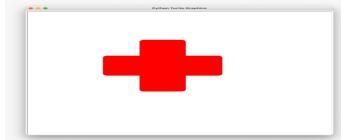




**REJUVENATE HOSPITALS**

# **HOSPITAL MANAGEMENT SYSTEM**



A Project Report Submitted for the  
AISSCE - All India Senior School  
Certificate Examination 2020-2021

In  
**COMPUTER SCIENCE (083)**

By:

- 1.Amizhthni.P.R.K.
- 2.Shashmitaa.P.
- 3.Sudiksha.M.V.
- 4.Shrinidhi.M.M.

## ACKNOWLEDGEMENT:

First and foremost, I would like to thank the almighty for having made my project "**Rejuvenate Hospitals - Hospital Management System**" a successful one. I hereby express my sincere gratitude towards my school "**Velammal Vidyalaya**" for having given me the wonderful opportunity to complete this project. I would also like to thank my benevolent teacher **Mrs. Sheeba. V** for having given me moral support and for being the backbone of this project. I also thank my group members **Sudiksha M. V, Shashmitaa P, Shrinidhi M. M.** for their help and support in the completion of this project. I would also like to extend my gratitude to my beloved parents for their never-ending support and encouragement.

# CERTIFICATE

This is to hereby certify that the **Computer Science** project had been solely, sincerely and satisfactorily completed by **Amizhthni.P.R.K.** of class XII for the practical examination of Central Board of Secondary Education (CBSE) in the academic year 2020-2021.

**Internal Examiner**

**External Examiner**

# CONTENTS

- HARDWARE AND SOFTWARE REQUIREMENTS
- PROGRAM DEFINITION
- PYTHON MODULES USED
- SOURCE CODE
- OUTPUT SCREENSHOTS
- TABLES IN MYSQL
- BIBLIOGRAPHY

## Hardware specifications of machine used:

- 1) Monitor : MacBookAir  
(Retina, 13-inch)
- 2) CPU : 1.6GHz Dual-core  
Intel Core i5
- 3) FDD : 1.44MB

## Software (OS, Language & Packages) used:

- 1) Operating System : Mac - OS BIG SUR
- 2) Language Version : Python 3.8  
Spyder 4.14

# PROGRAM DEFINITION

**TITLE:** REJUVENATE HOSPITALS-HOSPITAL MANAGEMENT SYSTEM

## OUR PROJECT:

Every hospital entails a large amount of data pertaining to its patients, doctors, pharmacy, blood bank, etc. This necessitates a structured, conducive, and user-friendly interface for managing this data and manipulating it in various ways to attain useful outputs. Especially in the time of a pandemic, it is instrumental that cases, deaths, and recoveries are monitored and duly analyzed to predict the far-reaching implications.

With this in mind, we started working on our project, which includes THREE main modules:

- 1.ADMIN MODULE
- 2.DOCTOR MODULE
- 3.COVID MODULE

1.The admin presides over the pharmacy, blood bank, appointments of doctors, patient data, and, regular updating. The admin is, also, in charge of maintaining wards and the software generates the bill for the patient at the time of discharge in no time.

2.The doctor views appointments, modifies them, and, deletes them at ease. They can also mail the voice-generated prescription to their patients.

3.The covid module gives the user with data of covid patients, their assigned doctors, personal information, health status, etc. It also gives the user numerous statistical data for interpretation.

The following **1849** lines of code make the above possible in a detailed manner.

## MODULES USED:

- 1.pyplot module from matplotlib library
- 2.turtle module
- 3.mysql.connector module
- 4.speech recognition module
- 5.numpy library-array module

## SOURCECODE:

```
from matplotlib import pyplot as pp
from turtle import *
import mysql.connector as ma
def connection():
    try:
        con=ma.connect(host="localhost",user="root",password="Jeeprep5*",
                       database="hospital_management_system")
        if con.is_connected()==False:
            print("database not connected")
        else:
            return con
    except ma.Error as er:
        print(er)
t=Turtle()
t.color("red","red")
t.begin_fill()
t.pensize(26)
t.speed(1000)
t.right(90)
t.forward(100)
t.left(90)
t.backward(100)
t.left(90)
t.forward(100)
t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.right(90)
t.forward(100)
t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.right(90)
t.forward(100)
```

```

t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.right(90)
t.forward(100)
t.end_fill()

def a_insert_ncpatients():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter patient's name: ")
        p_id=int(input('Enter patient id:'))
        date_in=input("Enter the date of joining the hospital:")
        date_out=input('enter the date of discharge')
        gender=input("Enter patient's gender' : ")
        age=int(input(" Enter patient's age: "))
        aadhar=int(input("Enter patient's aadhar number: "))
        insurance=int(input("Enter patient's health insurance number: "))
        phone=int(input("Enter patient's contact number ' : "))
        address=input("Enter patient's address ': ")
        disease=input("Enter patient's disease: ")
        room=input("Enter room type: " )
        room_number=int(input("Enter the room number : "))
        email=input("Enter email id of the patient : ")
        doc=int(input("Enter doctor id assigned : "))
        cur.execute("insert
                    into
                    patients(name,p_id,date_in,date_out,gender,age,aadhar,insurance,phon
e,address,disease,room,room_number,email,doc)values('%s',%d,'%s','%s
','%s',%d,%d,%d,%d,'%s','%s',%d,'%s',%d)"%
(name,p_id,date_in,date_out,gender,age,aadhar,insurance,phone,addres
s,disease,room,room_number,email,doc))
        print()
        print("Data inserted successfully")
        con.commit()
    except ma.Error as er:

```

```

print(er)
def a_display_ncpatients():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select * from patients")
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)

def a_update_ncpatients():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter patient's name: ")
        p_id=int(input('Enter patient id:'))
        date_in=input("Enter the date of joining the hospital:")
        date_out=input('enter date of discharge')
        gender=input("Enter patient's gender : ")
        age=int(input(" Enter patient's age: "))
        aadhar=int(input("Enter patient's aadhar number: "))
        insurance=int(input("Enter patient's health insurance number: "))
        phone=int(input("Enter patient's contact number ':"))
        address=input("Enter patient's address ':")
        disease=input("Enter patient's disease: ")
        room=input("Enter room type: ")
        room_number=int(input("Enter the room number :"))
        email=input("Enter email id of the patient :")
        doc=int(input("Enter doctor id assigned :"))
        cur.execute(" update patients set
name='%s',date_in='%s',date_out='%s',gender='%s',age=%d,aadhar=%d
,insurance=%d,phone=%d,address='%s',disease='%s',room='%s',room_
number=%d,email='%s',doc=%d
 where
p_id=%d"%(name,date_in,date_out,gender,age,aadhar,insurance,phone
,address,disease,room,room_number,email,doc,p_id))
        print()
    
```

```

        con.commit()
        print("Data updated successfully")
    except ma.Error as er:
        print(er)

def a_delete_ncpatients():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter patient's name to be deleted: ")
        cur.execute("delete from patients where name='%s'"%(name))
        print()
        con.commit()
        print("Data deleted successfully")
    except ma.error as er:
        print(er)
def d_insert_doctor():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter the doctor's name: ")
        d_id=int(input("Enter the doctor's id"))
        gend=input("Enter the gender of doctor")
        time=input("Enter the time of consultation: ")
        spec=input("Enter the specialization: ")
        exp=int(input("Enter the experience: "))
        fee=int(input("Enter the fee of consultation: "))
        email=input("Enter the doctor's email id: ")
        ph_no=int(input("Enter doctor's phone number: "))
        addr=input("Enter the doctor's address: ")

        cur.execute("insert into
                    doctors(name,d_id,gend,time,spec,exp,fee,email,ph_no,addr)values('%s'
                    ,%d,'%s','%s','%s',%d,%d,'%s',%d,'%s')"% (name,d_id,gend,time,spec,exp
                    ,fee,email,ph_no,addr))
        print()
        print("Date inserted successfully!")
    
```

```

        con.commit()
    except ma.Error as er:
        print(er)

def d_display_doctor():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select * from doctors")
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)

def d_update_doctor():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter the doctor's name: ")
        d_id=int(input("Enter the doctor's id"))
        gend=input("Enter the gender of doctor")
        time=input("Enter the time of consultation: ")
        spec=input("Enter the specialization: ")
        exp=int(input("Enter the experience: "))
        fee=int(input("Enter the fee for consultation"))
        email=input("Enter the doctor's email id: ")
        ph_no=int(input("Enter doctor's phone number: "))
        addr=input("Enter the doctor's address: ")

        cur.execute("Update doctors set
name='%s',gend='%s',time='%s',spec='%s',exp=%d,fee=%d,email='%s',p
h_no=%d,addr='%s'
where
d_id=%d"%(name,gend,time,spec,exp,fee,email,ph_no,addr,d_id))
        print()
        con.commit()
        print("Data updated successfully!")
    except ma.Error as er:

```

```

print(er)

def d_delete_doctor():
    try:
        con=connection()
        cur=con.cursor()
        name=input("Enter doctor's name to be deleted: ")
        cur.execute("Delete from doctors where name='%s'"%(name))
        print()
        con.commit()
        print("Data deleted successfully!")
    except ma.Error as er:
        print(er)

def a_insert_wards():
    try:
        con=connection()
        cur=con.cursor()
        building=input('enter building:')
        room_number=int(input("enter room no:"))
        number_of_beds=int(input("enter no. of beds:"))
        status=input("enter room status:")
        room_rent=int(input('enter daily charge for room:'))
        cur.execute("insert into
                    WARDS(building,room_number,number_of_beds,status,room_rent)
                    values('%s',%d,%d,'%s',%d)"%(building,room_number,number_of_beds,
                    status,room_rent))
        print("data inserted successfully")
        con.commit()
    except ma.Error as er:
        print(er)

def a_display_wards():
    try:
        con=connection()
        cur=con.cursor()

```

```

cur.execute("select * from WARDS")
for i in cur.fetchall():
    print(i)
except ma.Error as er:
    print(er)
def a_update_wards():
    try:
        con=connection()
        cur=con.cursor()
        building=input('enter building:')
        room_number=int(input("enter room no:"))
        number_of_beds=int(input("enter no. of beds:"))
        status=input("enter room status:")
        room_rent=int(input('enter daily charge for room:'))
        cur.execute("update WARDS set
building=%s,number_of_beds=%d,status=%s,room_rent=%d where
room_number=
%d"%(building,number_of_beds,status,room_rent,room_number))
        print()
        con.commit()
        print("records updated successfully")
    except ma.Error as er:
        print(er)

```

import numpy as np

```

def a_insert_covidpatients():

    try:
        con=connection()
        cur=con.cursor()
        print('INSERT COVIDPATIENTS')
        p=int(input('enter number of patients:'))
        for i in range(p):
            p_name=input("enter name of covid patients:")
            p_id=int(input("enter id for covid patients:"))
            p_age=int(input("enter age of covid patients:"))

```

```
p_gender=input("enter gender of covid patient;")
p_room=input("enter room number of covid patient in C264
TYPE:")
p_date=input("enter date of admission:")
p_status=input('Enter recovered/dead/ongoing')
p_doc=int(input("enter assigned doctor's id: "))
cur.execute("insert           into      COVID19_PATIENTS
(patient_id,age,gender,room,date,name,status_patient,p_doc)
values(%d,%d,'%s','%s','%s','%s',%d)"%(p_id,p_age,p_gender,p_woo
m,p_date,p_name,p_status,p_doc))
```

```
print("INSERTED")
con.commit()
except ma.Error as er:
    print(er)
```

```
def a_display_covidpatients():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select * from COVID19_PATIENTS")
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)
```

```
def a_update_covidpatients():
    try:
        con=connection()
        cur=con.cursor()
        p_name=input("enter name of covid patients:")
        p_id=int(input("enter id for covid patients:"))
        p_age=int(input("enter age of covid patients:"))
        p_gender=input("enter gender of covid patient:")
        p_room=input("enter room number of covid patient in C264 TYPE:")
        p_date=input("enter date of admission:")
        p_status=input('Enter recovered/dead/ongoing')
```

```

        cur.execute("update COVID19_PATIENTS set age=%d,gender='%s',
room='%s', date='%s', name='%s',status_patient='%s' where patient_id=
%d"%(p_age,p_gender,p_room,p_date,p_name,p_status,p_id))
        con.commit()
        print("RECORD UPDATION COMPLETED")
    except ma.Error as er:
        print(er)

def a_delete_covidpatients():
    try:
        con=connection()
        cur=con.cursor()
        pid=int(input("enter id of recovered/dead covid patients whose
records you want to delete:"))
        cur.execute("delete      from      COVID19_PATIENTS      where
patient_id=%d"%(pid))
        con.commit()
        print("RECORD DELETION COMPLETED or not found")
    except ma.Error as er:
        print(er)
def cov_init():
    try:
        con=connection()
        cur=con.cursor()
        #cur.execute("create      table      COVID_INIT(initiative
varchar(340),details varchar(1999))")
        #cur.execute("insert  into  COVID_INIT(initiative,details)  values
('CB-NAAT','REJUVENATE Hospitals takes COVID-19 testing to a whole
new level with CB-NAAT By embracing the Cartridge Based Nucleic
Acid Amplification Testing (CB-NAAT) method,REJUVENATE
Hospitals promises overall patient satisfaction, thanks to 100% accurate
test reports that are generated within just 2 hours')")
        #####cur.execute("insert      into
COVID_INIT(initiative,details)  values  ('REJUVENATE  Hospitals'
Separate Isolated Respiratory Block for Suspected COVID-19
Patients','REJUVENATE Hospitals' Separate Isolated Respiratory Block
is designed with dedicated pathways for patients & visitors to prevent

```

cross-contamination. With 300 beds, dedicated caregivers, separate entry, exit & kitchen, the block is battle-ready in the war against COVID-19.'")

```

initiative=input('you can add one initiative at a time.enter name of
initiative:')

details=input('enter details of initiative')
cur.execute("insert into COVID_INIT(initiative,details) values
('%s','%s')"% (initiative,details))
con.commit()

except ma.Error as er:
    print(er)

def cov_display_init():

try:
    con=connection()
    cur=con.cursor()
    cur.execute('select * from COVID_INIT')
    for i in cur.fetchall():
        print(i)
except ma.Error as er:
    print(er)

def daily_cc():

try:
    con=connection()
    cur=con.cursor()
    date=input('enter date:')
    status_patient='ongoing'
    cur.execute("SELECT COUNT(*) from COVID19_PATIENTS where
date='%s'and status_patient='%s'"%(date,status_patient))
    curry=cur.fetchone()
    print(curry[0])
except ma.Error as er:
    print(er)

def daily_cd():

try:
    con=connection()
    cur=con.cursor()
    date=input('enter date:')
    status_patient='dead'

```

```
    cur.execute("SELECT COUNT(*) from COVID19_PATIENTS where
date='%s'and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        print(curry[0])
    except ma.Error as er:
        print(er)
def daily_cr():
    try:
        con=connection()
        cur=con.cursor()
        date=input('enter date:')
        status_patient='recovered'
        cur.execute("SELECT COUNT(*) from COVID19_PATIENTS where
date='%s'and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        print(curry[0])
    except ma.Error as er:
        print(er)
def total_cc():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select count(*) from COVID19_PATIENTS")
        curry=cur.fetchone()
        print(curry[0])
    except ma.Error as er:
        print(er)
def total_cd():
    try:
        con=connection()
        cur=con.cursor()
        status_patient='dead'
        cur.execute("select count(*) from COVID19_PATIENTS where
status_patient='%s'"%(status_patient))
        curry=cur.fetchone()
        print(curry[0])
    except ma.Error as er:
```

```

    print(er)
def total_cr():
    try:
        con=connection()
        cur=con.cursor()
        status_patient='recovered'
        cur.execute("select count(*) from COVID19_PATIENTS where
status_patient=%s"%(status_patient))
        curry=cur.fetchone()
        print(curry[0])
    except ma.Error as er:
        print(er)

def line_chart_all_circle():
    try:
        con=connection()
        cur=con.cursor()
        ami=int(input("enter no. of dates' data' to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
        x=datu
        y=[ ]
        for j in range(len(datu)):

            date=datu[j]
            cur.execute("select count(*) from COVID19_PATIENTS where
date=%s"%(date))
            curry=cur.fetchone()
            y.append(curry[0])
        z=[ ]
        for j in range(len(datu)):
            status_patient='dead'

            date=datu[j]

```

```

        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s'and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        z.append(curry[0])
r=[ ]
for j in range(len(datu)):
    status_patient='recovered'
    date=datu[j]
    cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
    curry=cur.fetchone()
    r.append(curry[0])
pp.plot(x,y,'bo')
pp.plot(x,z,'ro')
pp.plot(x,r,'go')
pp.title('covid cases,deaths,recoveries')
pp.xlabel('DATES')
pp.ylabel('CASES, DEATHS,RECOVERIES')
pp.legend(['this is cases','this is DEATHS','this is RECOVERIES'])
pp.show()
except ma.Error as er:
    print(er)
def line_chart_all():
try:
    con=connection()
    cur=con.cursor()
    ami=int(input("enter no. of dates' data' to be compiled"))
    datu=[ ]
    for i in range(ami):
        kj=input('enter date')
        datu.append(kj)

    y=[ ]
    for j in range(len(datu)):
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s'"%(date))

```

```

        curry=cur.fetchone()
        y.append(curry[0])
    z=[ ]
    for j in range(len(datu)):
        status_patient='dead'
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        z.append(curry[0])
    r=[ ]
    for j in range(len(datu)):
        status_patient='recovered'
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        r.append(curry[0])
    x=datu
    x=np.array(x)
    y=np.array(y)
    z=np.array(z)
    r=np.array(r)
    pp.figure(figsize=(13, 3))
    pp.subplot(222)
    pp.xlabel('dates')
    pp.ylabel('cases')
    pp.plot(x,y) #line chart
    pp.suptitle('COVID cases IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')

    pp.figure(figsize=(13, 3))
    pp.subplot(222)
    pp.xlabel('dates')
    pp.ylabel('deaths')
    pp.plot(x,z) #line chart
    pp.suptitle('COVID DEATHS IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')

```

```

pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('recoveries')
pp.plot(x,r)    #line chart
pp.suptitle('COVID recoveries IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.show()
except ma.Error as er:
    print(er)
def scatter_plot_all():
try:
    con=connection()
    cur=con.cursor()
    ami=int(input("enter no. of dates' data' to be compiled"))
    datu=[ ]
    for i in range(ami):
        kj=input('enter date')
        datu.append(kj)
    x=datu
    y=[ ]
    for j in range(len(datu)):
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s'"%(date))
        curry=cur.fetchone()
        y.append(curry[0])
    z=[ ]
    for j in range(len(datu)):
        status_patient='dead'

        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        z.append(curry[0])
    r=[ ]

```

```

for j in range(len(datu)):
    status_patient='recovered'
    date=datu[j]
    cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
    curry=cur.fetchone()
    r.append(curry[0])
x=np.array(x)
y=np.array(y)
z=np.array(z)
r=np.array(r)
pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('cases')
pp.scatter(x,y) #line chart
pp.suptitle('COVID cases IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('deaths')
pp.scatter(x,z) #line chart
pp.suptitle('COVID DEATHS IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('recoveries')
pp.scatter(x,r) #line chart
pp.suptitle('COVID recoveries IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.show()
except ma.Error as er:
    print(er)

