn.commit()
                    st.success("Application Submitted Successfully.")

                except Exception as e:
                    print(e)

            else:
                st.error("Please fill out all required fields.")
```

```
        if (st.button(" Match")):
            query = "Select Donor_Basic.DonorID from Donor_Basic inner join
Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup and
Receiver_Basic.ReceiverID = '%s'" % organ_rec_ID + " inner join Donor_Liver on
Donor_Basic.DonorID = Donor_Liver.DonorID inner join Receiver_Liver on
Receiver_Basic.ReceiverID = Receiver_Liver.ReceiverID where
(Donor_Liver.TissueType = Receiver_Liver.TissueType and Donor_Liver.BCF =
Receiver_Liver.BCF)"
            cursor.execute(query)
            match_donorID = cursor.fetchall()

            if (len(match_donorID) > 0):
                st.subheader("CONGRATULATIONS !!! ITS A MATCH")
                st.write("You have been matched with DonorID " +
str(match_donorID[0][0]))
                querytemp = 'Select count(*) from Successful_Transactions'
                cursor.execute(querytemp)
                count = cursor.fetchall()

                query1 = 'Insert into Successful_Transactions(Transaction_ID,
Transaction_date, DonorID, ReceiverID, Status) values(%s,%s,%s,%s,%s)'
                values = ('OD2023' + str(count[0][0]), date.today(),
match_donorID[0][0], organ_rec_ID, "SUCCESS")
                cursor.execute(query1, values)

                query2 = 'Delete from Donor_Liver where DonorID = "%s"' %
match_donorID[0][0]
                cursor.execute(query2)
                conn.commit()

                query3 = "DELETE from Receiver_Liver where ReceiverID = '%s'" %
organ_rec_ID
                cursor.execute(query3)
                conn.commit()
                st.balloons()
                st.write("Database updated. ")
            else:
                st.error("Currently no match found. Application queued.")


    elif (organ == 'Pancreas'):
        organ_rec_ID = rec_ID[:len(rec_ID) - 1] + str(int(rec_ID[len(rec_ID) -
1]) - 1)
        st.subheader("Enter Organ Specific attributes for Pancreas ")
        f4 = st.form(key="form4")
        bmi = f4.number_input(label="BMI")
        if (f4.form_submit_button(label="Submit")):
            if bmi:
```

```
                try:
                    query = 'insert into Receiver_Pancreas(ReceiverID, BMI)
values(%s, %s)'
                    values = (organ_rec_ID, bmi)
                    cursor.execute(query, values)
                    conn.commit()
                    st.success("Application Submitted Successfully.")

                except Exception as e:
                    print(e)
            else:
                st.error("Please fill out all required fields.")

        if (st.button(" Match")):
            query = 'Select Donor_Basic.DonorID from Donor_Basic inner join
Receiver_Basic on Donor_Basic.BloodGroup = Receiver_Basic.BloodGroup inner join
Donor_Pancreas on Donor_Basic.DonorID = Donor_Pancreas.DonorID where
Donor_Pancreas.BMI<30 and datediff(curdate(), Donor_Basic.DonationDate) < 3 '
            cursor.execute(query)
            match_donorID = cursor.fetchall()

            if (len(match_donorID) > 0):
                st.subheader("MATCH FOUND")
                st.write("You have been matched with DonorID: " +
str(match_donorID[0][0]))
                querytemp = 'Select count(*) from Successful_Transactions'
                cursor.execute(querytemp)
                count = cursor.fetchall()

                query1 = 'Insert into Successful_Transactions(Transaction_ID,
Transaction_date, Donor_ID, Receiver_ID, Status) values(%s,%s,%s,%s,%s)'
                values = ('OD2023' + str(count[0][0]), date.today(),
match_donorID[0][0], organ_rec_ID, "SUCCESS")
                cursor.execute(query1, values)

                query2 = 'Delete from Donor_Pancreas where DonorID = "%s"' %
match_donorID[0][0]
                cursor.execute(query2)
                conn.commit()

                query3 = "DELETE from Receiver_Pancreas where ReceiverID = '%s'"
% organ_rec_ID
                cursor.execute(query3)
                conn.commit()
                st.write("Database updated. ")
                st.balloons()
            else:
                st.error("Currently no match found. Application queued.")
```