Introduction

The development of a fully functional fitness module was the problem statement. The problem statement further specified what modules were required- We dealt with creating a monitoring system for blood pressure, pulse, blood sugar and heartbeat, calculating calories burnt and calculating body mass index.

Our vision was to create an elegant output window which we achieved by incorporating a graphical user interface with the help of Tkinter.

We divided our problem into two parts, wherein we developed our basic code as per our understanding. As we progressed in further learning of the python programming language we would enhance our code accordingly. This is the fitness module. It consists of the basic code to understand individually the code for blood pressure, pulse, blood sugar and heartbeat, calculating calories burnt and body mass index.

As the second part of the project, we dealt with adding a graphic user interface to the code. Our output window would essentially provide radio buttons to choose what the user would like to check, which would further lead to a window as per user inputs where the user is expected to enter the respective values and proceed in order to receive the output. The user can choose to exit or continue for further examination.

On completion of the examination, the User may also choose to generate a fitness report based on the recent examinations.

Design

```
'''This module will help you to know your overall fitness'''


#BMI
def bmi(weight,height):
    bmi1 = float( weight/height**2)
    if bmi1 <16:
        r=("BMI="+" "+str(round(bmi1,2))+" "+"Severely UnderWeight")
    elif bmi1 >=16 and bmi1 <17:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Underweight")
    elif bmi1 >=17 and bmi1 <18.5:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Slightly UnderWeight")
    elif bmi1 >= 18.5 and bmi1 <=25.0:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Normal weight")
    elif bmi1 >25.0 and bmi1 <30:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Slightly Overweight")
    elif bmi1 >=30.0 and bmi1 <35:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Overweight")
    else:
        r=("BMI="+" "+str(round(bmi1,2))+"  "+"Severely Overweight")
    return r



#BP
def bp(sys,dia):
    if sys<90 and dia<=60:
        r=("LOW BLOOD PRESSURE")
    elif (sys>=90 and sys<120) and (dia>60 and dia<80):
        r=("NORMAL BLOOD PRESSURE")
    elif (sys>=120 and sys<130) and (dia>=80 and dia<90):
        r=("ELEVATED BLOOD PRESSURE")
```

```
    elif (sys>=130 and sys<140) and (dia>=90 and dia<100):
        r=("HIGH BLOOD PRESSURE-HYPERTENSION")
    elif sys>=140 and dia>100:
        r=("HYPERTENSIVE CRISIS")
    else:
        r=("invalid values")

    return r


#CALORIES BURNT
def calorie(time):
    return ("You lost"+" "+str(round(time * 3.4,3))+" "+"calories.")


#HEART BEAT
def heart(a,b):#a=age;b=bpm
    if a<0.5:
        if b>=160:
            r="Heart beat is high."
        elif b<160 and b>100:
            r="Heart in good condition."
        else:
            r="Heart beat is low."
    elif a<1:
        if b>=140:
            r="Heart beat is high."
        elif b<140 and b>80:
            r="Heart in good condition."
        else:
            r="Heart beat is low."
    elif a<4:
        if b>=130:
```

```
        r="Heart beat is high."
    elif b<130 and b>80:
        r="Heart in good condition."
    else:
        r="Heart beat is low."
elif a<6:
    if b>=120:
        r="Heart beat is high."
    elif b<120 and b>80:
        r="Heart beat in good."
    else:
        r="Heart beat is low."
elif a<11:
    if b>=110:
        r="Heart beat is high."
    elif b<110 and b>70:
        r="Heart beat  is good."
    else:
        r="Heart beat is low."
elif a<15:
    if b>=105:
        r="Heart beat is high."
    elif b<105 and b>60:
        r="Heart beat is good."
    else:
        r="Heart beat is low."
else:
    if b>=100:
        r="Heart beat is high."
    elif b<100 and b>60:
        r="Heart beat is good."
    else:
        r="Heart beat is low."