```

```
def bar_chart_all():
```

```

try:
    con=connection()
    cur=con.cursor()
    ami=int(input("enter no. of dates' data' to be compiled"))
    datu=[ ]
    for i in range(ami):
        kj=input('enter date')
        datu.append(kj)
    x=datu
    y=[ ]
    for j in range(len(datu)):
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s'"%(date))
        curry=cur.fetchone()
        y.append(curry[0])
    z=[ ]
    for j in range(len(datu)):
        status_patient='dead'
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        z.append(curry[0])
    r=[ ]
    for j in range(len(datu)):
        status_patient='recovered'
        date=datu[j]
        cur.execute("select count(*) from COVID19_PATIENTS where
date='%s' and status_patient='%s'"%(date,status_patient))
        curry=cur.fetchone()
        r.append(curry[0])
    x=np.array(x)
    y=np.array(y)
    z=np.array(z)
    r=np.array(r)
    pp.figure(figsize=(13, 3))

```

```
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('cases')
pp.bar(x,y)    #line chart
pp.suptitle('COVID cases IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('deaths')
pp.bar(x,z)    #line chart
pp.suptitle('COVID DEATHS IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
pp.figure(figsize=(13, 3))
pp.subplot(222)
pp.xlabel('dates')
pp.ylabel('recoveries')
pp.bar(x,r)    #line chart
pp.suptitle('COVID recoveries IN REJUVENATE HOSPITAL -ON A
DAILY BASIS')
except ma.Error as er:
    print(er)
```

```
def analyse_covidata():
    try:
        print()
        print("HOW'D YOU LIKE TO VIEW GRAPHICAL COVID DATA
OF REJUVENATE HOSPITAL")
        print('1.d,c,r line chart \n2.d,c,r line chart with circular denotation for
all three in one graph \n3.d,c,r bar graph\n4.d,c,r scatter plot\n')
        chu=int(input('enter choice:'))

        if chu==1:
            print()
            line_chart_all()
        elif chu==2:
            line_chart_all_circle()
```

```

        elif chu==3:
            print()
            bar_chart_all()
        elif chu==4:
            scatter_plot_all()
    except ma.Error as er:
        print(er)

def avg_case():
    try:
        con=connection()
        cur=con.cursor()
        #     cur.execute("create      table      date_wise_covidcase(date_of
varchar,cases int(34))")
        ami=int(input("enter no. of dates' data to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
        valu=[ ]
        for i in datu:
            date=i
            cur.execute("SELECT  COUNT(*)  from  COVID19_PATIENTS
where date='%s'"%(date))
            curry=cur.fetchone()
            valu.append(curry[0])
        for m in range(len(datu)):
            date_of=datu[m]
            cases=valu[m]
            cur.execute("insert      into      date_wise_covidcase(date_of,cases)
values ('%s','%d")% (date_of,cases)")
            con.commit()
            cur.execute('select avg(DISTINCT cases) from date_wise_covidcase')
            curry=cur.fetchone()
            print(curry[0])
    except ma.Error as er:

```

```

print(er)

def avg_death():
    try:
        con=connection()
        cur=con.cursor()
        #   cur.execute("create      table      date_wise_coviddeaths(date_of
varchar,deaths int(34))")
        ami=int(input("enter no. of dates' data to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
        valu=[ ]
        for i in datu:
            date=i
            status_patient='dead'
            cur.execute("SELECT  COUNT(*)  from  COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))
            curry=cur.fetchone()
            valu.append(curry[0])
        for m in range(len(datu)):
            date_of=datu[m]
            deaths=valu[m]
            cur.execute("insert  into  date_wise_coviddeaths(date_of,deaths)
values ('%s','%d")%(date_of,deaths))
            con.commit()
            cur.execute('select      avg(DISTINCT      deaths)      from
date_wise_coviddeaths')
            curry=cur.fetchone()
            print(curry[0])
    except ma.Error as er:
        print(er)

```

```

def avg_recoveries():
    try:

```

```

con=connection()
cur=con.cursor()
# cur.execute("create table date_wise_covidrecoveries(date_of
varchar,recoveries int(34))")
ami=int(input("enter no. of dates' data to be compiled"))
datu=[ ]
for i in range(ami):
    kj=input('enter date')
    datu.append(kj)
valu=[ ]
for i in datu:
    date=i
    status_patient='recovered'
    cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))
    curry=cur.fetchone()
    valu.append(curry[0])
for m in range(len(datu)):
    date_of=datu[m]
    recoveries=valu[m]
    cur.execute("insert into
date_wise_covidrecoveries(date_of,recoveries) values
('%s','%d")%(date_of,recoveries)")
    con.commit()
    cur.execute('select      avg(DISTINCT      recoveries)      from
date_wise_covidrecoveries')
    curry=cur.fetchone()
    print(curry[0])
except ma.Error as er:
    print(er)

def most_cases():
try:
    con=connection()
    cur=con.cursor()
    # cur.execute("create table date_wise_covidcase(date_of
varchar,cases int(34))")

```

```

ami=int(input("enter no. of dates' data to be compiled"))
datu=[ ]
for i in range(ami):
    kj=input('enter date')
    datu.append(kj)
valu=[ ]
for i in datu:
    date=i
    cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'"%(date))
    curry=cur.fetchone()
    valu.append(curry[0])
for m in range(len(datu)):
    date_of=datu[m]
    cases=valu[m]
    cur.execute("insert into date_wise_covidcase(date_of,cases)
values ('%s','%d")%(date_of,cases))")
    con.commit()
    cur.execute('select max(DISTINCT cases) from
date_wise_covidcase')
    curry=cur.fetchone()
    print(curry[0])
except ma.Error as er:
    print(er)

```

```

def most_deaths():
try:
    con=connection()
    cur=con.cursor()
    # cur.execute("create table date_wise_coviddeaths(date_of
varchar,deaths int(34))")
    ami=int(input("enter no. of dates' data to be compiled"))
    datu=[ ]
    for i in range(ami):
        kj=input('enter date')
        datu.append(kj)
    valu=[ ]

```

```

for i in datu:
    date=i
    status_patient='dead'
    cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))
    curry=cur.fetchone()
    valu.append(curry[0])
for m in range(len(datu)):
    date_of=datu[m]
    deaths=valu[m]
    cur.execute("insert into date_wise_coviddeaths(date_of,deaths)
values ('%s','%d")%(date_of,deaths)")
    con.commit()
    cur.execute('select      max(DISTINCT      deaths)      from
date_wise_coviddeaths')
    curry=cur.fetchone()
    print(curry[0])
except ma.Error as er:
    print(er)

```

```

def most_recoveries():
try:
    con=connection()
    cur=con.cursor()
    # cur.execute("create table date_wise_covidrecoveries(date_of
varchar,recoveries int(34))")
    ami=int(input("enter no. of dates' data to be compiled"))
    datu=[ ]
    for i in range(ami):
        kj=input('enter date')
        datu.append(kj)
    valu=[ ]
    for i in datu:
        date=i
        status_patient='recovered'
        cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))

```

```

        curry=cur.fetchone()
        valu.append(curry[0])
        for m in range(len(datu)):
            date_of=datu[m]
            recoveries=valu[m]
            cur.execute("insert into
date_wise_covidrecoveries(date_of,recoveries)
values
('%" + str(date_of) + "','" + str(recoveries) + "')
            con.commit()
            cur.execute('select      max(DISTINCT      recoveries)      from
date_wise_covidrecoveries')
            curry=cur.fetchone()
            print(curry[0])
        except ma.Error as er:
            print(er)

```

```

def least_cases():
    try:
        con=connection()
        cur=con.cursor()
        #     cur.execute("create      table      date_wise_covidcase(date_of
varchar,cases int(34))")
        ami=int(input("enter no. of dates' data to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
            valu=[ ]
            for i in datu:
                date=i
                status_patient='ongoing'
                cur.execute("SELECT  COUNT(*)  from  COVID19_PATIENTS
where date='%" + str(date) + "' and status_patient='%" + str(status_patient) + "'")
                curry=cur.fetchone()
                valu.append(curry[0])
            for m in range(len(datu)):

```

```

date_of=datu[m]
cases=valu[m]
cur.execute("insert into date_wise_covidcase(date_of,cases)
values ('%s','%d")%(date_of,cases))
con.commit()
cur.execute('select min(DISTINCT cases) from
date_wise_covidcase')
curry=cur.fetchone()
print(curry[0])
except ma.Error as er:
    print(er)

def least_deaths():
    try:
        con=connection()
        cur=con.cursor()
        # cur.execute("create table date_wise_coviddeaths(date_of
varchar,deaths int(34))")
        ami=int(input("enter no. of dates' data to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
        valu=[ ]
        for i in datu:
            date=i
            status_patient='dead'
            cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))
            curry=cur.fetchone()
            valu.append(curry[0])
        for m in range(len(datu)):
            date_of=datu[m]
            deaths=valu[m]
            cur.execute("insert into date_wise_coviddeaths(date_of,deaths)
values ('%s','%d")%(date_of,deaths))
            con.commit()

```

```

        cur.execute('select      min(DISTINCT      deaths)      from
date_wise_coviddeaths')
        curry=cur.fetchone()
        print(curry[0])
except ma.Error as er:
    print(er)

def least_recoveries():
    try:
        con=connection()
        cur=con.cursor()
        # cur.execute("create table date_wise_covidrecoveries(date_of
varchar,recoveries int(34))")
        ami=int(input("enter no. of dates' data to be compiled"))
        datu=[ ]
        for i in range(ami):
            kj=input('enter date')
            datu.append(kj)
        valu=[ ]
        for i in datu:
            date=i
            status_patient='recovered'
            cur.execute("SELECT COUNT(*) from COVID19_PATIENTS
where date='%s'and status_patient='%s'"%(date,status_patient))
            curry=cur.fetchone()
            valu.append(curry[0])
        for m in range(len(datu)):
            date_of=datu[m]
            recoveries=valu[m]
            cur.execute("insert
date_wise_covidrecoveries(date_of,recoveries)
('%s',%d)"%(date_of,recoveries))
            con.commit()
        cur.execute('select      min(DISTINCT      recoveries)      from
date_wise_covidrecoveries')
        curry=cur.fetchone()

```

```
    print(curry[0])
except ma.Error as er:
    print(er)
print('welcome user! COVID SPECIALITY UNIT\n JOIN US IN OUR
BATTLE AGAINST THE NOVEL CORONAVIRUS')
print('1.DAILY CASES')
print('2.DAILY COVID DEATHS')
print('3.DAILY RECOVERIES')
print('4.TOTAL CASES IN OUR HOSPITAL')
print('5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID')
print('6.TOTAL RECOVERIES IN OUR HOSPITAL')
print('7.ANALYSE COVID DATA')
print("8.SEE ALL COVID PATIENTS' DATA")
print('9.OUR COVID INITIATIVES')
print('10.AVERAGE CASES PER DAY')
print('11.AVERAGE DEATHS PER DAY')
print('12.AVERAGE RECOVERIES PER DAY')
print('13.most cases')
print('14.most deaths')
print('15.most recoveries')
print('16.least cases')
print('17.least deaths')
print('18.least recoveries')
kree=int(input('enter choice'))
if kree==1:
    print('DAILY COVID CASES')
    daily_cc()
if kree==2:
    print('DAILY COVID DEATHS')
    daily_cd()
if kree==3:
    print('DAILY COVID RECOVERIES')
    daily_cr()
if kree==4:
    print('TOTAL CASES IN OUR HOSPITAL')
    total_cc()
if kree==5:
```

```
print('TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID')
total_cd()
if kree==6:
    print('TOTAL RECOVERIES IN OUR HOSPITAL')
    total_cr()
if kree==7:
    print('ANALYSE COVID DATA')
    analyse_covidata()
if kree==8:
    print('DISPLAY COVID PATIENTS DATA')
    a_display_covidpatients()
if kree==9:
    print('OUR COVID INITIATIVES!')
    cov_display_init()
if kree==10:
    print('10.AVERAGE CASES PER DAY')
    avg_case()
if kree==11:
    print('11.AVERAGE DEATHS PER DAY')
    avg_death()
if kree==12:
    print('12.AVERAGE RECOVERIES PER DAY')
    avg_recoveries()
if kree==13:
    most_cases()
if kree==14:
    most_deaths()
if kree==15:
    most_recoveries()
if kree==16:
    least_cases()
if kree==17:
    least_deaths()
if kree==18:
    least_recoveries()
```

```
def doctor_view_prof():
    try:
        con=connection()
        cur=con.cursor()
        d_id=int(input("Enter your id doctor: "))
        cur.execute("select * from doctors where d_id=%d"%(d_id))
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)

def appointment_insertion():
    try:
        con=connection()
        cur=con.cursor()
        d_id=int(input("Enter the doctor's id:"))
        p_id=int(input("Enter the patient's id:"))
        time=input("Enter the consultation time:")
        cur.execute("insert into
appointments(p_id,d_id,time)values(%d,%d,'%s')"% (d_id,p_id,time))
        print()
        print("Appointments added successfully!")
        con.commit()
    except ma.Error as er:
        print(er)

def appointment_display_admin():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select * from appointments")
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)
```

```
def appointment_deletion_admin():
    try:
        con=connection()
        cur=con.cursor()
        p_id=int(input("Enter the patient's id to be deleted"))
        cur.execute("delete from appointments where p_id=%d"%(p_id))
        print()
        con.commit()
        print("Appointment deleted successfully!")
    except ma.Error as er:
        print(er)

def appointment_display_doc():
    try:
        con=connection()
        cur=con.cursor()
        d_id=int(input("Enter the doctor's id: "))
        cur.execute("select * from appointments where d_id=%d"%(d_id))
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)

def appointment_deletion_doc():
    try:
        con=connection()
        cur=con.cursor()
        d_id=int(input("Enter the doctor's id: "))
        p_id=int(input("Enter the patient's id to be deleted"))
        cur.execute("delete from appointments where p_id=%d and
d_id=%d"%(p_id,d_id))
        print()
        con.commit()
        print("Appointment deleted successfully!")
    except ma.Error as er:
        print(er)
```

```
def doctor():
    try:
        idi=int(input('enter doctor id'))
        con=connection()
        cur=con.cursor()
        b=int(input("enter the doc pass: "))
        cur.execute("select      doc_pwd      from      doc_pwd      where
doc_id=%d"%(idi))
        passw=cur.fetchone()
        password=passw[0]
        ans='y'
        if b==password:
            while ans=='y':
                print("Welcome Doctor")
                print("Rejuvenate Hospitals")
                print("1.Profile")
                print("2.Appointments")
                print("3.Change password")
                print("4.Prescription")
                choice=int(input("enter your choice"))
                if choice==1:
                    doctor_view_prof()

                elif choice==2:
                    print("1.Appointment display")
                    print("2.Appointment deletion")
                    opt=int(input("Enter your choice:"))
                    if opt==1:
                        print("Your appointments are below doctor:")
                        appointment_display_doc()
                    else :
                        s=input("Do you want to delete an appointment: ")
                        if s=='y':
                            n=int(input("Number of appointments wanted to be
deleted: "))
```

```

for i in range(n):
    appointment_deletion_doc()
    print("Appointment deleted successfully!")
else:
    print("you cannot delete")
elif choice==3:
    print('CHANGE PASSWORD')
    m=int(input('enter old password'))
    if m==password:
        new=int(input('enter new password'))
        newr=int(input('confirm new doctor password'))
        if new==newr:
            cur.execute("update doc_pwd set doc_pwd=%d where
doc_id=%d"%(new,idi))
            con.commit()
            print(' password change was successful')
        else:
            print('OOPS!confirmation failure')

elif choice==4:
    #1] TAKING VOICE INPUT
    import speech_recognition as sr

    # obtain audio from the microphone
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Say patient name!")
        r.adjust_for_ambient_noise(source)
        audio = r.listen(source)
        name=r.recognize_google(audio)

    with sr.Microphone() as source:
        print("Symptoms")
        r.adjust_for_ambient_noise(source)
        audio = r.listen(source)
        sympt=r.recognize_google(audio)

```

```
with sr.Microphone() as source:  
    print("Prescription")  
    r.adjust_for_ambient_noise(source)  
    audio = r.listen(source)  
    diag=r.recognize_google(audio)  
  
with sr.Microphone() as source:  
    print("Advice")  
    r.adjust_for_ambient_noise(source)  
    audio = r.listen(source)  
    advice=r.recognize_google(audio)  
  
# recognize speech using Google Speech Recognition  
try:  
    # for testing purposes, we're just using the default API  
key  
    # to use another API key, use `r.recognize_google(audio,  
key="GOOGLE_SPEECH_RECOGNITION_API_KEY")`  
    # instead of `r.recognize_google(audio)`  
    print("Name-",name)  
    print("Symptoms-",sympt)  
    print("Prescription-",diag)  
    print("Advice-",advice)  
    pass  
  
except sr.UnknownValueError:  
    print("Google Speech Recognition could not understand  
audio")  
except sr.RequestError as e:  
    print("Could not request results from Google Speech  
Recognition service; {0}".format(e))
```

```
#2] CREATING TEXT FILE
# Python code to create a file
file = open('pres.txt','w')
file.write(name+"\n")
file.write(sympt+"\n")
file.write(diag+"\n")
file.write(advice+"\n")
file.close()
import os
file = "pres.txt"
os.system(file)
```

```
#3]CREATING DIGITAL PRESCRIPTION
l=[['Name','Symptoms','Prescription','Advice']]
li=[]
```

```
with open("pres.txt") as file:
    dataa = file.read()
    li.append(dataa.split("\n"))
[li]=li
li.pop()
l.append(li)
data=l
fileName = 'pres.pdf'
```

```
from reportlab.platypus import SimpleDocTemplate
from reportlab.lib.pagesizes import letter
```

```
pdf = SimpleDocTemplate(
    fileName,
    pagesize=letter
)
```

```
from reportlab.platypus import Table
table = Table(data)
```

```

# add style
from reportlab.platypus import TableStyle
from reportlab.lib import colors

style = TableStyle([
    ('BACKGROUND', (0,0), (3,0), colors.green),
    ('TEXTCOLOR',(0,0),(-1,0),colors.whitesmoke),

    ('ALIGN',(0,0),(-1,-1),'CENTER'),
    ('FONTNAME', (0,0), (-1,0), 'Courier-Bold'),
    ('FONTSIZE', (0,0), (-1,0), 14),

    ('BOTTOMPADDING', (0,0), (-1,0), 12),
    ('BACKGROUND',(0,1),(-1,-1),colors.beige),
])
table.setStyle(style)

# Alternate background color
rowNumb = len(data)
for i in range(1, rowNumb):
    if i % 2 == 0:
        bc = colors.burlywood
    else:
        bc = colors.beige

    ts = TableStyle(
        [ ('BACKGROUND', (0,i),(-1,i), bc)]
    )
    table.setStyle(ts)

# Add borders
ts = TableStyle(
    [
        ('BOX',(0,0),(-1,-1),2,colors.black),

```

```

        ('LINEBEFORE',(2,1),(2,-1),2,colors.white),
        ('LINEABOVE',(0,2),(-1,2),2,colors.white),
        ('GRID',(0,1),(-1,-1),2,colors.black),
    ]
)
table.setStyle(ts)

elems = []
elems.append(table)

pdf.build(elems)

ans=input('enter y to continue')
else:
    print("password is incorrect")
except ma.Error as er:
    print(er)

def pharm_add():
try:
    con=connection()
    cur=con.cursor()
    mid=int(input("Enter the id of the medicine:"))
    med=input("Enter the name of the medicine:")
    pharma=input("Enter the pharma company:")
    price=int(input("Enter the price:"))
    num=int(input("Enter the number of medicines available:"))
    mfg=input("Enter the mfg date:")
    exp=input("enter the expiry date:")#create table pharma (mid
int(5),med varchar(25),pharma varchar(25),price int(10),num int(20),mfg
date ,exp date);
    cur.execute("insert into pharma
(mid,med,pharma,price,num,mfg,exp)values(%d,'%s','%s',%d,%d,'%s','
%s')%(mid,med,pharma,price,num,mfg,exp))
```

```
print()
con.commit()
print("Data inserted successfully")
except ma.Error as er:
    print(er)
```

```
def pharm_update():
    try:
        con=connection()
        cur=con.cursor()
        mid=int(input("Enter the id of the medicine you want to update:"))
        med=input("Enter the name of the medicine: ")
        pharma=input("Enter the name of the pharma company: ")
        price=int(input("Enter the price:"))
        num=int(input("Enter the number of medicines available:"))
        mfg=input("Enter the mfg date:")
        exp=input("enter the expiry date:")
        cur.execute("update pharma set
med=%s,pharma=%s,price=%d,num=%d,mfg=%s,exp=%s where
mid=%d"%(med,pharma,price,num,mfg,exp,mid))
        con.commit()
        print("Data updated successfully")
    except ma.Error as er:
        print(er)
```

```
def pharm_delete():
    try:
        con=connection()
        cur=con.cursor()
        mid=int(input("Enter the id of the medicine you want to delete:"))
        cur.execute("delete from pharma where mid=%d"%(mid))
        con.commit()
        print("Record deleted")
    except ma.Error as er:
        print(er)
```

```
def pharm_show():
    try:
        con=connection()
        cur=con.cursor()
        cur.execute("select * from pharma")
        for i in cur.fetchall():
            print(i)
    except ma.Error as er:
        print(er)

def bb_ins():
    try:
        con=connection()
        cur=con.cursor()
        Id=int(input('enter donor id'))
        fname=input("enter father's name")
        mname=input("enter mother's name")
        lname=input("enter last name")
        sex=input('enter gender of donor')
        b_type=input('enter blood group')
        bday=input('enter birthdate of donor in yy-mm-dd')
        h_address=input('enter address')
        city=input('enter city')
        don_date=input('enter date of donation(latest)')
        ##### stats=input('enter status')
        temp=input('enter body temperature at the time of donation in
fahrenheit')
        pulse=input('enter pulse rate at donation')
        bp=input('enter blood pressure at donation')
        weight=int(input('enter weight'))
        haemoglobin=input('enter haemoglobin count and specifications')
        plateletcount=int(input('enter platelet count'))
        hbsag=input('enter hbsag')
        aids=input('enter HIV status')
        malaria_smear=input('enter malaria smear')
```



```

hbsag=input('enter hbsag')
aids=input('enter HIV status')
malaria_smear=input('enter malaria smear')
hematocrit=input('enter hematocrit')
phone=int(input('enter phone no.'))
mobile=int(input('enter mobile number'))
cur.execute("update          tblblooddonors      set
fname='%s',mname='%s',lname='%s',sex='%s',b_type='%s',bday='%s',h_
address='%s',city='%s',don_date='%s',temp='%s',pulse='%s',bp='%s',wei
ght=%d,haemoglobin='%s',plateletcount=%d,hbsag='%s',aids='%s',mala
ria_smear='%s',hematocrit='%s',phone=%d,mobile=%d      where
Id=%d"%(fname,mname,lname,sex,b_type,bday,h_address,city,don_dat
e,temp,pulse,bp,weight,haemoglobin,plateletcount,hbsag,aids,malaria_s
mear,hematocrit,phone,mobile,Id))
con.commit()
print('updated successfully')
except ma.Error as er:
    print(er)
def bb_del():
try:
    con=connection()
    cur=con.cursor()
    Id=int(input('enter donor id'))
    cur.execute('delete from tblblooddonors where Id=%d'%(Id))
    con.commit()
    print('donor data deleted successfully')
except ma.Error as er:
    print(er)
def bb_bg_upd():
try:
    con=connection()
    cur=con.connection()
    bg_id=int(input('enter blood group id'))
    BloodGroup=input('enter blood group')
    PostingDate=input('enter latest posting date in yy-mm-dd format')
    stock=input('enter blood stock')

```

```
        cur.execute("update          tblbloodgroup      set
        BloodGroup='%s',PostingDate='%s',stock='%s'
        bg_id=%d"%(bg_id,BloodGroup,PostingDate,stock))    where
        con.commit()
        print('updation success')
except ma.Error as er:
        print(er)

def elli_don():
    try:
        print('1.to view the standard GENERAL POLICY FOR BLOOD
DONATION \n2.COVID EXTRA RULES ON BLOOD DONATION
SAFETY \n ')
        chi=int(input('enter your choice'))
        if chi==1:
            print('Must be at least 16 years old (16- and 17-year-olds must
bring a signed permission form from a parent or guardian, if required by
state or school')
            print('Weigh at least 110 pounds. Certain height/weight criteria
may apply for donors 22 years old or younger.')
            print('Be in good general health.')
            print('For your safety and to ensure a positive donation
experience, make sure you eat within two hours ahead of your donation.
Drink plenty of water that day and 24 to 48 hours beforehand. Feel free
to help yourself to something to eat and drink in our refreshment area.')
            print('Bring your ID – something with your name and one of the
following: date of birth, donor ID number or your photo.')
            print('You must wait eight weeks between whole blood donations.
Learn more about specific intervals for other types of donation.')
            print('You should not be under the influence of alcohol or
recreational drugs at the time of donation.')
            print('Additional Requirements: Component-Specific Donation
include that Platelet donors should avoid aspirin and/or aspirin-
containing products 48 hours prior to donation and other anti-platelet
medications')
            print('Updated Donation Eligibility Criteria')
            print('Health Conditions')
```

```
print('Medications')
print('HIV/ AIDS Risk Behaviors')
print('Tattoos, Piercings, Permanent Make-up & Acupuncture')
print('European Travel/Residency Criteria')
print('Travel to Malaria Risk Areas')
print('Source Plasma')

elif chi==2:
    print("Due to COVID-19 (SARS-CoV-2 Coronavirus), we are
requiring that people should not donate today ,if in the PAST 4 WEEKS:
    You had symptoms from a lab-diagnosed or suspected
COVID-19 infection and have not had a subsequently-negative
nasopharyngeal swab test result
    You have lived with or been in close contact with individuals
diagnosed with or suspected of having COVID-19 infection
    You are a health care worker who has been caring for a patient
diagnosed with or suspected of having COVID-19 and have not
consistently been able to use recommended personal protective
equipment (face mask, gown and gloves)

Masks Required
All staff and donors are required to wear a mask or cloth face
covering. One-way valve masks are not permitted.

COVID-19 Convalescent Plasma
Learn more about convalescent plasma to see if you are
eligible to give this lifesaving donation to help COVID-19 patients. We
encourage healthy donors who don't qualify to give convalescent plasma
to continue scheduling whole blood, platelet and other donation type
appointments to help patients in need")

except ma.Error as er:
    print(er)

def admin():
    try:
        a=int(input('enter admin password:'))
```

```

con=connection()
cur=con.cursor()
cur.execute("select admin_pwd from adminpwd where
name_admin='administrator'")
passw=cur.fetchone()
password=passw[0]
ans='y'
if a==password:
    while ans=='y':
        print('WELCOME ADMIN')
        print('REJUVENATE HOSPITALS')
        print('1.PATIENT MODULE')
        print('2.DOCTOR MODULE')
        print('3.WARDS MODULE')
        print('4.REJUVENATE BLOOD BANK MODULE')
        print('5.COVID MODULE')
        print('6.CHANGE YOUR PASSWORD')
        print('7.REJUVENATE PHARMACY')
        print('8.GENERATE BILL')
        choice=int(input('enter your choice'))
if choice==1:
    print("PATIENT'S MODULE")
    print("1.NEW PATIENT")
    print("2.PATIENTS RECORDS")
    print("3.UPDATE PATIENTS RECORD")
    print("4.DISCHARGE OF PATIENT")
    x=int(input("Enter your choice:"))
    if x==1:
        a_display_ncpatients()
        n=int(input("Enter the number of records you want to add:
"))
        for i in range(n):
            a_insert_ncpatients()
            print()
            print("INSERTION SUCCESSFUL")

```

```
elif x==2:  
    print("HERE IS THE LIST OF PATIENTS RECORDS:")  
    print()  
    a_display_ncpatients()  
    print()  
  
elif x==3:  
    a_display_ncpatients()  
    m=int(input("Enter the number of records you want to  
update:"))  
    for i in range (m):  
        a_update_ncpatients()  
    print(" UPDATION SUCCESSFUL")  
    a_display_ncpatients()  
  
else:  
    a_display_ncpatients()  
    p=int(input("Enter the number of records you want to  
delete:"))  
    for i in range(p):  
        a_delete_ncpatients()  
    print("DELETION SUCCESSFUL")  
    a_display_ncpatients()  
  
elif choice==2:  
    print('doctor module')  
    print("submodules")  
    print("1.ADD DOCTOR")  
    print("2.LIST OF DOCTORS")  
    print("3.UPDATE DOCTOR RECORD")  
    print("4.REMOVAL OF A DOCTOR RECORD")  
    print("5.APPOINTMENTS")  
    ch=int(input("enter your choice"))  
    if ch==1:  
        d_display_doctor()
```

```

n=int(input("Enter the number of records you want to add
"))
for i in range(n):
    d_insert_doctor()
    print()
    print("DOCTOR'S DETAIL ADDED SUCCESSFULLY!")
elif ch==2:
    print("Here is the list of doctors:")
    print()
    d_display_doctor()
    print()
elif ch==3:
    d_display_doctor()
n=int(input("Enter the number of records you want to
update:"))
for i in range(n):
    d_update_doctor()
print("UPDATED SUCCESSFULLY!")
d_display_doctor()
elif ch==4:
    d_display_doctor()
n=int(input("Enter the number of records you want to
delete:"))
for i in range(n):
    d_delete_doctor()
print("DELETED SUCCESSFULLY!")
d_display_doctor()
else:
    d_display_doctor()
    print("1.Appointment Addition")
    print("2.Appointment Display")
    print("3.Appointment Deletion")
    ch=int(input("Enter your choice: "))
if ch==1:
    n=int(input("Enter number of appointments"))
    for i in range(n):
        appointment_insertion()

```

```

        print()
        print("Appointment added successfully")

elif ch==2:
    print("The appointments are: ")
    appointment_display_admin()
    print()

else:
    op=input("Do you want to delete an appointment")
    if op=='y':
        appointment_deletion_admin()
        print("Appointment deleted successfully!")
    else:
        print("*****")
    #ADD NEW WARD: INPUT.
BUILDING ,TYPE: (icu [I65]ward, covid [COR35]isolation ward,
# general[7869],VIP[VIP448]),ROOM NO.,NUMBER OF BEDS, STATUS:
OCCUPIED/FREE, ROOM RENT ROOM INFORMATION. INPUT
ROOM NO. DISPLAY ABOVE .
# UPDATE WARD: INPUT ROOM NO.
    print('3.wards module')
    print('submodules')
    print('1.ADD NEW WARD')
    print('2.ROOM INFORMATION and AVAILABILITY')
    print("3.UPDATE WARD")
    choke=int(input('enter choice'))
    #cur.execute('create      table      WARDS(building
varchar(10),room_number      int(10),number_of_beds      int(10),status
varchar(10),room_rent int(55))')
    if choke==1:
        a_display_wards()
        print()
        a_insert_wards()

```

```
print()
a_display_wards()
elif choke==2:
    a_display_wards()
    print()

elif choke==3:
    a_display_wards()
    print()
    a_update_wards()
    print("Do You Want To update More Records")
    c=input("Enter y to update further")
    if c=='y':
        a_update_wards()
    else:
        print("^-^")
        a_display_wards()

elif choice==4:
    print('4.REJUVENATE BLOOD BANK MODULE')
    print('\n{Drop of Blood})*20)
    print('1.insert into blood donors')
    print('2.change donor details')
    print('3.delete donor')
    print('4.update blood group details')
    print('5.see the list of eligible donors of this day')
    chuc=int(input('enter valid choice'))
    ascb='y'
    while ascb=='y':
        if chuc==1:
            bb_ins()
        if chuc==2:
            bb_upd()
        if chuc==3:
            bb_del()
        if chuc==4:
            bb_bg_upd()
```

```

elif chuc==5:
    elli_don()
ascb=input('enter y to continue in bbms')

elif choice==5:
    print('5.COVID MODULE')
    print('SUBMODULES')
    print()
    print('1.COVID PATIENTS')
    choic=int(input('enter your choice'))
    #cur.execute('create table COVID19_PATIENTS(patient_id
int(20),age int(2),gender varchar(10),room int ,date' )
    #cur.execute("ALTER TABLE COVID19_PATIENTS ADD
COLUMN name varchar(20) ")
    # ive commented as it has already been created
    if choic==1:
        print('1.current covid patients data\n2.add new
patient\n3.delete recovered patient records\n4.update patient info\n
5.insertion-OUR COVID INITIATIVES\n6. OUR COVID INITIATIVES')
        dep=int(input('enter choice'))
        if dep==1:
            a_display_covidpatients()
        if dep==2:
            a_display_covidpatients()
            print()
            a_insert_covidpatients()
        if dep==3:
            a_display_covidpatients()
            print()
            a_delete_covidpatients()
            print("Do You Want To Delete More Records")
            c=input("Enter y to delete further")
            if c=='y':
                a_delete_covidpatients()
            else:
                print("^-^")

```

```

if dep==4:
    a_display_covidpatients()
    print()
    print(a_update_covidpatients())
    print("Do You Want To update More Records")
    c=input("Enter y to update further")
    if c=='y':
        a_update_covidpatients()
    else:
        print("^_^")
if dep==5:
    print('TO ADD NEW TEXT UNDER COVID 19
INITIATIVES')
    cov_init()
if dep==6:
    print('our COVID INITIATIVES')
    cov_display_init()

if choice==6:
    print('change password')
    m=int(input('enter old password'))
    if m==password:
        new=int(input('enter new password'))
        newr=int(input('confirm new admin password'))
        if new==newr:
            cur.execute("update adminpwd set admin_pwd=%d
where name_admin='administrator'||%(new))
            con.commit()
            print(' password change was successful')

    else:
        print('OOPS!confirmation failure')
if choice==7:
    print("WELCOME TO BIOHEAL PHARMACY")
    print("MENU")
    print("1.Show Stock")
    print("2.Add medicine into stock")

```

```

print("3.Update medicine stock")
print("4.Delete medicine stock")

ansg='y'

while ansg=='y':
    choice=int(input("Enter your choice:"))
    if choice==1:
        pharm_show()
    elif choice==2:
        pharm_add()
    elif choice==3:
        pharm_update()
    elif choice==4:
        pharm_delete()
    ansg=input("Enter y to continue in the pharmacy
module:")

elif choice==8:
    def patients():
        print("PATIENT BILL")
        n=int(input("Please enter your patient id : "))
        try:
            con=connection()
            cur=con.cursor()
            cur.execute("select * from patients where p_id=%d"%(n))
            for i in cur.fetchall():
                print(i)

        print("Please verify your details")
        print("TYPE OF TREATMENT:")
        print("1.CONSULTATION")
        print("2.SURGERY")

        x=int(input("Enter your type of treatment:"))
        if x==1:
            print("CONSULTAION FEES:₹500")
        elif x==2:

```

```
surgery()

except ma.Error as er:
    print(er)

def surgery():
    room=input("ENTER      YOUR      ROOM      TYPE
(GENERAL/VIP/COVID) : ")
    if room=="general"or "GENERAL":
        a=200#admission rate
        b=2500#LAB TESTS TO BE INCLUDED ?bed charge
        s=45000#surgeon charge
        a_s=25000#asst. surgeon charge
        tc=6000#Theatre charges
        af=12000#Anathetist fees
        sc=22164#Service charges
        e=18400#Equipment charges
    elif room=="VIP" or "vip":
        a=200#admission rate
        b=2500#LAB TESTS TO BE INCLUDED ?bed charge
        s=45000#surgeon charge
        a_s=25000#asst. surgeon charge
        tc=6000#Theatre charges
        af=12000#Anathetist fees
        sc=22164#Service charges
        e=18400#Equipment charges
    else :
        a=200#admission rate
        b=2500#LAB TESTS TO BE INCLUDED ?bed charge
        s=45000#surgeon charge
        a_s=25000#asst. surgeon charge
        tc=6000#Theatre charges
        af=12000#Anathetist fees
        sc=22164#Service charges
        e=18400#Equipment charges
```

```
n=input("ENTER THE BILL NO :")  
  
try:  
  
    con=connection()  
    cur=con.cursor()  
  
        cur.execute("create table bill"+n+" (SLNO int (2),  
SERVICES varchar(50) ,QTY int (20) ,RATE int(25) , AMOUNT int(50)) ;")  
        tat=1  
        eat=2  
        wat=3  
        cat=4  
        rat=5  
        sat=6  
        fat=7  
        kat=8  
        lat=9  
        yat=0  
        dog="ADMISSION_CHARGES"  
        fog="BED_CHARGES"  
        log="SURGEON_CHARGES"  
        gog="ASST_SURGEON_CHARGES"  
        kog="THEATRE_CHARGES"  
        sog="ANAESTHETIST_FEES "  
        tog="SERVICES"  
        aog="EQUIPMENT"  
        bog="TOTAL"  
  
        qb=int(input("ENTER THE NUMBER OF BEDS  
USED:"))  
        qt=int(input("ENTER THE NUMBER OF THEATERS  
USED :"))  
        qa=int(input("ENTER THE NUMBER OF  
ANAESTHESIA WAS GIVEN TO THE PATIENT: "))  
        qe=int(input("ENTER THE NUMBER OF  
EQUIPMENTS USED: "))
```

```

        cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%
d,%d)"%(tat,dog,tat,a,a))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%
d,%d)"%(eat,fog,qb,b,b*qb))
                con.commit()
                cur.execute("insert           into      bill"+n+
(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%d,%d)"%
(wat,log,tat,s,s))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%
d,%d)"%(cat,gog,tat,a_s,a_s))
                con.commit()
                cur.execute("insert           into      bill"+n+
(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%d,%d)"%
(rat,kog,qt,tc,tc*qt))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)
values(%d,'%s',%d,%d,%d)"%(sat,sog,qa,af,af*qa))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)
values(%d,'%s',%d,%d,%d)"%(fat,tog,tat,sc,sc))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)
values(%d,'%s',%d,%d,%d)"%(kat,aog,qe,e,qe*e))
                con.commit()
                cur.execute("insert           into
bill"+n+"(SLNO,SERVICES,QTY,RATE,AMOUNT)values(%d,'%s',%d,%
d,%d)"%(lat,bog,yat,yat,a+b*qb+s+a_s+tc*qt+af*qa+sc+qe*e))
                con.commit()
                print("data inserted successfully")