```

```
    return r


#PULSE RATE
def pulse(age,pulse4):

    max_heart_rate= int(220-age)
    min_target = (65/100)*max_heart_rate
    max_target =(85/100)*max_heart_rate

    if pulse4 >=min_target and pulse4 <= max_target:
        r=("You are exercising well keep going!!")
    elif pulse4 >max_target and pulse4 <= max_heart_rate:
         r=("Slow down your target heart rate should be in between"+" "+str(min_target)+"
"+"and"+" "+str(max_target))
    elif  pulse4 > max_heart_rate:
             r=("Stop your target heart rate should be in between"+" "+str(min_target)+"
"+"and"+" "+str(max_target))
    elif pulse4 < min_target :
                  r=(" Exercise harder your target heart rate should be in between"+"
"+str(min_target)+" "+"and"+" "+str(max_target))

    return r


#S U G A R
def random(r):#random
    if (r>=60 and r<80):
        a=("low Blood Sugar level-Hypoglycemia")
    elif (r>=80 and r<=140):
        a=("Normal Blood sugar level")
    elif (r>140 and r<=160):
        a=("Boderline Blood sugar level ")
```

```
    elif (r>160):
        a=("high Blood Sugar level-Hyperglycemia")
    else:
        a=("enter valid values")
    return a


def fasting(f):#fasting
    if (f>40 and f<80):
        a=("low Blood Sugar level-Hypoglycemia ")
    elif (f>=80 and f<=110):
        a=("Normal Blood sugar level ")
    elif (f>110 and f<=125):
        a=("Boderline Blood sugar level ")
    elif (f>125):
        a=("high Blood Sugar level-Hyperglycemia ")
    else:
        a=("enter valid values")
    return a


def p_p(p):#post-prandial
    if (p>=70 and p<140 ):
        c=("low Blood Sugar level-Hypoglycemia")
    elif (p>=140 and p<=160):
        c=("Normal Blood sugar level")
    elif (p>160 and p<200):
        c=("Boderline Blood Sugar level")
    elif (p>=200):
        c=("high Blood Sugar level-Hyperglycemia")
    else:
        c=("enter valid values")
    return c
```

```
def Hb1Ac(h):#HbA1c
    if (h>3 and h<=4):
        d=("low Blood Sugar level-Hypoglycemia")
    elif (h>4 and h<=6):
        d=("Normal Blood sugar level")
    elif (h>6 and h<=8):
        d=("Boaderline Blood sugar level ")
    elif (h>8 and h<=14):
        d=("high Blood Sugar level-Hyperglycemia")
    else:
        d=("enter valid values")
    return d
##############################################################################
'''This graphical user interface will use the above fitness module'''

from tkinter import *
from fitness import *
from tkinter import messagebox as m
root = Tk()
root.geometry('700x700')

root.title('Fitness')


def f1_f2():
  frame1.forget()
  frame2.pack(fill='both', expand=1)


frame1 = Frame(root, bg="#AED6F1")
```

```
# ------------------BMI--------------------

frame2 = Frame(root, bg='#000000')
lbl2 = Label(frame2, text='BMI CHECKUP', bg='#000000', fg='cyan', font = ('Helvetica', 20, 'bold'))
lbl2.pack(pady=20)


def click():
    global k

    height = float(e1.get())
    weight = float(e2.get())
    k = bmi(weight, height)
     Label(f1_frame2, text = k, bg='#AED6F1',font = ('Helvetica', 12, 'bold')).grid(row=5, column=0, columnspan=2)

f1_frame2 = Frame(frame2, bg='#000000')
f1_frame2.pack(expand=True)

lbl1 = Label(f1_frame2, text = 'Enter your height in m: ',bg='#000000',fg='white',font = ('Helvetica', 16))
lbl1.grid(row=0, column=0, sticky='e', pady=5)
e1 = Entry(f1_frame2)
e1.grid(row=0, column=1, pady=5)