```

```

        except ma.Error as er:
            print(er)
    def payment():
        print("PAYMENT MODES:")
        print("1.CREDIT CARD")
        print("2.CASH")
    def final():
        print("REJUVENATE HOSPITAL")
        patients()
        payment()

    final()
    ans=input('enter y to continue')
else:
    print('password incorrect')
t='-----'
return t
except ma.Error as er:
    print(er)

def totale():
    print('Welcome user ! \n REJUVENATE HOSPITALS \n HOSPITAL
MANAGEMENT SYSTEM \n LEADERS IN HEALTHCARE
ADMINISTRATION')
    print('-----')
    print('1.CONTINUE AS ADMIN\n 2.CONTINUE AS DOCTOR \n
3.COVID SPECIALITY UNIT \n ')
    fun=int(input('enter choice'))
    if fun==1:
        admin()
    elif fun==2:
        doctor()
        print()
    elif fun==3:
        print('welcome user! COVID SPECIALITY UNIT\n JOIN US IN
OUR BATTLE AGAINST THE NOVEL CORONAVIRUS')
        print('1.DAILY CASES')

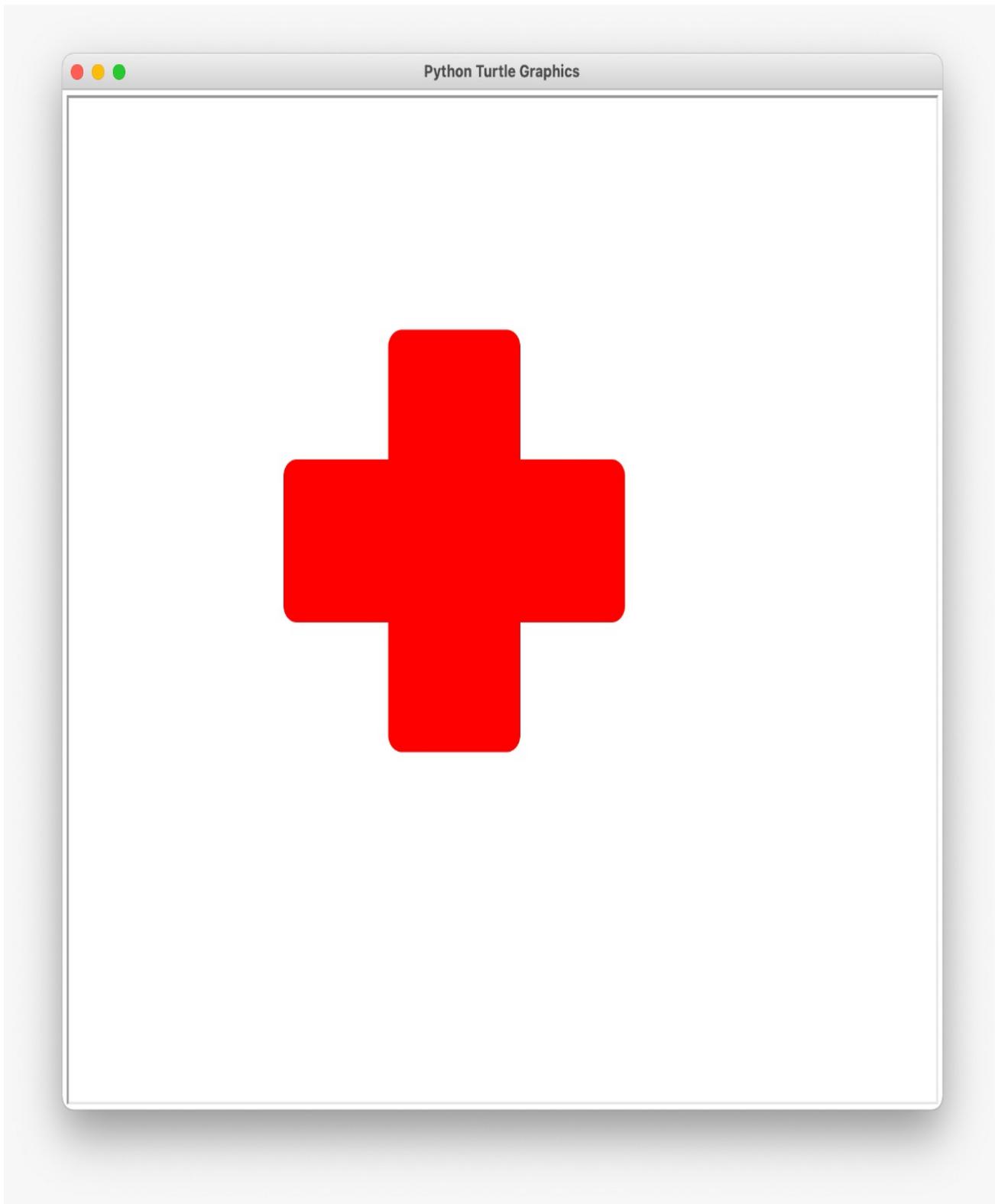
```

```
print('2.DAILY COVID DEATHS')
print('3.DAILY RECOVERIES')
print('4.TOTAL CASES IN OUR HOSPITAL')
print('5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID')
print('6.TOTAL RECOVERIES IN OUR HOSPITAL')
print('7.ANALYSE COVID DATA')
print("8.SEE ALL COVID PATIENTS' DATA")
print('9.OUR COVID INITIATIVES')
print('10.AVERAGE CASES PER DAY')
print('11.AVERAGE DEATHS PER DAY')
print('12.AVERAGE RECOVERIES PER DAY')
print('13.most cases')
print('14.most deaths')
print('15.most recoveries')
print('16.least cases')
print('17.least deaths')
print('18.least recoveries')
kree=int(input('enter choice'))
if kree==1:
    print('DAILY COVID CASES')
    daily_cc()
if kree==2:
    print('DAILY COVID DEATHS')
    daily_cd()
if kree==3:
    print('DAILY COVID RECOVERIES')
    daily_cr()
if kree==4:
    print('TOTAL CASES IN OUR HOSPITAL')
    total_cc()
if kree==5:
    print('TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID')
    total_cd()
if kree==6:
    print('TOTAL RECOVERIES IN OUR HOSPITAL')
    total_cr()
if kree==7:
```

```
print('ANALYSE COVID DATA')
analyse_covidata()
if kree==8:
    print('DISPLAY COVID PATIENTS DATA')
    a_display_covidpatients()
if kree==9:
    print('OUR COVID INITIATIVES!')
    cov_display_init()
if kree==10:
    print('10.AVERAGE CASES PER DAY')
    avg_case()
if kree==11:
    print('11.AVERAGE DEATHS PER DAY')
    avg_death()
if kree==12:
    print('12.AVERAGE RECOVERIES PER DAY')
    avg_recoveries()
if kree==13:
    most_cases()
if kree==14:
    most_deaths()
if kree==15:
    most_recoveries()
if kree==16:
    least_cases()
if kree==17:
    least_deaths()
if kree==18:
    least_recoveries()
totale()
```

# OUTPUT SCREENSHOTS:

\*HOSPITAL RED CROSS (USING TURTLE):



# MODULE 1 - COVID MODULE:

Welcome user !

REJUVENATE HOSPITALS

HOSPITAL MANAGEMENT SYSTEM

LEADERS IN HEALTHCARE ADMINISTRATION

---

1.CONTINUE AS ADMIN

2.CONTINUE AS DOCTOR

3.COVID SPECIALITY UNIT

enter choice3

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice

# 1.DAILY CASES:

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice1

DAILY COVID CASES

enter date:2020.11.20

3

## 2.DAILY COVID DEATHS:

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice2

DAILY COVID DEATHS

enter date:2020.11.20

3

### 3.DAILY RECOVERIES:

welcome user! COVID SPECIALITY UNIT  
JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

- 1.DAILY CASES
- 2.DAILY COVID DEATHS
- 3.DAILY RECOVERIES
- 4.TOTAL CASES IN OUR HOSPITAL
- 5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID
- 6.TOTAL RECOVERIES IN OUR HOSPITAL
- 7.ANALYSE COVID DATA
- 8.SEE ALL COVID PATIENTS' DATA
- 9.OUR COVID INITIATIVES
- 10.AVERAGE CASES PER DAY
- 11.AVERAGE DEATHS PER DAY
- 12.AVERAGE RECOVERIES PER DAY
- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice3  
DAILY COVID RECOVERIES

enter date:2020.11.21

3

## 4.TOTAL CASES:

```
untitled11.py', wdir='/Users/rajakumarisureshbabu')
welcome user! COVID SPECIALITY UNIT
JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS
1.DAILY CASES
2.DAILY COVID DEATHS
3.DAILY RECOVERIES
4.TOTAL CASES IN OUR HOSPITAL
5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID
6.TOTAL RECOVERIES IN OUR HOSPITAL
7.ANALYSE COVID DATA
8.SEE ALL COVID PATIENTS' DATA
9.OUR COVID INITIATIVES
10.AVERAGE CASES PER DAY
11.AVERAGE DEATHS PER DAY
12.AVERAGE RECOVERIES PER DAY
13.most cases
14.most deaths
15.most recoveries
16.least cases
17.least deaths
18.least recoveries
```

enter choice4

TOTAL CASES IN OUR HOSPITAL

18

## 5.TOTAL DEATHS:

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice5

TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

## 6.TOTAL RECOVERIES:

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

- 1.DAILY CASES
- 2.DAILY COVID DEATHS
- 3.DAILY RECOVERIES
- 4.TOTAL CASES IN OUR HOSPITAL
- 5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID
- 6.TOTAL RECOVERIES IN OUR HOSPITAL
- 7.ANALYSE COVID DATA
- 8.SEE ALL COVID PATIENTS' DATA
- 9.OUR COVID INITIATIVES
- 10.AVERAGE CASES PER DAY
- 11.AVERAGE DEATHS PER DAY
- 12.AVERAGE RECOVERIES PER DAY
- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice6

TOTAL RECOVERIES IN OUR HOSPITAL

## 7.ANALYSE COVID DATA:

```
In [1]: runfile('/Users/rajakumarisureshbabu/untitled11.py', wdir='/Users/rajakumarisureshbabu')
```

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice7

ANALYSE COVID DATA

# A) LINE CHART:

HOW'D YOU LIKE TO VIEW GRAPHICAL COVID DATA OF REJUVENATE HOSPITAL

- 1.d,c,r line chart
- 2.d,c,r line chart with circular denotion for all three in one graph
- 3.d,c,r bar graph
- 4.d,c,r scatter plot

enter choice:1

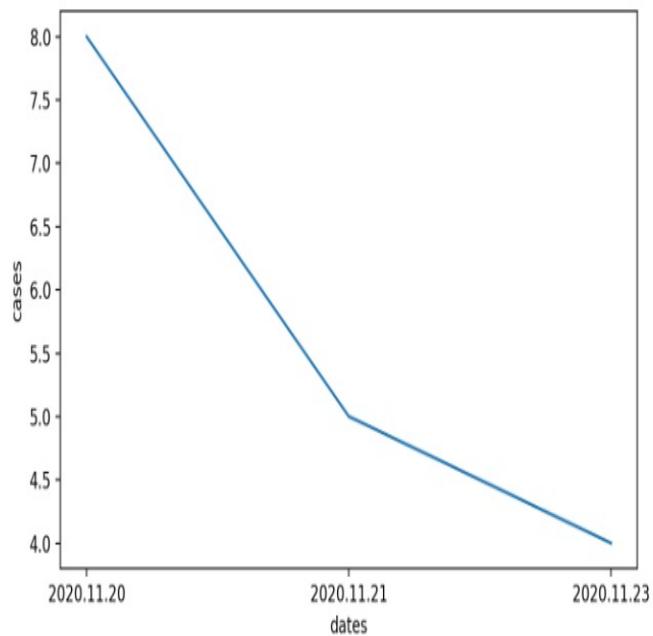
enter no. of dates' data' to be compiled3

enter date2020.11.20

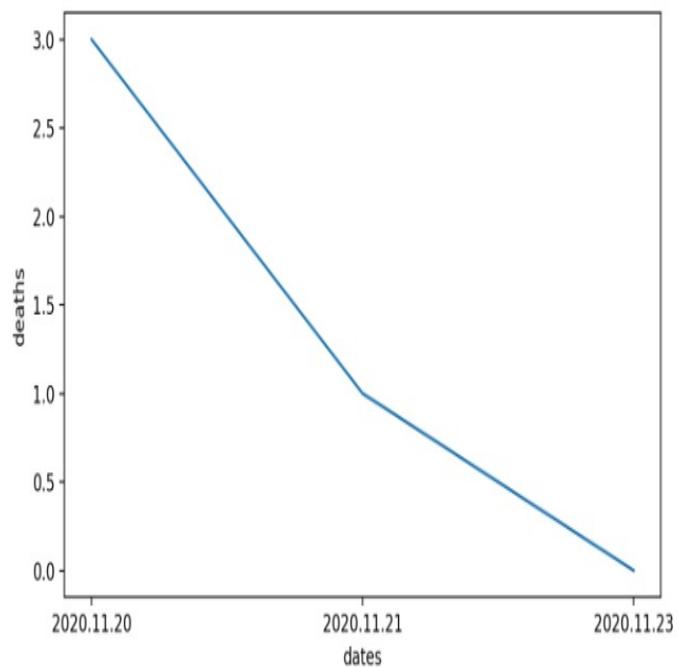
enter date2020.11.21

enter date2020.11.23

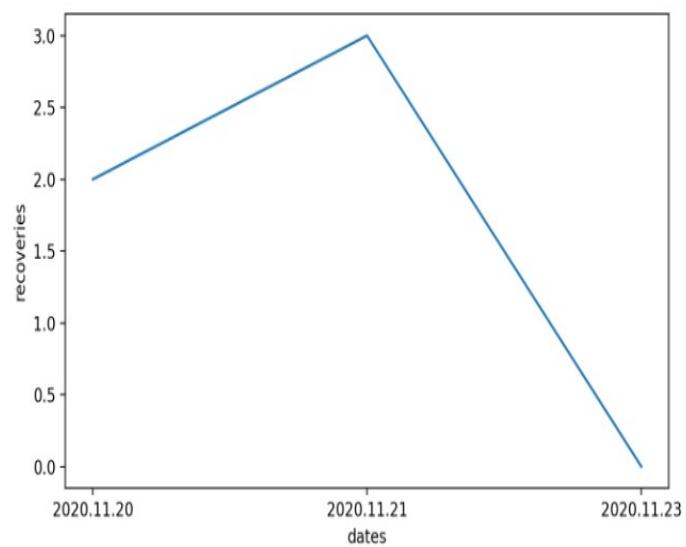
COVID cases IN REJUVENATE HOSPITAL -ON A DAILY BASIS



### COVID DEATHS IN REJUVENATE HOSPITAL -ON A DAILY BASIS



### COVID recoveries IN REJUVENATE HOSPITAL -ON A DAILY BASIS



## B) ALL DATA IN A NUTSHELL:

HOW'D YOU LIKE TO VIEW GRAPHICAL COVID DATA OF REJUVENATE HOSPITAL

1.d,c,r line chart

2.d,c,r line chart with circular denotation for all three in one graph

3.d,c,r bar graph

4.d,c,r scatter plot

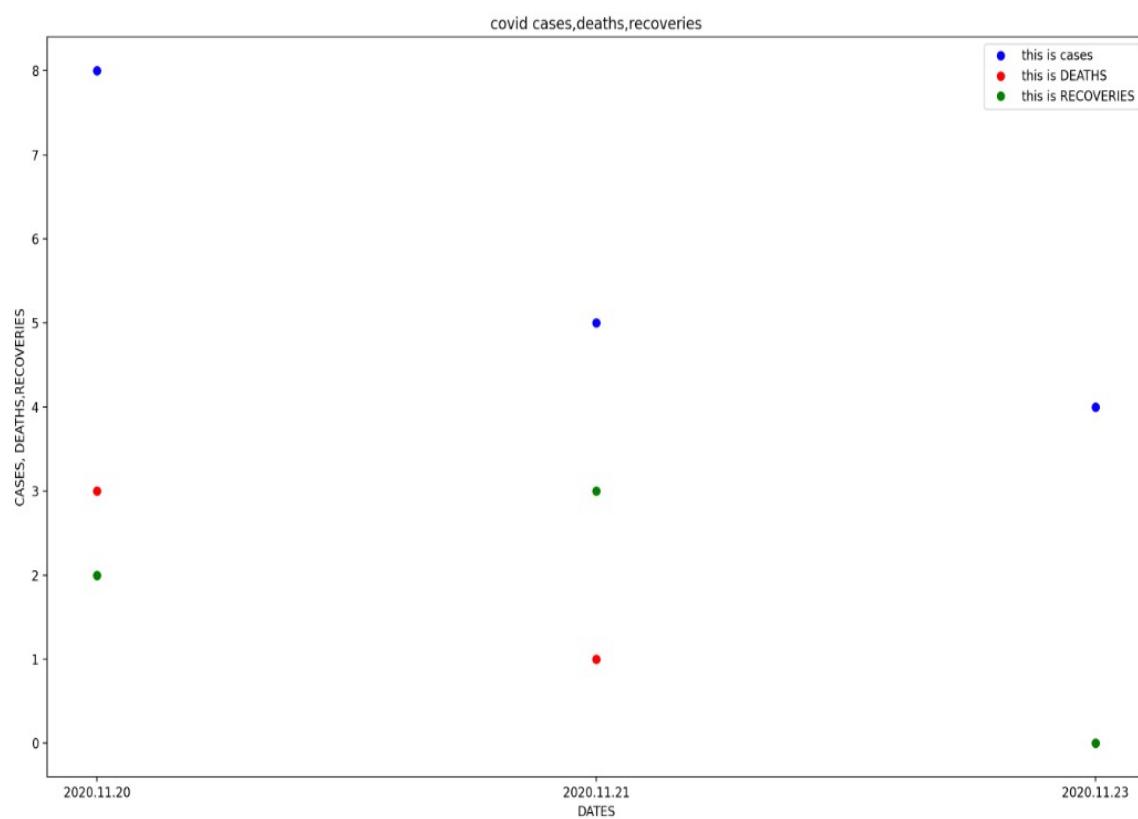
enter choice:2

enter no. of dates' data' to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23