lbl2 = Label(f1_frame2, text = 'Enter your weight in kg: ', bg='#000000',fg='white',font = ('Helvetica', 16))
lbl2.grid(row=1, column=0, sticky='e', pady=5)
e2 = Entry(f1_frame2)
e2.grid(row=1, column=1, pady=5)
```

```
btn = Button(f1_frame2, text='Find BMI',font = ('Helvetica', 10,'bold'), width=20, command=click)
btn.grid(row=2, column=0, columnspan=2, pady=16)


def f1_f2():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame2.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame1_2 = Frame(frame2, bg="#AED6F1")
frame1_2.pack(fill='x')

next1=Button(frame1_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f2)
next1.pack(side=RIGHT, padx=16, pady=20, ipadx=20)



# ----------------------Blood pressure-------------------------------

def bpcal():
    global p
    sys = float(bpe1.get())
    dia = float(bpe2.get())
    p = bp(sys, dia)
            Label(f1_frame3, text = p, bg='#AED6F1', font = ('Helvetica', 12, 'bold')).grid(row=5,column=0, columnspan=2)
```

```
frame3 = Frame(root, bg='#000000')

lbl3 = Label(frame3, text='BLOOD PRESSURE CHECKUP', bg='#000000', fg='cyan', font = ('Helvetica', 20, 'bold'))
lbl3.pack(pady=20)

f1_frame3 = Frame(frame3, bg='#000000')
f1_frame3.pack(expand=True)

bplb1 = Label(f1_frame3, text = "Systolic blood pressure :",bg='#000000',fg='white',font = ('Helvetica', 16))
bplb1.grid(row = 0, column = 0, sticky = 'e')
bpe1 = Entry(f1_frame3)
bpe1.grid(row = 0, column = 1)

bplb2 = Label(f1_frame3, text = "Diastolic blood pressure :", pady=16, bg='#000000',fg='white',font = ('Helvetica', 16))
bplb2.grid(row = 1, column = 0, sticky = 'e')
bpe2 = Entry(f1_frame3)
bpe2.grid(row = 1, column = 1)

bpbtn = Button(f1_frame3, text='Find BP',font = ('Helvetica', 10,'bold'), command=bpcal, width=20)
bpbtn.grid(row = 2, column = 0, columnspan=2, pady=16)



def f1_f3():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame3.forget()
        frame_choice.pack(fill='both', expand=1)
```

```
    else:
        reportgen()



frame1_3 = Frame(frame3, bg="#AED6F1")
frame1_3.pack(fill='x')

nextbp1=Button(frame1_3,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f3)
nextbp1.pack(side=RIGHT, padx=16, pady=20, ipadx=20)



# ----------------------Calorie-------------------------------

def caloriecal():
    global c
    weight = float(ce1.get())

    c = calorie(weight)
            Label(f1_frame4, text = c, bg='#AED6F1', font = ('Helvetica', 12,
'bold')).grid(row=5,column=0, columnspan=2)

frame4 = Frame(root, bg='#000000')

lbl4 = Label(frame4, text='CALORIES BURNT CHECKUP', bg='#000000', fg='cyan', font
= ('Helvetica', 20, 'bold'))
lbl4.pack(pady=20)

f1_frame4 = Frame(frame4, bg='#000000')
f1_frame4.pack(expand=True)
```

```
clb1 = Label(f1_frame4, text = 'Duration of exercise in minutes :',bg='#000000',fg='white',font = ('Helvetica', 16))
clb1.grid(row = 0, column = 0, sticky = 'e')
ce1 = Entry(f1_frame4)
ce1.grid(row = 0, column = 1)



cbtn = Button(f1_frame4, text='Find Calorie Burn',font = ('Helvetica', 10,'bold'), command=caloriecal, width=20)
cbtn.grid(row = 2, column = 0, columnspan=2, pady=16)



def f1_f4():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame4.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()

def f11_f4():
    frame4.forget()
    frame_choice.pack(fill='both', expand=1)

frame4_3 = Frame(frame4, bg="#AED6F1")
frame4_3.pack(fill='x')