## C)BAR PLOT:

HOW'D YOU LIKE TO VIEW GRAPHICAL COVID DATA OF REJUVENATE HOSPITAL

- 1.d,c,r line chart
- 2.d,c,r line chart with circular denotion for all three in one graph
- 3.d,c,r bar graph
- 4.d,c,r scatter plot

enter choice:3

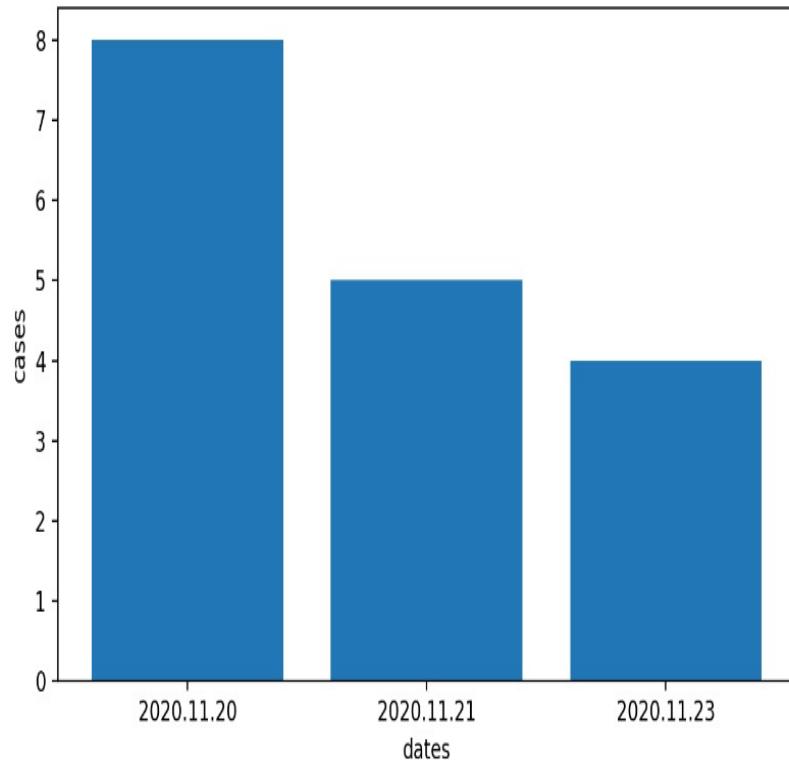
enter no. of dates' data' to be compiled3

enter date2020.11.20

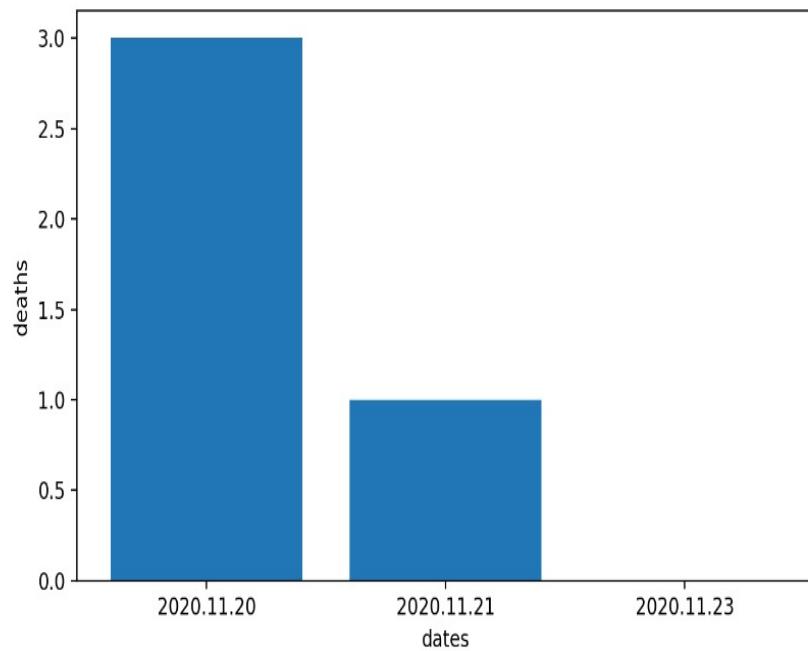
enter date2020.11.21

enter date2020.11.23

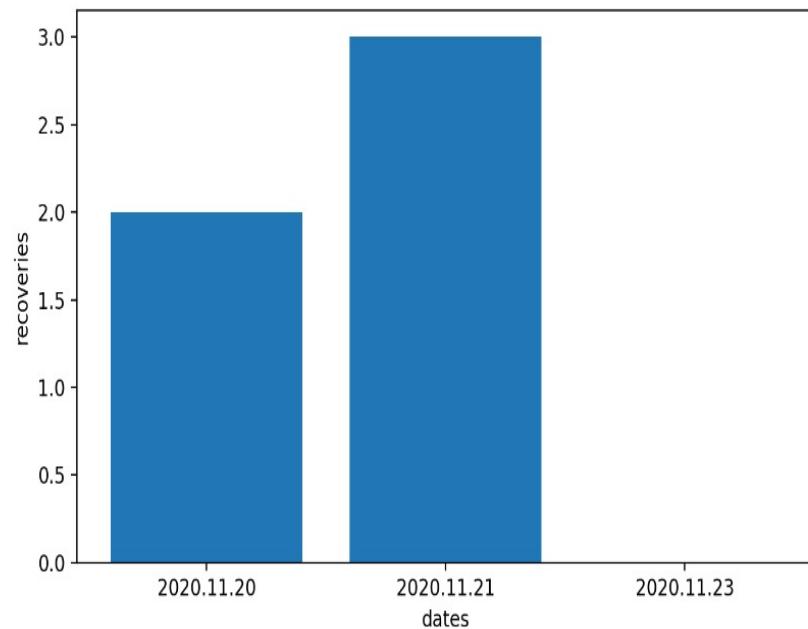
COVID cases IN REJUVENATE HOSPITAL -ON A DAILY BASIS



COVID DEATHS IN REJUVENATE HOSPITAL -ON A DAILY BASIS



COVID recoveries IN REJUVENATE HOSPITAL -ON A DAILY BASIS



## 4)SCATTER PLOT:

HOW'D YOU LIKE TO VIEW GRAPHICAL COVID DATA OF REJUVENATE HOSPITAL

- 1.d,c,r line chart
- 2.d,c,r line chart with circular denotion for all three in one graph
- 3.d,c,r bar graph
- 4.d,c,r scatter plot

enter choice:4

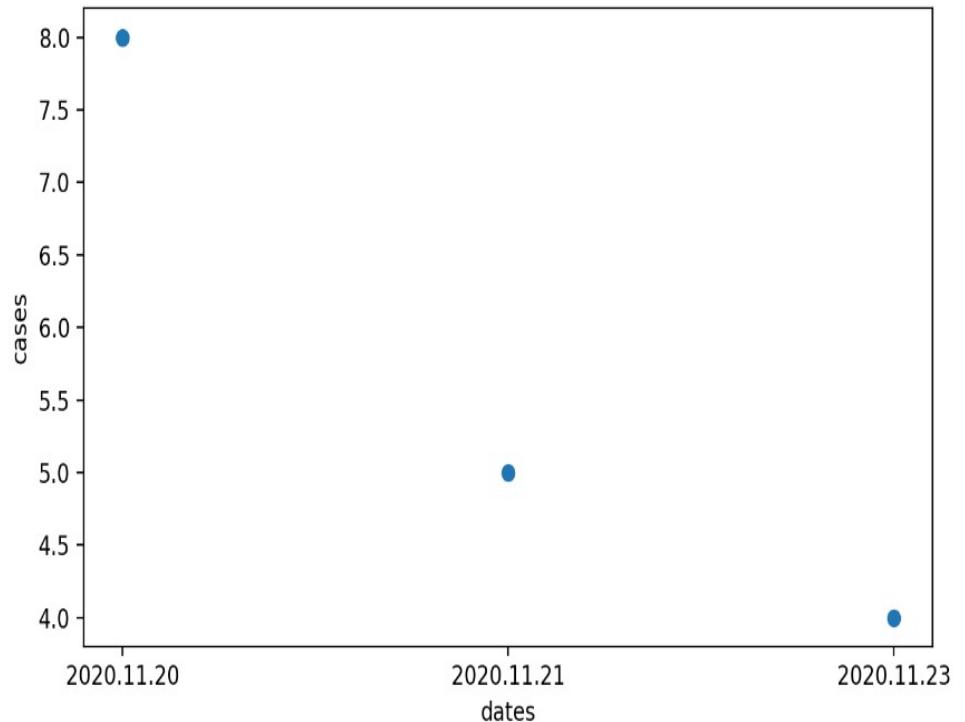
enter no. of dates' data' to be compiled3

enter date2020.11.20

enter date2020.11.21

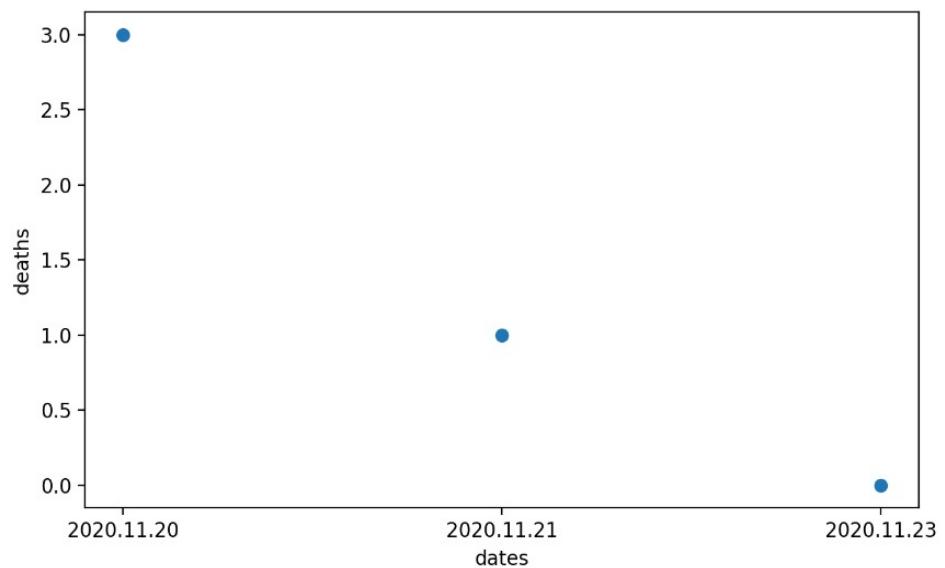
enter date2020.11.23

COVID cases IN REJUVENATE HOSPITAL -ON A DAILY BASIS



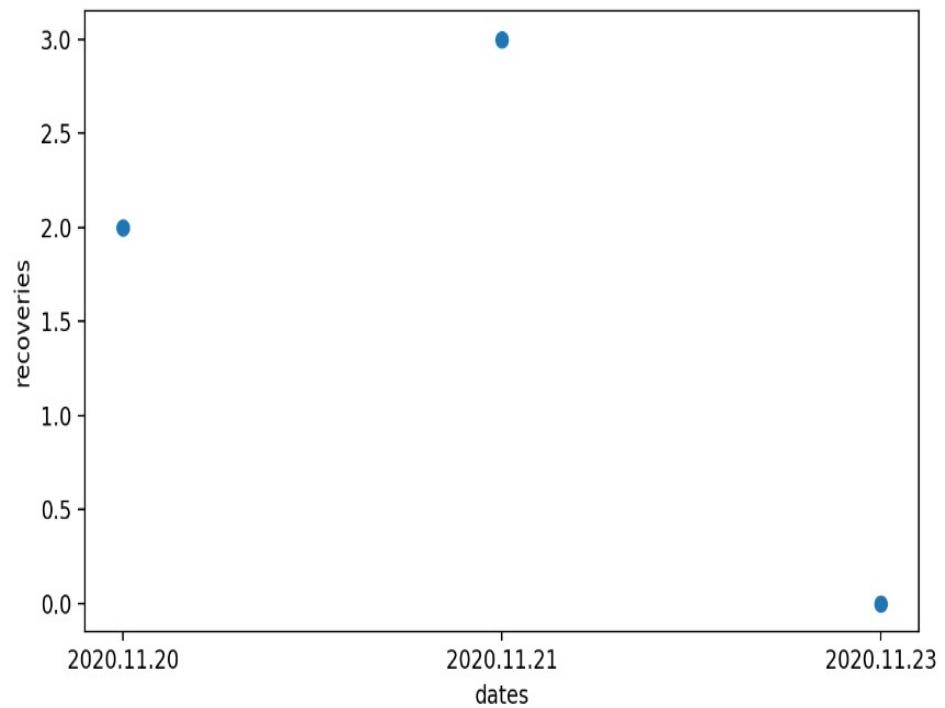
---

COVID DEATHS IN REJUVENATE HOSPITAL -ON A DAILY BASIS

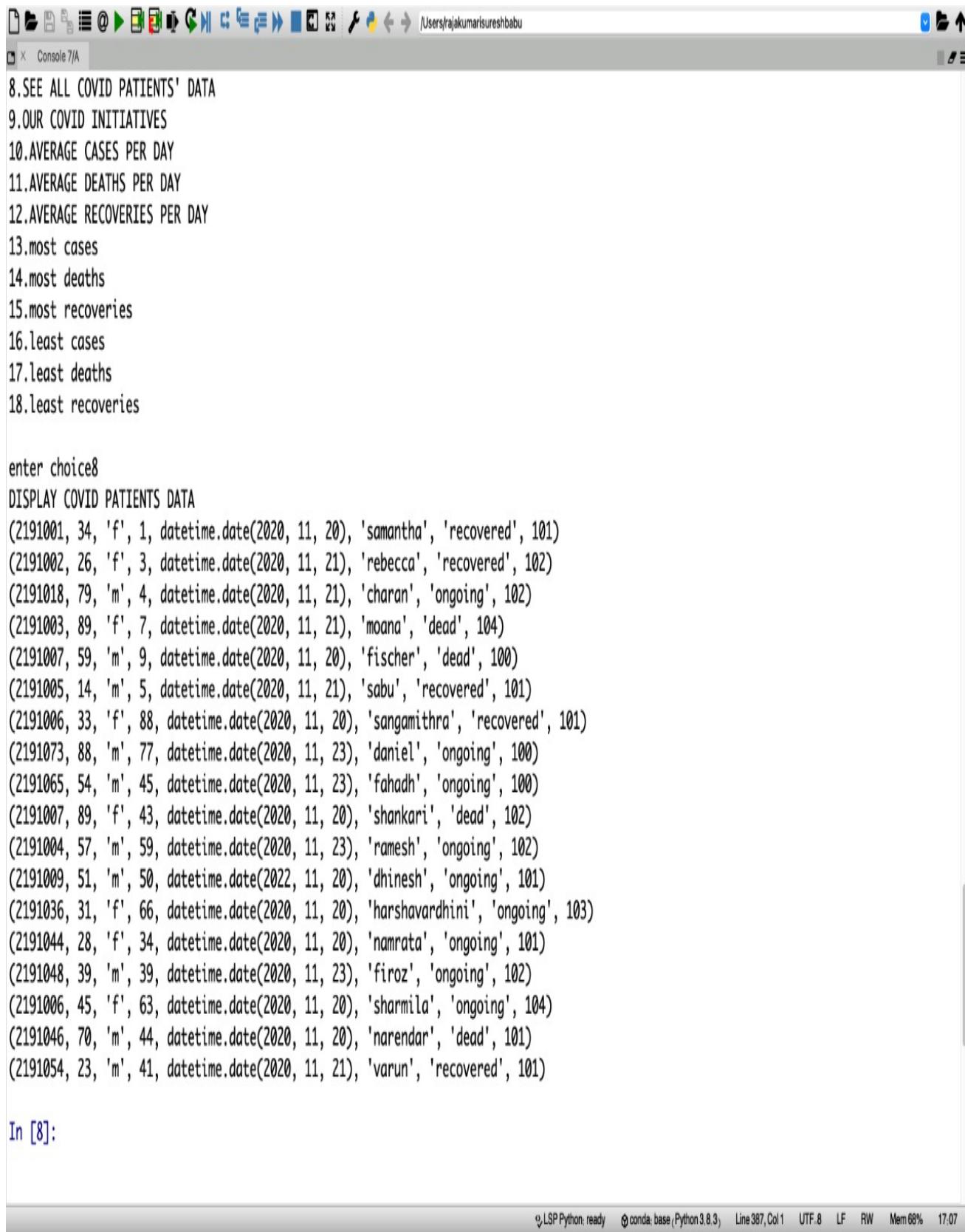


---

COVID recoveries IN REJUVENATE HOSPITAL -ON A DAILY BASIS



## 8)DISPLAY COVID PATIENTS' DATA:



The screenshot shows a Jupyter Notebook interface with a toolbar at the top and a code cell below it. The code cell contains a list of 18 numbered options related to COVID patients, followed by an input prompt and the displayed patient data.

```
8.SEE ALL COVID PATIENTS' DATA
9.OUR COVID INITIATIVES
10.AVERAGE CASES PER DAY
11.AVERAGE DEATHS PER DAY
12.AVERAGE RECOVERIES PER DAY
13.most cases
14.most deaths
15.most recoveries
16.least cases
17.least deaths
18.least recoveries

enter choice8
DISPLAY COVID PATIENTS DATA
(2191001, 34, 'f', 1, datetime.date(2020, 11, 20), 'samantha', 'recovered', 101)
(2191002, 26, 'f', 3, datetime.date(2020, 11, 21), 'rebecca', 'recovered', 102)
(2191018, 79, 'm', 4, datetime.date(2020, 11, 21), 'charan', 'ongoing', 102)
(2191003, 89, 'f', 7, datetime.date(2020, 11, 21), 'moana', 'dead', 104)
(2191007, 59, 'm', 9, datetime.date(2020, 11, 20), 'fischer', 'dead', 100)
(2191005, 14, 'm', 5, datetime.date(2020, 11, 21), 'sabu', 'recovered', 101)
(2191006, 33, 'f', 88, datetime.date(2020, 11, 20), 'sangamithra', 'recovered', 101)
(2191073, 88, 'm', 77, datetime.date(2020, 11, 23), 'daniel', 'ongoing', 100)
(2191065, 54, 'm', 45, datetime.date(2020, 11, 23), 'fahadh', 'ongoing', 100)
(2191007, 89, 'f', 43, datetime.date(2020, 11, 20), 'shankari', 'dead', 102)
(2191004, 57, 'm', 59, datetime.date(2020, 11, 23), 'ramesh', 'ongoing', 102)
(2191009, 51, 'm', 50, datetime.date(2022, 11, 20), 'dhinesh', 'ongoing', 101)
(2191036, 31, 'f', 66, datetime.date(2020, 11, 20), 'harshavardhini', 'ongoing', 103)
(2191044, 28, 'f', 34, datetime.date(2020, 11, 20), 'namrata', 'ongoing', 101)
(2191048, 39, 'm', 39, datetime.date(2020, 11, 23), 'firoz', 'ongoing', 102)
(2191006, 45, 'f', 63, datetime.date(2020, 11, 20), 'sharmila', 'ongoing', 104)
(2191046, 70, 'm', 44, datetime.date(2020, 11, 20), 'narendar', 'dead', 101)
(2191054, 23, 'm', 41, datetime.date(2020, 11, 21), 'varun', 'recovered', 101)

In [8]:
```

LSP Python: ready    conda: base /Python 3.8.3    Line 387, Col 1    UTF-8    LF    RW    Mem 68%    17:07

## 9)COVID INITIATIVES:

welcome user! COVID SPECIALITY UNIT

JOIN US IN OUR BATTLE AGAINST THE NOVEL CORONAVIRUS

1.DAILY CASES

2.DAILY COVID DEATHS

3.DAILY RECOVERIES

4.TOTAL CASES IN OUR HOSPITAL

5.TOTAL DEATHS IN OUR HOSPITAL DUE TO COVID

6.TOTAL RECOVERIES IN OUR HOSPITAL

7.ANALYSE COVID DATA

8.SEE ALL COVID PATIENTS' DATA

9.OUR COVID INITIATIVES

10.AVERAGE CASES PER DAY

11.AVERAGE DEATHS PER DAY

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice9

OUR COVID INITIATIVES!

('CB-NAAT', 'REJUVENATE Hospitals takes COVID-19 testing to a whole new level with CB-NAAT By embracing the Cartridge Based Nucleic Acid Amplification Testing (CB-NAAT) method,REJUVENATE Hospitals promises overall patient satisfaction, thanks to 100% accurate test reports that are generated within just 2 hours')

## 10)AVERAGE CASES:

- 10.AVERAGE CASES PER DAY
- 11.AVERAGE DEATHS PER DAY
- 12.AVERAGE RECOVERIES PER DAY
- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice10

10.AVERAGE CASES PER DAY

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

5.6667

## 11)AVERAGE DEATHS:

- 11.AVERAGE DEATHS PER DAY
- 12.AVERAGE RECOVERIES PER DAY
- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice11

11.AVERAGE DEATHS PER DAY

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

1.3333

## 12)AVERAGE RECOVERIES:

.....

- 10.AVERAGE CASES PER DAY
- 11.AVERAGE DEATHS PER DAY
- 12.AVERAGE RECOVERIES PER DAY
- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice12

12.AVERAGE RECOVERIES PER DAY

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

1.6667

## 13) MOST CASES IN A DAY:

- 11. AVERAGE DEATHS PER DAY
- 12. AVERAGE RECOVERIES PER DAY
- 13. most cases
- 14. most deaths
- 15. most recoveries
- 16. least cases
- 17. least deaths
- 18. least recoveries

enter choice13

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

## 14.MOST DEATHS:

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice14

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

3

## 15) MOST RECOVERIES:

12. AVERAGE RECOVERIES PER DAY

13. most cases

14. most deaths

15. most recoveries

16. least cases

17. least deaths

18. least recoveries

enter choice15

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

3

16)LEAST CASES IN A DAY:

12.AVERAGE RECOVERIES PER DAY

13.most cases

14.most deaths

15.most recoveries

16.least cases

17.least deaths

18.least recoveries

enter choice16

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

1

## 17)LEAST DEATHS IN A DAY:

- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice17

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

0

## 18)LEAST RECOVERIES:

- 13.most cases
- 14.most deaths
- 15.most recoveries
- 16.least cases
- 17.least deaths
- 18.least recoveries

enter choice18

enter no. of dates' data to be compiled3

enter date2020.11.20

enter date2020.11.21

enter date2020.11.23

0

## **MODULE 2 - ADMIN:**

### **1. ADMIN:**

Welcome user !

REJUVENATE HOSPITALS

HOSPITAL MANAGEMENT SYSTEM

LEADERS IN HEALTHCARE ADMINISTRATION

---

1.CONTINUE AS ADMIN

2.CONTINUE AS DOCTOR

3.COVID SPECIALITY UNIT

enter choice1

enter admin password:1234

WELCOME ADMIN

REJUVENATE HOSPITALS

1.PATIENT MODULE

2.DOCTOR MODULE

3.WARDS MODULE

4.REJUVENATE BLOOD BANK MODULE

5.COVID MODULE

6.CHANGE YOUR PASSWORD

## 2.PHARMACY:

### A) DISPLAY STOCK:

```
enter choice1

enter admin password:1234
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice7
WELCOME TO BIOHEAL PHARMACY
MENU
1.Show Stock
2.Add medicine into stock
3.Update medicine stock
4.Delete medicine stock

Enter your choice:1
(1, 'CROGIN', 'GSK_HEALTH_CONSUMER', 30, 200, datetime.date(2020, 7, 17), datetime.date(2021, 7, 27))
(2, 'FORACORT', 'CIPLA', 304, 350, datetime.date(2020, 8, 24), datetime.date(2021, 8, 24))
(3, 'LEVOLIN', 'CIPLA', 192, 230, datetime.date(2020, 3, 24), datetime.date(2022, 2, 24))
(4, 'BRUFFIN', 'ABBOTT', 9, 150, datetime.date(2019, 4, 30), datetime.date(2021, 4, 25))
(5, 'COMBIFLAM', 'SANOFI', 34, 400, datetime.date(2020, 1, 16), datetime.date(2021, 1, 15))
(6, 'BIFILAC', 'TABLETS_IMDIA_LIMITED', 98, 379, datetime.date(2020, 11, 25), datetime.date(2021, 11, 24))
(7, 'THYRONORM', 'ABBOTT', 148, 468, datetime.date(2019, 12, 24), datetime.date(2021, 12, 23))
(8, 'MEFTAL_SPAS', 'BLUE_CROSS', 42, 478, datetime.date(2020, 4, 16), datetime.date(2021, 4, 16))
(9, 'DOLO650', 'MICRO_LABS_LIMITED', 30, 400, datetime.date(2020, 3, 30), datetime.date(2021, 3, 29))
(10, 'P500', 'APEX', 15, 342, datetime.date(2020, 1, 24), datetime.date(2021, 1, 23))
(11, 'ibugesic', 'cipla', 500, 21, datetime.date(2020, 11, 20), datetime.date(2022, 11, 22))
```

## B) INSERTION OF RECORD:

```
Console 16/A @ > /Users/rajakumarisureshbabu
3.Update medicine stock
4.Delete medicine stock

Enter your choice:1
(1, 'CROCIN', 'GSK_HEALTH_CONSUMER', 30, 200, datetime.date(2020, 7, 17), datetime.date(2021, 7, 27))
(2, 'FORACORT', 'CIPLA', 304, 350, datetime.date(2020, 8, 24), datetime.date(2021, 8, 24))
(3, 'LEVOLIN', 'CIPLA', 192, 230, datetime.date(2020, 3, 24), datetime.date(2022, 2, 24))
(4, 'BRUFIN', 'ABBOTT', 9, 150, datetime.date(2019, 4, 30), datetime.date(2021, 4, 25))
(5, 'COMBIFLAM', 'SANOFI', 34, 400, datetime.date(2020, 1, 16), datetime.date(2021, 1, 15))
(6, 'BIFILAC', 'TABLETS_IMDIA_LIMITED', 98, 379, datetime.date(2020, 11, 25), datetime.date(2021, 11, 24))
(7, 'THYRONORM', 'ABBOTT', 148, 468, datetime.date(2019, 12, 24), datetime.date(2021, 12, 23))
(8, 'MEFTAL_SPAS', 'BLUE_CROSS', 42, 478, datetime.date(2020, 4, 16), datetime.date(2021, 4, 16))
(9, 'DOLOG650', 'MICRO_LABS_LIMITED', 30, 400, datetime.date(2020, 3, 30), datetime.date(2021, 3, 29))
(10, 'P500', 'APEX', 15, 342, datetime.date(2020, 1, 24), datetime.date(2021, 1, 23))
(11, 'ibugesic', 'cipla', 500, 21, datetime.date(2020, 11, 20), datetime.date(2022, 11, 22))

Enter y to continue in the pharmacy module:

Enter your choice:2

Enter the id of the medicine:12

Enter the name of the medicine:fwwf

Enter the pharma company:fwf

Enter the price:33

Enter the number of medicines available:33

Enter the mfg date:2020.11.11

enter the expiry date:2021.11.11

Data inserted successfully
```

```
Console 16/A @ > /Users/rajakumarisureshbabu
LSP Python: ready conda: base /Python 3.8.3 Line 1085, Col 49 UTF-8 LF RW Mem 69% 20:57
Enter y to continue in the pharmacy module:

Enter your choice:2

Enter the id of the medicine:12

Enter the name of the medicine:fwwf

Enter the pharma company:fwf

Enter the price:33

Enter the number of medicines available:33

Enter the mfg date:2020.11.11

enter the expiry date:2021.11.11

Data inserted successfully

Enter y to continue in the pharmacy module:

Enter your choice:1
(1, 'CROCIN', 'GSK_HEALTH_CONSUMER', 30, 200, datetime.date(2020, 7, 17), datetime.date(2021, 7, 27))
(2, 'FORACORT', 'CIPLA', 304, 350, datetime.date(2020, 8, 24), datetime.date(2021, 8, 24))
(3, 'LEVOLIN', 'CIPLA', 192, 230, datetime.date(2020, 3, 24), datetime.date(2022, 2, 24))
(4, 'BRUFIN', 'ABBOTT', 9, 150, datetime.date(2019, 4, 30), datetime.date(2021, 4, 25))
(5, 'COMBIFLAM', 'SANOFI', 34, 400, datetime.date(2020, 1, 16), datetime.date(2021, 1, 15))
(6, 'BIFILAC', 'TABLETS_IMDIA_LIMITED', 98, 379, datetime.date(2020, 11, 25), datetime.date(2021, 11, 24))
(7, 'THYRONORM', 'ABBOTT', 148, 468, datetime.date(2019, 12, 24), datetime.date(2021, 12, 23))
(8, 'MEFTAL_SPAS', 'BLUE_CROSS', 42, 478, datetime.date(2020, 4, 16), datetime.date(2021, 4, 16))
(9, 'DOLOG650', 'MICRO_LABS_LIMITED', 30, 400, datetime.date(2020, 3, 30), datetime.date(2021, 3, 29))
(10, 'P500', 'APEX', 15, 342, datetime.date(2020, 1, 24), datetime.date(2021, 1, 23))
(11, 'ibugesic', 'cipla', 500, 21, datetime.date(2020, 11, 20), datetime.date(2022, 11, 22))
(12, 'fwwf', 'fwf', 33, 33, datetime.date(2020, 11, 11), datetime.date(2021, 11, 11))
```

## C)UPDATION OF RECORD:

MENU

- 1.Show Stock
- 2.Add medicine into stock
- 3.Update medicine stock
- 4.Delete medicine stock

Enter your choice:3

Enter the id of the medicine you want to update:12

Enter the name of the medicine: ffwfwf

Enter the name of the pharma company: fwf

Enter the price:33

Enter the number of medicines available:1

Enter the mfg date:2020.11.11

enter the expiry date:2021.11.11

Data updated successfully

Data updated successfully

Enter y to continue in the pharmacy module:y

Enter your choice:1

- (1, 'CROGIN', 'GSK\_HEALTH\_CONSUMER', 30, 200, datetime.date(2020, 7, 17), datetime.date(2021, 7, 27))
- (2, 'FORACORT', 'CIPLA', 304, 350, datetime.date(2020, 8, 24), datetime.date(2021, 8, 24))
- (3, 'LEVOLIN', 'CIPLA', 192, 230, datetime.date(2020, 3, 24), datetime.date(2022, 2, 24))
- (4, 'BRUFFIN', 'ABBOTT', 9, 150, datetime.date(2019, 4, 30), datetime.date(2021, 4, 25))
- (5, 'COMBIFLAM', 'SANOFI', 34, 400, datetime.date(2020, 1, 16), datetime.date(2021, 1, 15))
- (6, 'BIFILAC', 'TABLETS\_IMDIA\_LIMITED', 98, 379, datetime.date(2020, 11, 25), datetime.date(2021, 11, 24))
- (7, 'THYRONORM', 'ABBOTT', 148, 468, datetime.date(2019, 12, 24), datetime.date(2021, 12, 23))
- (8, 'MEFTAL\_SPAS', 'BLUE\_CROSS', 42, 478, datetime.date(2020, 4, 16), datetime.date(2021, 4, 16))
- (9, 'DOL0650', 'MICRO\_LABS\_LIMITED', 30, 400, datetime.date(2020, 3, 30), datetime.date(2021, 3, 29))
- (10, 'P500', 'APEX', 15, 342, datetime.date(2020, 1, 24), datetime.date(2021, 1, 23))
- (11, 'ibugesic', 'cipla', 500, 21, datetime.date(2020, 11, 20), datetime.date(2022, 11, 22))
- (12, 'ffwwf', 'fwf', 33, 1, datetime.date(2020, 11, 11), datetime.date(2021, 11, 11))

## D)DELETION OF RECORD:

The screenshot shows a terminal window with a light gray background and a dark gray header bar. The header bar contains various icons for file operations like copy, paste, and search, followed by the path '/Users/rajakumarisureshbabu' and some status indicators.

The main area of the terminal displays the following text:

```
Enter y to continue in the pharmacy module:  
Enter your choice:4  
Enter the id of the medicine you want to delete:12  
Record deleted  
Enter y to continue in the pharmacy module:  
Enter your choice:1  
(1, 'CROCIN', 'GSK_HEALTH_CONSUMER', 30, 200, datetime.date(2020, 7, 17), datetime.date(2021, 7, 27))  
(2, 'FORACORT', 'CIPLA', 304, 350, datetime.date(2020, 8, 24), datetime.date(2021, 8, 24))  
(3, 'LEVOLIN', 'CIPLA', 192, 230, datetime.date(2020, 3, 24), datetime.date(2022, 2, 24))  
(4, 'BRUFIN', 'ABBOTT', 9, 150, datetime.date(2019, 4, 30), datetime.date(2021, 4, 25))  
(5, 'COMBIFLAM', 'SANOFI', 34, 400, datetime.date(2020, 1, 16), datetime.date(2021, 1, 15))  
(6, 'BIFILAC', 'TABLETS_IMDIA_LIMITED', 98, 379, datetime.date(2020, 11, 25), datetime.date(2021, 11, 24))  
(7, 'THYRONORM', 'ABBOTT', 148, 468, datetime.date(2019, 12, 24), datetime.date(2021, 12, 23))  
(8, 'MEFTAL_SPAS', 'BLUE_CROSS', 42, 478, datetime.date(2020, 4, 16), datetime.date(2021, 4, 16))  
(9, 'DOLO650', 'MICRO_LABS_LIMITED', 30, 400, datetime.date(2020, 3, 30), datetime.date(2021, 3, 29))  
(10, 'P500', 'APEX', 15, 342, datetime.date(2020, 1, 24), datetime.date(2021, 1, 23))  
(11, 'ibugesic', 'cipla', 500, 21, datetime.date(2020, 11, 20), datetime.date(2022, 11, 22))  
Enter y to continue in the pharmacy module:  
↳ LSP Python: ready    conda.base/Python 3.8.3    Line 1085, Col 49    UTF-8    LF    RW    Mem 69%    21:00
```

### 3)WARDS:

#### A) DISPLAY WARDS:

```
enter choice1
enter admin password:1234
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL
enter your choice3
3.wards module
submodules
1.ADD NEW WARD
2.ROOM INFORMATION and AVAILABILITY
3.UPDATE WARD
enter choice2
('a', 1, 28, 'occupied', 878)
('a', 30, 28, 'free', 626)
('a', 68, 30, 'occupied', 8378)
('a', 39, 39, 'occupied', 3838)
('b', 44, 33, 'free', 666)
('b', 66, 33, 'occupied', 989)
('b', 44, 32, 'free', 983)
('a', 54, 32, 'free', 940)
('b', 54, 30, 'occupied', 948)
('a', 38, 54, 'occupied', 987)

enter y to continue|
```

## B)INSERT WARD:

```
Console 6/A
3.wurus module
submodules
1.ADD NEW WARD
2.ROOM INFORMATION and AVAILABILITY
3.UPDATE WARD
enter choice1
('a', 1, 28, 'occupied', 878)
('a', 30, 28, 'free', 626)
('a', 68, 30, 'occupied', 8378)
('a', 39, 39, 'occupied', 3838)
('b', 44, 33, 'free', 666)
('b', 66, 33, 'occupied', 989)
('b', 44, 32, 'free', 983)
('a', 54, 32, 'free', 940)
('b', 54, 30, 'occupied', 948)
('a', 38, 54, 'occupied', 987)

enter building:b
enter room no:13
enter no. of beds:45
enter room status:free
enter daily charge for room:666
data inserted successfully

('a', 1, 28, 'occupied', 878)
('a', 30, 28, 'free', 626)
('a', 68, 30, 'occupied', 8378)
('a', 39, 39, 'occupied', 3838)
('b', 44, 33, 'free', 666)
('b', 66, 33, 'occupied', 989)
('b', 44, 32, 'free', 983)
('a', 54, 32, 'free', 940)
('b', 54, 30, 'occupied', 948)
('a', 38, 54, 'occupied', 987)
('b', 13, 45, 'free', 666)
enter y to continue|
```

### C) WARD UPDATION:

```
Console 6/A /Users/rajakumarisureshbabu
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL
enter your choice3
3.wards module
submodules
1.ADD NEW WARD
2.ROOM INFORMATION and AVAILABILITY
3.UPDATE WARD
enter choice3
('a', 1, 28, 'occupied', 878)
('a', 30, 28, 'free', 626)
('a', 68, 30, 'occupied', 8378)
('a', 39, 39, 'occupied', 3838)
('b', 44, 33, 'free', 666)
('b', 66, 33, 'occupied', 989)
('b', 44, 32, 'free', 983)
('a', 54, 32, 'free', 940)
('b', 54, 30, 'occupied', 948)
('a', 38, 54, 'occupied', 987)
('b', 13, 45, 'free', 666)

enter building:b
enter room no:13
enter no. of beds:45
enter room status:occupied
enter daily charge for room:666

records updated successfully
Do You Want To update More Records
Enter y to update furthern
^_
```

## 4)NON-COVID PATIENTS:

### A) DISPLAY PATIENTS' DETAILS:



```
Console 9/A
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice1
PATIENT'S MODULE
1.NEW PATIENT
2.PATIENTS RECORDS
3.UPDATE PATIENTS RECORD
4.DISCHARGE OF PATIENT

Enter your choice:2
HERE IS THE LIST OF PATIENTS RECORDS:

('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagonal alley', 'heart failure and arrhythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskruti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ', 'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkhkh', 'Alzheimer's disease', '82', 45, 'ghdtdhj', 203)
('thamizh', 1191065, datetime.date(2020, 11, 7), datetime.date(2020, 11, 25), 'f', 66, 456, 77, 6666, 'dfgfdgfhggkgkj', 'pneumonia', 'vip', 66, 'giuh', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihoungomc', 'Nephritis', 'general', 87, 'hjhjjhh', 202)

enter y to continue
```

## B) INSERT PATIENT RECORD:

```
enter your choice1
PATIENT'S MODULE
1.NEW PATIENT
2.PATIENTS RECORDS
3.UPDATE PATIENTS RECORD
4.DISCHARGE OF PATIENT

Enter your choice:1
('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagon alley', 'heart failure and arrythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskruti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ', 'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkhkhjh', 'Alzheimer's disease', '82', 45, 'ghdtdhj', 203)
('thamizh', 1191065, datetime.date(2020, 11, 7), datetime.date(2020, 11, 25), 'f', 66, 456, 77, 6666, 'dfgfdgfhggkgkj', 'pneumonia', 'vip', 66, 'giuh', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc', 'Nephritis', 'general', 87, 'hjhjjhh', 202)

Enter the number of records you want to add: 1

Enter patient's name: ashmitha

Enter patient id:1191065

Enter the date of joining the hospital:2020.11.02
```

The screenshot shows a Jupyter Notebook interface with a light gray header bar. The header bar contains various icons for file operations like opening, saving, and running cells, along with the path '/Users/rajakumarisureshbabu' and a red close button. Below the header is a toolbar with a 'Console 12/A' tab, a search bar, and other notebook-related icons.

The main content area displays a series of text inputs and their corresponding values:

- Enter patient's name: ashmitha
- Enter patient id:1191065
- Enter the date of joining the hospital:2020.11.02
- enter the date of discharge2020.11.29
- Enter patient's gender' : f
- Enter patient's age: 36
- Enter patient's aadhar number: 27826
- Enter patient's health insurance number: 57575
- Enter patient's contact number ' : 57656
- Enter patient's address ': fgghjgjhg
- Enter patient's disease: fggjgg
- Enter room type: vip
- Enter the room number : 27
- Enter email id of the patient : ghhkjhkj
- Enter doctor id assigned : 201
- Data inserted successfully
- INSERTION SUCCESSFUL
- enter y to continue|

At the bottom of the interface, there is a footer bar with the text 'LSP Python: ready', 'conda: base (Python 3.8.3)', 'Line 25, Col 1', 'UTF-8 GUESSED', 'LF RW Mem 68% 22.01'.

The screenshot shows a terminal window titled "Console 12/A" with the path "/Users/rajakumarisureshbabu". The window contains the following text:

```
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice1
PATIENT'S MODULE
1.NEW PATIENT
2.PATIENTS RECORDS
3.UPDATE PATIENTS RECORD
4.DISCHARGE OF PATIENT

Enter your choice:2
HERE IS THE LIST OF PATIENTS RECORDS:

('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagon alley', 'heart failure and arrythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskruti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ', 'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkhkhjh', 'Alzheimer's disease', '82', 45, 'ghdtdhj', 203)
('thamizh', 1191065, datetime.date(2020, 11, 7), datetime.date(2020, 11, 25), 'f', 66, 456, 77, 6666, 'dfgfdgfhggkgkj', 'pneumonia', 'vip', 66, 'giuh', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc', 'Nephritis', 'general', 87, 'hjhjjhh', 202)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'fgghjgjhg', 'fggjgg', 'vip', 27, 'ghhkjhkj', 201)

enter y to continue
```

## C)UPDATION OF RECORDS:

The screenshot shows a Jupyter Notebook interface with a toolbar at the top and a code cell containing Python code. The code defines a tuple with patient details: name ('ashmitha'), ID (1191065), joining date (2020-11-2), discharge date (2020-11-29), gender ('f'), age (36), Aadhar number (27826), insurance number (57575), contact number (57656), address ('freddy street ram bhavan'), disease ('dengue'), room type ('vip'), room number (20), and email ('jhgjghjhj').

```
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'fgghjgjhg', 'fggjgg', 'vip', 27, 'ghhkjhkj', 201)
```

Enter the number of records you want to update:1

Enter patient's name: ashmitha

Enter patient id:1191065

Enter the date of joining the hospital:2020.11.2

enter date of discharge2020.11.29

Enter patient's gender' : f

Enter patient's age: 36

Enter patient's aadhar number: 27826

Enter patient's health insurance number: 57575

Enter patient's contact number ' : 57656

Enter patient's address ': freddy street ram bhavan

Enter patient's disease: dengue

Enter room type: vip

Enter the room number : 20

Enter email id of the patient : jhgjghjhj

Enter doctor id assigned : 201

Data updated successfully

LSP Python: ready conda: base (Python 3.8.3) Line 106, Col 21 UTF-8 LF RW Mem 69% 22.07

The screenshot shows a terminal window with a light gray background and a dark gray header bar. The header bar contains various icons for file operations like copy, paste, and search, followed by the path '/Users/rajakumarisureshbabu' and some status indicators. Below the header is a tab bar with 'Console 13/A'. The main area of the terminal contains the following text:

```
Enter patient's address : freddy street ram bhavan
Enter patient's disease: dengue
Enter room type: vip
Enter the room number : 20
Enter email id of the patient : jhgjghjhj
Enter doctor id assigned : 201
Data updated successfully
UPDATION SUCCESSFUL
('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagon alley', 'heart failure and arrythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskruti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ', 'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkhkhkh', 'Alzheimer's disease', '82', 45, 'ghdtdhj', 203)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'freddy street ram bhavan', 'dengue', 'vip', 20, 'jhgjghjhj', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc', 'Nephritis', 'general', 87, 'hjhjjhh', 202)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'freddy street ram bhavan', 'dengue', 'vip', 20, 'jhgjghjhj', 201)
```

## D)DELETION OF PATIENT RECORD:

```
enter your choice1
PATIENT'S MODULE
1.NEW PATIENT
2.PATIENTS RECORDS
3.UPDATE PATIENTS RECORD
4.DISCHARGE OF PATIENT

Enter your choice:4
('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagon alley', 'heart failure and arrythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskruti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ', 'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkjhkhlh', 'Alzheimer's disease', '82', 45, 'ghdtdhj', 203)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'freddy street ram bhavan', 'dengue', 'vip', 20, 'jhgjghjhj', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc', 'Nephritis', 'general', 87, 'hjhjjhh', 202)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'freddy street ram bhavan', 'dengue', 'vip', 20, 'jhgjghjhj', 201)

Enter the number of records you want to delete:1

Enter patient's name to be deleted: ashmitha

Data deleted successfully
DELETION SUCCESSFUL
('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
```

LSP Python: ready conda: base /Python 3.8.3 Line 106, Col 21 UTF-8 LF RW Mem 68% 22.07

```
'dengue', 'vip', 20, 'jhgjghjhj', 201)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc',
'Nephritis', 'general', 87, 'hjhjjhh', 202)
('ashmitha', 1191065, datetime.date(2020, 11, 2), datetime.date(2020, 11, 29), 'f', 36, 27826, 57575, 57656, 'freddy street ram bhavan',
'dengue', 'vip', 20, 'jhgjghjhj', 201)

Enter the number of records you want to delete:1

Enter patient's name to be deleted: ashmitha

Data deleted successfully
DELETION SUCCESSFUL
('AMITESH ', 1191001, datetime.date(2020, 11, 1), datetime.date(2020, 11, 14), 'M', 23, 123, 123, 123456789, '10 DOWNING STREET , SANSAD
BHAVAN', 'INFLUENZA', 'VIP', 12, 'AMITACE@GMAIL.COM', 201)
('varun ', 1191002, datetime.date(2020, 11, 2), datetime.date(2020, 11, 18), 'm', 54, 142, 144, 1998283, 'manhattan diagonal alley', 'heart
failure and arrythmias', 'general', 13, 'vrn32@gmail.com', 209)
('sanghavi', 1191089, datetime.date(2020, 11, 3), datetime.date(2020, 11, 10), 'f', 28, 231, 211, 76529827, 'highrise paradise,buckingham
street,jafferkhanpet', 'sepsis', 'vip', 33, 'sanhv@hotmail.com', 209)
('swastika ', 1191008, datetime.date(2020, 11, 4), datetime.date(2020, 11, 14), 'f', 37, 119, 181, 12893649, 'Bermuda Haven,Ellingham
street,serpeck nagar', 'subarachnoid hemorrhage', 'general', 65, 'swasti@yahoo.com', 204)
('sanskriti ', 1191088, datetime.date(2020, 11, 2), datetime.date(2020, 11, 30), 'f', 74, 233, 321, 332224, 'finland corner, koyembedu
nagar', 'seizure', 'vip', 29, 'sanspurvi@gmail.com', 201)
('swapnil rathore', 1191089, datetime.date(2020, 11, 6), datetime.date(2020, 11, 9), 'm', 32, 1943, 228, 28392, 'manohar vikas bhavan ',
'anemia', 'vip', 38, 'swapthore@gmail.com', 201)
('dewaker', 1191072, datetime.date(2020, 11, 6), datetime.date(2020, 11, 23), 'm', 94, 43441, 2425, 2466, 'ghghgjbjkjhkhlh', 'Alzheimer's
disease', '82', 45, 'ghdtdhj', 203)
('gookle', 1191092, datetime.date(2020, 11, 5), datetime.date(2020, 11, 30), 'm', 65, 3636, 33234, 33782, 'fdgehjaihounngomc',
'Nephritis', 'general', 87, 'hjhjjhh', 202)

enter y to continue
```

In [2]: |

## 5) DOCTOR MODULE UNDER ADMIN:

### A) DISPLAY DOCTOR:

```
enter choice1

enter admin password:1234
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice2
Here is the list of doctors:

('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjh@gmail.com', 638180491, 'fgfkkgkghkgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkkgkghkgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkghkhgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkghkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')

LSP Python: ready  conda: base (Python 3.8.3)  Line 106, Col 21  UTF-8  LF  RW  Mem 69%  22:11
```

**B) INSERT DOCTOR RECORD:**

```
File Edit View Insert Cell Kernel Help /Users/rajakumarisureshbabu
Console 15/A
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfgkkgkhgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkhkgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')

Enter the number of records you want to add 1

Enter the doctor's name: george

Enter the doctor's id205

Enter the gender of doctorm

Enter the time of consultation: 01:00:00

Enter the specialization: eye

Enter the experience: 19

Enter the fee of consultation: 999

Enter the doctor's email id: georgekutty@gmail.com

Enter doctor's phone number: 893676

Enter the doctor's address: farmhouse,idukki-12

Date inserted successfully!

DOCTOR'S DETAIL ADDED SUCCESSFULLY!

enter y to continue
```

The screenshot shows a terminal window titled "Console 15/A" with the path "/Users/rajakumarisureshbabu". The window contains the following text:

```
7.REJOVENATE PHARMACY
8.GENERATE BILL

enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice2
Here is the list of doctors:

('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkhgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkhgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkhgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
('george', 205, 'm', datetime.timedelta(seconds=3600), 'eye', 19, 999, 'georgekutty@gmail.com', 893676, 'farmhouse,idukki-12')

enter y to continue
```

## C)UPDATION:

```
3.COVID SPECIALITY UNIT

enter choice1

enter admin password:1234
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice3
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhjikyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
('george', 205, 'm', datetime.timedelta(seconds=3600), 'eye', 19, 999, 'georgekutty@gmail.com', 893676, 'farmhouse,idukki-12')
```

The screenshot shows a Jupyter Notebook interface with a console tab open. The user has entered several pieces of data for updating a database:

- Enter the number of records you want to update:1
- Enter the doctor's name: george
- Enter the doctor's id:205
- Enter the gender of doctor:m
- Enter the time of consultation: 01:00:00
- Enter the specialization: eye
- Enter the experience: 21
- Enter the fee for consultation:1000
- Enter the doctor's email id: fi@gmail.com
- Enter doctor's phone number: 78668
- Enter the doctor's address: hhjhhfghg

Data updated successfully!

UPDATED SUCCESSFULLY!

```
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfgkkgkghkhgkyfkfy')  
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkhgkyfkfy')  
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkhgkyfkfy')  
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')  
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')  
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')  
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')  
('george', 205, 'm', datetime.timedelta(seconds=3600), 'eye', 21, 1000, 'fi@gmail.com', 78668, 'hhjhhfghg')
```

enter y to continue

## D)DELETE DOCTOR:

```
enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice4
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkhgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkhgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
('george', 205, 'm', datetime.timedelta(seconds=3600), 'eye', 21, 1000, 'fi@gmail.com', 78668, 'hhjhhfghg')

Enter the number of records you want to delete1

Enter doctor's name to be deleted: george

Data deleted successfully!
DELETED SUCCESSFULLY!
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkhgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkhgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')

enter y to continue
```

## E) APPOINTMENT INSERTION:



enter your choice2  
doctor module  
submodules  
1.ADD DOCTOR  
2.LIST OF DOCTORS  
3.UPDATE DOCTOR RECORD  
4.REMOVAL OF A DOCTOR RECORD  
5.APPOINTMENTS

enter your choice5

```
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkgkgkhgkyfkfy')  
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghhgkyfkfy')  
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')  
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')  
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')  
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')  
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
```

1.Appointment Addition  
2.Appointment Updation  
3.Appointment Display  
4.Appointment Deletion

Enter your choice: 1

Enter number of appointments3

Enter the doctor's id:201

Enter the patient's id:2191007

Enter the consultation time:11:00:00

Appointments added successfully!

LSP Python: ready    conda: base / Python 3.8.3    Line 825, Col 47    UTF-8    LF    RW    Mem 67%    22:42

The screenshot shows a terminal window with a light gray background and a dark gray header bar. The header bar contains various icons on the left, the path '/Users/rajakumarisureshbabu' in the center, and three small icons on the right. Below the header is a title bar with a 'Console 18/A' tab and a close button. The main area of the terminal displays the following text:

```
Enter the doctor's id:101
Enter the patient's id:1191078
Enter the consultation time:12:00:00
Appointments added successfully!

Enter the doctor's id:204
Enter the patient's id:1191009
Enter the consultation time:03:00:00
Appointments added successfully!
Appointment added successfully
enter y to continue|
```

At the bottom of the terminal window, there is a footer bar with several status indicators: 'LSP Python: ready', 'conda: base /Python 3.8.3', 'Line 825, Col 47', 'UTF-8', 'LF', 'RW', 'Mem 68%', and '22:42'.

## F) APPOINTMENT DISPLAY:

```
enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice5
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfgkghkghgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkghgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkghgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
1.Appointment Addition
2.Appointment Updation
3.Appointment Display
4.Appointment Deletion

Enter your choice: 3
The appointments are:
(201, 2191007, datetime.timedelta(seconds=39600))
(101, 1191078, datetime.timedelta(seconds=43200))
(204, 1191009, datetime.timedelta(seconds=10800))

enter y to continue

LSP Python: ready  conda-base /Python 3.8.3  Line 825, Col 47  UTF-8  LF  RW  Mem 69%  22.43
```

## G) APPOINTMENT DELETE:

```
enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice5
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkgkgkhgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkhgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkhgjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
1.Appointment Addition
2.Appointment Updation
3.Appointment Display
4.Appointment Deletion

Enter your choice: 4

Do you want to delete an appointmenty

Enter the patient's id to be deleted204

Appointment deleted successfully!
Appointment deleted successfully!

enter y to continuey
WELCOME ADMIN
REJUVENATE HOSPITALS
1 PATIENT MODULF
```

```
Console 19/A /Users/rajkumarisureshbabu
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice2
doctor module
submodules
1.ADD DOCTOR
2.LIST OF DOCTORS
3.UPDATE DOCTOR RECORD
4.REMOVAL OF A DOCTOR RECORD
5.APPOINTMENTS

enter your choice5
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkgkyfkfy')
('richard', 102, 'm', datetime.timedelta(seconds=21600), 'neuro', 13, 500, 'ggjg@gmail.com', 635180491, 'fkgkkghkgkyfkfy')
('kavita', 103, 'f', datetime.timedelta(seconds=7200), 'gynaec', 13, 500, 'ggj@gmail.com', 63510491, 'fkgkhkgkyfkfy')
('karen', 104, 'f', datetime.timedelta(seconds=28800), 'gynaec', 13, 500, 'gg@gmail.com', 6350491, 'fkgkkhgkyfkfy')
('ruben', 204, 'm', datetime.timedelta(seconds=3600), 'physio', 13, 500, 'g@gmail.com', 635091, 'fkkhgkyfkfy')
('paul', 201, 'm', datetime.timedelta(seconds=68400), 'heart', 18, 500, 'guuj@gmail.com', 6354091, 'fkkh gjjkyfkfy')
('daisy', 202, 'f', datetime.timedelta(seconds=79200), 'heart', 18, 500, 'guu@gmail.com', 635491, 'fkkyfkfy')
1.Appointment Addition
2.Appointment Updation
3.Appointment Display
4.Appointment Deletion

Enter your choice: 3
The appointments are:
(201, 2191007, datetime.timedelta(seconds=39600))
(101, 1191078, datetime.timedelta(seconds=43200))

enter y to continue|
```

## 6.ADMIN PASSWORD CHANGE:

The screenshot shows a Jupyter Notebook interface with a Python 3.8.1 kernel. The code cell In [1] contains the command `runfile('/Users/rajakumarisureshbabu/untitled20.py', wdir='/Users/rajakumarisureshbabu')`. The output shows the program's interaction with the user:

```
Python 3.8.1 (default, Jan  8 2020, 16:15:59)
Type "copyright", "credits" or "license" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]: runfile('/Users/rajakumarisureshbabu/untitled20.py', wdir='/Users/rajakumarisureshbabu')
enter admin password:123
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL
enter your choice6
change passwordenter old password123

enter new password1234
confirm new admin password1234
password change was successful
enter y to continuey
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL
enter your choice|
```

The status bar at the bottom indicates the following details: LSP Python: ready, conda base / Python 3.8.3, Line 387, Col 159, UTF-8, LF, RW, Mem 67%, 20.08.

```
In [1]: runfile('/Users/rajakumarisureshbabu/untitled20.py', wdir='/Users/rajakumarisureshbabu')

enter admin password:123
password incorrect
```

## 7.BILL GENERATOR:

```
Python 3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:37:50) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Venkata Sudiksha/AppData/Local/Programs/Python/Python38/bill.py
REJUVENATE HOSPITAL
PATIENT BILL
Please enter your patient id : 1
('RAHUL', 1, datetime.date(2020, 3, 31), datetime.date(2020, 4, 2), 'MALE', 36, 1324463, 423756, 985432651, 'SKYCITY_TOWERS', 'MALARIA', 'GENERAL_WARD', 339, 'rahul@gmail.com', 1)
Please verify your details
TYPE OF TREATMENT:
1.CONSULTATION
2.SURGERY
Enter your type of treatment:2
ENTER YOUR ROOM TYPE (GENERAL/VIP/COVID) : general
ENTER THE BILL NO :23
ENTER THE NUMBER OF BEDS USED:2
ENTER THE NUMBER OF THEATERS USED :1
ENTER THE NUMBER OF ANAESTHESIA WAS GIVEN TO THE PATIENT: 2
ENTER THE NUMBER OF EQUIPMENTS USED: 1
data inserted successfully
PAYMENT MODES:
1.CREDIT CARD
2.CASH
>>>|
```

## BILL STORED IN MYSQL:

7  
8

The screenshot shows a MySQL database result grid with the following data:

SLNO	SERVICES	QTY	RATE	AMOUNT
1	ADMISSION_CHARGES	1	200	200
2	BED_CHARGES	2	2500	5000
3	SURGEON_CHARGES	1	45000	45000
4	ASST_SURGEON_CHARGES	1	25000	25000
5	THEATRE_CHARGES	1	6000	6000
6	ANAESTHETIST_FEES	2	12000	24000
7	SERVICES	1	22164	22164
8	EQUIPMENT	1	18400	18400
9	TOTAL	0	0	145764

# 8.BLOOD BANK:

## A) INSERTION:

```
enter admin password:123
WELCOME ADMIN
REJUVENATE HOSPITALS
1.PATIENT MODULE
2.DOCTOR MODULE
3.WARDS MODULE
4.REJUVENATE BLOOD BANK MODULE
5.COVID MODULE
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice4
4.REJUVENATE BLOOD BANK MODULE
/red heart icons/
1.insert into blood donors
2.change donor details
3.delete donor
4.update blood group details
5.see the list of eligible donors of this day

enter valid choice1

enter donor id1

enter father's namesf

enter mother's namedfs

enter last namesfdfss

enter gender of donorf

enter blood groupa+

enter birthdate of donor in yy-mm-dd2004.11.22
```

```
LSP Python: ready   conda: base /Python 3.8.3   Line 1, Col 1   UTF-8 GUESSED   LF   RW   Mem 66%   19:50

enter last namesfdfss

enter gender of donorf

enter blood groupa+

enter birthdate of donor in yy-mm-dd2004.11.22

enter addresshhkjh

enter cityhbjhb

enter date of donation(latest)2020.11.22

enter body temperature at the time of donation in fahrenheit99

enter pulse rate at donation67

enter blood pressure at donation56

enter weight44

enter haemoglobin count and specifications555

enter platelet count6666

enter hbsag666

enter HIV statusnil

enter malaria smear5235

enter hematocrit523523

enter phone no.87983793
```

## B) UPDATION:

The screenshot shows a terminal window titled "Console 4/A" with a path of "/Users/rajakumarisureshbabu". The window contains the following text output from a Python script:

```
enter your choice4
4.REJUVENATE BLOOD BANK MODULE
滴滴滴滴滴滴滴滴滴滴滴滴
1.insert into blood donors
2.change donor details
3.delete donor
4.update blood group details
5.see the list of eligible donors of this day

enter valid choice2

enter donor id6

enter father's namejhbjbjk

enter mother's namebjbb

enter last namehjhj

enter gender of donorf

enter blood groupa-

enter birthdate of donor in yy-mm-dd2001.22

enter addresshihi

enter cityhjbfhj

enter date of donation(latest)2020.11.22

enter body temperature at the time of donation in fahrenheit98

enter pulse rate at donation44
```

At the bottom of the terminal, there is a status bar with the following information:

LSP Python: ready conda: base (Python 3.8.3) Line 1, Col 1 UTF-8 GUESSED LF RW Mem 68% 19:58

C)DELETION:

ԱՐԱՐԱԿԱՆ ՀԱՅՈՒԹՅՈՒՆ ԵՎ ՀԱՅՈՒԹՅՈՒՆԻ ՏԻՄԱԳՈՐԾՈՒՄ

enter admin password:123

WELCOME ADMIN

## REJUVENATE HOSPITALS

## 1. PATIENT MODULE

## 2. DOCTOR MODULE

### 3. WARDS MODULE

#### 4. REJUVENATE BLOOD BANK MODULE

5. COVID MODULE

## 6 CHANGE YOUR PASSWORD

7 REJUVENATE PHARMACY

## 8 GENERATE RTLI

enter your choice4

#### **4. REJUVENATE BLOOD BANK MODULE**



### 1.insert into blood donors

## 2. change donor details

### 3. delete donor

#### 4. update blood group details

5. see the list of eligible donors of this day

enter valid choice3

enter donor id6

donor data deleted successfully

enter v to continue in bbmsv

enter donor id

# D)ELIGIBILITY FOR DONATION:

```
Console 6/A /Users/rajakumarisureshbabu
4.REJUVENATE BLOOD BANK MODULE
滴滴滴滴滴滴滴滴滴滴
1.insert into blood donors
2.change donor details
3.delete donor
4.update blood group details
5.see the list of eligible donors of this day

enter valid choice5
1.to view the standard GENERAL POLICY FOR BLOOD DONATION
2.COVID EXTRA RULES ON BLOOD DONATION SAFETY
3.elligible donors on a particular date satisying terms

enter your choice1
Must be at least 16 years old (16- and 17-year-olds must bring a signed permission form from a parent or guardian, if required by state or school
Weigh at least 110 pounds. Certain height/weight criteria may apply for donors 22 years old or younger.
Be in good general health.
For your safety and to ensure a positive donation experience, make sure you eat within two hours ahead of your donation. Drink plenty of water that day and 24 to 48 hours beforehand. Feel free to help yourself to something to eat and drink in our refreshment area.
Bring your ID(something with your name and one of the following: date of birth, donor ID number or your photo.
You must wait eight weeks between whole blood donations. Learn more about specific intervals for other types of donation.
You should not be under the influence of alcohol or recreational drugs at the time of donation.
Additional Requirements: Component-Specific Donation include that Platelet donors should avoid aspirin and/or aspirin-containing products 48 hours prior to donation and other anti-platelet medications
Updated Donation Eligibility Criteria
Health Conditions
Medications
HIV/AIDS Risk Behaviors
Tattoos, Piercings, Permanent Make-up & Acupuncture
European Travel/Residency Criteria
Travel to Malaria Risk Areas
Source Plasma

enter y to continue in bbms
```

```
LSP Python: ready conda_base :Python 3.8.3 Line 387, Col 159 UTF-8 LF RW Mem 68% 20:07
Console 7/A /Users/rajakumarisureshbabu
6.CHANGE YOUR PASSWORD
7.REJUVENATE PHARMACY
8.GENERATE BILL

enter your choice4
4.REJUVENATE BLOOD BANK MODULE
滴滴滴滴滴滴滴滴滴滴
1.insert into blood donors
2.change donor details
3.delete donor
4.update blood group details
5.see the list of eligible donors of this day

enter valid choice5
1.to view the standard GENERAL POLICY FOR BLOOD DONATION
2.COVID EXTRA RULES ON BLOOD DONATION SAFETY
3.elligible donors on a particular date satisying terms

enter your choice2
Due to COVID-19 (SARS-CoV-2 Coronavirus), we are requiring that people should not donate today ,if in the PAST 4 WEEKS:
    You had symptoms from a lab-diagnosed or suspected COVID-19 infection and have not had a subsequently-negative nasopharyngeal swab test result
        You have lived with or been in close contact with individuals diagnosed with or suspected of having COVID-19 infection
            You are a health care worker who has been caring for a patient diagnosed with or suspected of having COVID-19 and have not consistently been able to use recommended personal protective equipment (face mask, gown and gloves)
                Masks Required
                    All staff and donors are required to wear a mask or cloth face covering. One-way valve masks are not permitted.
                    COVID-19 Convalescent Plasma
                        Learn more about convalescent plasma to see if you are eligible to give this lifesaving donation to help COVID-19 patients. We encourage healthy donors who don't qualify to give convalescent plasma to continue scheduling whole blood, platelet and other donation type appointments to help patients in need

enter y to continue in bbms
```

# MODULE 3 – DOCTOR MODULE:

## 1)VIEW DOCTOR PROFILE:

The screenshot shows a Jupyter Notebook interface with the following content:

```
Python 3.8.1 (default, Jan  8 2020, 16:15:59)
Type "copyright", "credits" or "license" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]: runfile('/Users/rajakumarisureshbabu/untitled21.py', wdir='/Users/rajakumarisureshbabu')
Welcome user !
REJUVENATE HOSPITALS
HOSPITAL MANAGEMENT SYSTEM
LEADERS IN HEALTHCARE ADMINISTRATION
-----
1.CONTINUE AS ADMIN
2.CONTINUE AS DOCTOR
3.COVID SPECIALITY UNIT

enter choice2
enter doctor id101
enter the doc pass: 9821
Welcome Doctor
Rejuvenate Hospitals
1.Profile
2.Appointments
3.Change password
4.Prescription

enter your choice1
Enter your id doctor: 101
('samuel', 101, 'm', datetime.timedelta(seconds=43200), 'lungs', 12, 233, 'ggjhg@gmail.com', 638180491, 'fgfkkgkghkgkyfkfy')

enter y to continue
```

At the bottom of the interface, there is a status bar with the following information:

LSP Python: ready conda: base / Python 3.8.3 Line 888, Col 25 UTF-8 LF RW Mem 68% 23.07

## 2)CHANGE PASSWORD:

Welcome Doctor

Rejuvenate Hospitals

1.Profile

2.Appointments

3.Change password

4.Prescription

enter your choice3

CHANGE PASSWORD

enter old password9821

enter new password12345

confirm new doctor password12345

password change was successful

Welcome user !

REJUVENATE HOSPITALS

HOSPITAL MANAGEMENT SYSTEM

LEADERS IN HEALTHCARE ADMINISTRATION

---

1.CONTINUE AS ADMIN

2.CONTINUE AS DOCTOR

3.COVID SPECIALITY UNIT

enter choice2

enter doctor id101

enter the doc pass: 12345

Welcome Doctor

### 3)PRESCRIPTION:

Rejuvenate Hospitals

1.Profile

2.Appointments

3.Change password

4.Prescription

enter your choice4

Say patient name!

Symptoms

Prescription

Advice

Name- Raju

Symptoms- cold and fever

**Say patient name!**

**Symptoms**

**Prescription**

**Advice**

**Name- Raju**

**Symptoms- cold and fever**

**Prescription- Paracetamol**

**Advice- take good rest**

**PRESCRIPTION GENERATED AS PDF:**

TO  
HEAL

<b>Name</b>	<b>Symptoms</b>	<b>Prescription</b>	<b>Advice</b>
Raju	cold and fever	Paracetamol	take good rest



## 4)APPOINTMENTS:

### A) DISPLAY APPOINTMENTS OF A DOCTOR:

```
Console 26/A /Users/rajakumarisureshbabu
Welcome user !
REJUVENATE HOSPITALS
HOSPITAL MANAGEMENT SYSTEM
LEADERS IN HEALTHCARE ADMINISTRATION

-----
1.CONTINUE AS ADMIN
2.CONTINUE AS DOCTOR
3.COVID SPECIALITY UNIT

enter choice2

enter doctor id101

enter the doc pass: 12345
Welcome Doctor
Rejuvenate Hospitals
1.Profile
2.Appointments
3.Change password
4.Prescription

enter your choice2
1.Appointment display
2.Appointment deletion

Enter your choice:1
Your appointments are below doctor:

Enter the doctor's id: 101
(1191078, 101, datetime.timedelta(seconds=43200))

enter y to continue|
```

## B) APPOINTMENT DELETION:

```
enter y to continuey
Welcome Doctor
Rejuvenate Hospitals
1.Profile
2.Appointments
3.Change password
4.Prescription

enter your choice2
1.Appointment display
2.Appointment deletion

Enter your choice:2

Do you want to delete an appointment: y

Number of appointments wanted to be deleted: 1

Enter the doctor's id: 101

Enter the patient's id to be deleted1191078

Appointment deleted successfully!
Appointment deleted successfully!

enter y to continue|
```

```
Enter your choice:2

Do you want to delete an appointment: y

Number of appointments wanted to be deleted: 1

Enter the doctor's id: 101

Enter the patient's id to be deleted1191078

Appointment deleted successfully!
Appointment deleted successfully!

enter y to continuey
Welcome Doctor
Rejuvenate Hospitals
1.Profile
2.Appointments
3.Change password
4.Prescription

enter your choice2
1.Appointment display
2.Appointment deletion

Enter your choice:1
Your appointments are below doctor:

Enter the doctor's id: 101

enter y to continue|
```

## 5.LOGIN FROM A DIFFERENT DOCTOR ID:

The screenshot shows a Jupyter Notebook interface with the following details:

- Header:** /Users/rajakumarisureshbabu
- Toolbar:** Includes icons for file operations (New, Open, Save, etc.), cell execution, and help.
- Cell 1 (Python 3.8.1):** Displays the Python startup message: "Python 3.8.1 (default, Jan 8 2020, 16:15:59)  
Type "copyright", "credits" or "license" for more information."
- Cell 2 (IPython 7.16.1):** Displays the IPython startup message: "IPython 7.16.1 -- An enhanced Interactive Python."
- Cell 3 (In [1]):** Runs the command `runfile('/Users/rajakumarisureshbabu/untitled21.py', wdir='/Users/rajakumarisureshbabu')`. The output shows:
  - Welcome user !
  - REJUVENATE HOSPITALS
  - HOSPITAL MANAGEMENT SYSTEM
  - LEADERS IN HEALTHCARE ADMINISTRATION
- User Interaction:** The user has entered the following choices:
  - 1.CONTINUE AS ADMIN
  - 2.CONTINUE AS DOCTOR
  - 3.COVID SPECIALITY UNIT
- System Response:** The system has prompted the user to enter choice2 and doctor id100, followed by a welcome message for the doctor.
- Final Output:** The system displays the rejuvenate Hospitals menu with options 1.Profile, 2.Appointments, 3.Change password, and 4.Prescription, and asks the user to enter their choice.
- Bottom Status Bar:** Shows LSP Python: ready, conda: base , Python 3.8.3, Line 864, Col 51, UTF-8, LF, RW, Mem 71%, 23:25.

# 14 MYSQL TABLES:

## 1. APPOINTMENTS TABLE:

	p_id	d_id	time
▶	2191007	201	11:00:00

appointments 1

## 2. TBLBLOODGROUP:

	bg_id	BloodGroup	PostingDate
▶	1	a+	2021-01-21 00:00:00
	2	a-	2021-01-20 00:00:00
	3	b+	2021-01-18 00:00:00
	4	o+	2021-01-17 00:00:00

tblbloodgroup 1

## 3. WARDS:

	building	room_number	number_of_beds	status	room_rent
▶	a	1	28	occupied	878
	a	30	28	free	626
	a	68	30	occupied	8378
	a	39	39	occupied	3838
	b	44	33	free	666
	b	66	33	occupied	989
	b	44	32	free	983
	a	54	32	free	940
	b	54	30	occupied	948
	a	38	54	occupied	987

#### 4.PHARMACY:

mid	med	pharma	price	num	mfg	exp
► 1	CROCIN	GSK_HEALTH_CONSUMER	30	200	2020-07-17	2021-07-27
2	FORACORT	CIPLA	304	350	2020-08-24	2021-08-24
3	LEVOLIN	CIPLA	192	230	2020-03-24	2022-02-24
4	BRUFFIN	ABBOTT	9	150	2019-04-30	2021-04-25
5	COMBIFLAM	SANOFI	34	400	2020-01-16	2021-01-15
6	BIFILAC	TABLETS_IMDIA_LIMITED	98	379	2020-11-25	2021-11-24
7	THYRONORM	ABBOTT	148	468	2019-12-24	2021-12-23
8	MEFTAL_SPAS	BLUE_CROSS	42	478	2020-04-16	2021-04-16
9	DOLO650	MICRO_LABS_LIMITED	30	400	2020-03-30	2021-03-29
10	P500	APEX	15	342	2020-01-24	2021-01-23
11	ibugesic	cipla	500	21	2020-11-20	2022-11-22

pharma 1

#### 5.NCPATIENTS:

name	p_id	date_in	date_out	gender	age	aadhar	insurance	phone	address	disease	room	room_number	email	doc
► AMITESH	1191001	2020-11-01	2020-11-14	M	23	123	123	123456789	10 DOWNING STREET , SANSAD BHAVAN	INFLUENZA	VIP	12	AMITACE@GMAIL.COM	201
varun	1191002	2020-11-02	2020-11-18	m	54	142	144	1998283	manhattan diacon alley	heart failure and arrhythmias	general	13	vm32@gmail.com	209
sanghavi	1191008	2020-11-03	2020-11-10	f	28	231	211	76529827	highrise paradise,buckingham street,jafferkharpet	sepsis	vip	33	sanhv@hotmail.com	209
swastika	1191008	2020-11-04	2020-11-14	f	37	119	181	12893649	Bermuda Haven,Ellingham street,serpeck nagar	subarachnoid hemorrhage	general	65	swasti@yahoo.com	204
sanskriti	1191088	2020-11-02	2020-11-30	f	74	233	321	332224	finland corner, koyembedu nagar	seizure	vip	29	sanspuri@gmail.com	201
swapnil rathore	1191089	2020-11-06	2020-11-09	m	32	1943	228	28392	manohar vikas bhavan	anemia	vip	38	swapthore@gmail.com	201
dewaker	1191072	2020-11-06	2020-11-23	m	94	43441	2425	2466	ghghgjbjkjhkh	Alzheimer's disease	82	45	ghdtdhj	203
thamizh	1191065	2020-11-07	2020-11-25	f	66	456	77	6666	dfgfdgfhggkgkj	pneumonia	vip	66	giuh	201
goookie	1191092	2020-11-05	2020-11-30	m	65	3636	33234	33782	fdgehjaihounngomc	Nephritis	general	87	hjhjjh	202

#### 6.DOCTORS' PASSWORDS:

doc_id	doc_pwd
► 100	2305
101	12345
102	9450
103	9720

doc\_pwd 1

## 7. DATE\_WISE\_COVID DEATHS:

#### 8.COVID INITIATIVES TABLE:

initiative	details
► CB-NAAT	REJUVENATE Hospitals takes COVID-19 testing to a whole new level with CB-NAAT By embracing the Cartridge Based Nu...

#### **9.ADMINISTRATOR PASSWORD:**

	name_admin	admin_pwd
▶	administrator	1234

## 10. DATE WISE COVID CASES:

## 11. BLOOD DONORS' TABLE:

## 12. DOCTORS TABLE:

	Name	d_id	gend	time	spec	exp	fee	email	ph_no	addr
►	samuel	101	m	12:00:00	lungs	12	233	ggjhg@gmail.com	638180491	fjfkgkghkhgkyfkfy
richard	102	m	06:00:00	neuro	13	500	ggjg@gmail.com	635180491	fkgkkghkhgkyfkfy	
kavita	103	f	02:00:00	gynaec	13	500	ggj@gmail.com	63510491	fkgkhkhgkyfkfy	
karen	104	f	08:00:00	gynaec	13	500	gg@gmail.com	6350491	fkgkkhgkyfkfy	
ruben	204	m	01:00:00	physio	13	500	g@gmail.com	635091	fkkhgkyfkfy	
paul	201	m	19:00:00	heart	18	500	guuj@gmail.com	6354091	fkkhgjjkyfkfy	
daisy	202	f	22:00:00	heart	18	500	guu@gmail.com	635491	fkkyfkfy	

13. DATEWISE COVID RECOVERIES:

date_of	recoveries
► 2020.11.21	3
2020.11.23	0
2020.11.21	3
2020.11.20	2
2020.11.23	0
2020.11.20	2
2020.11.21	3
2020.11.23	0
2020.11.20	2
2020.11.21	3
2020.11.23	0
2020.11.20	2
2020.11.21	3
2020.11.23	0
2020.11.20	2
2020.11.21	3
2020.11.23	0

date\_wise\_covidrecoveries 1

14. COVID PATIENTS:

patient_id	age	gender	room	date	name	status_patient	p_doc	
► 2191001	34	f	1	2020-11-20	samantha	recovered	101	
2191002	26	f	3	2020-11-21	rebecca	recovered	102	
2191018	79	m	4	2020-11-21	charan	ongoing	102	
2191003	89	f	7	2020-11-21	moana	dead	104	
2191007	59	m	9	2020-11-20	fischer	dead	100	
2191005	14	m	5	2020-11-21	sabu	recovered	101	
2191006	33	f	88	2020-11-20	sangamithra	recovered	101	
2191073	88	m	77	2020-11-23	daniel	ongoing	100	
2191065	54	m	45	2020-11-23	fahadh	ongoing	100	
2191007	89	f	43	2020-11-20	shankari	dead	102	
2191004	57	m	59	2020-11-23	ramesh	ongoing	102	
2191009	51	m	50	2022-11-20	dhinesh	ongoing	101	
2191036	31	f	66	2020-11-20	harshavar...	ongoing	103	
2191044	28	f	34	2020-11-20	namrata	ongoing	101	
2191048	39	m	39	2020-11-23	firoz	ongoing	102	
2191006	45	f	63	2020-11-20	sharmila	ongoing	104	
2191046	70	m	44	2020-11-20	narendar	dead	101	
2191054	23	m	41	2020-11-21	varun	recovered	101	

COVID19\_PATIENTS 1

# BIBLIOGRAPHY

- Sumita Arora - Python for class XII textbook and Practical Workbook.
- [www.stackoverflow.com](http://www.stackoverflow.com)
- [www.python4csip.com](http://www.python4csip.com)
- [www.W3schools.com](http://www.W3schools.com)