next2=Button(frame4_3,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f4)
next2.pack(side=RIGHT, padx=16, pady=20, ipadx=20)
```

```
# ----------------------Heart Beat------------------------------

frame5 = Frame(root, bg='#000000')
hlbl2 = Label(frame5, text='HEART BEAT CHECKUP', bg='#000000', fg='cyan', font = ('Helvetica', 20, 'bold'))
hlbl2.pack(pady=20)


def heartbeatcal():
    global heartbeat1
    age = float(he1.get())
    bpm = float(he2.get())
    heartbeat1= heart(age,bpm)
        Label(f1_frame5, text = heartbeat1, bg='#AED6F1',font = ('Helvetica', 12, 'bold')).grid(row=5, column=0, columnspan=2)

f1_frame5 = Frame(frame5, bg='#000000')
f1_frame5.pack(expand=True)

hlbl1 = Label(f1_frame5, text = 'Enter your age: ',bg='#000000',fg='white',font = ('Helvetica', 16))
hlbl1.grid(row=0, column=0, sticky='e', pady=5)
he1 = Entry(f1_frame5)
he1.grid(row=0, column=1, pady=5)

hlbl2 = Label(f1_frame5, text = 'Enter your bpm: ', bg='#000000',fg='white',font = ('Helvetica', 16))
hlbl2.grid(row=1, column=0, sticky='e', pady=5)
he2 = Entry(f1_frame5)
```

```
he2.grid(row=1, column=1, pady=5)


hbtn = Button(f1_frame5, text='Find Heartbeat Cond.',font = ('Helvetica', 10,'bold'), width=20, command=heartbeatcal)
hbtn.grid(row=2, column=0, columnspan=2, pady=16)


def f1_f5():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame5.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame5_2 = Frame(frame5, bg="#AED6F1")
frame5_2.pack(fill='x')

nexth=Button(frame5_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f5)
nexth.pack(side=RIGHT, padx=16, pady=20, ipadx=20)



# ----------------------Pulse------------------------------

frame6 = Frame(root, bg='#000000')
pplbl2 = Label(frame6, text='PULSE CHECKUP', bg='#000000', fg='cyan', font = ('Helvetica', 20, 'bold'))
pplbl2.pack(pady=20)
```

```
def pulse6():
   global pu
   age1 = float(pe1.get())
   pulse1 = float(pe2.get())
   pu = pulse(age1,pulse1)
    Label(f1_frame6, text = pu, bg='#AED6F1',font = ('Helvetica', 12, 'bold')).grid(row=5,
column=0, columnspan=2)

f1_frame6 = Frame(frame6, bg='#000000')
f1_frame6.pack(expand=True)

plbl1 = Label(f1_frame6, text = 'Enter your age: ',bg='#000000',fg='white',font =
('Helvetica', 16))
plbl1.grid(row=0, column=0, sticky='e', pady=5)
pe1 = Entry(f1_frame6)
pe1.grid(row=0, column=1, pady=5)

plbl2 = Label(f1_frame6, text = 'Enter your pulse: ', bg='#000000',fg='white',font =
('Helvetica', 16))
plbl2.grid(row=1, column=0, sticky='e', pady=5)
pe2 = Entry(f1_frame6)
pe2.grid(row=1, column=1, pady=5)


pbtn = Button(f1_frame6, text='Find Pulse Cond.',font = ('Helvetica', 10,'bold'), width=20,
command=pulse6)
pbtn.grid(row=2, column=0, columnspan=2, pady=16)


def f1_f6():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
```

```
    if condition:
        frame6.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame6_2 = Frame(frame6, bg="#AED6F1")
frame6_2.pack(fill='x')

next4=Button(frame6_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f6)
next4.pack(side=RIGHT, padx=16, pady=20, ipadx=20)



# ----------------------Sugar-------------------------------

#RANDOM
frame7 = Frame(root, bg='#000000')
rrlbl2 = Label(frame7, text='RANDOM CHECKUP', bg='#000000', fg='cyan', font =
('Helvetica', 20, 'bold'))
rrlbl2.pack(pady=20)



def randomtest():
    global u
    r = float(re1.get())

    u = random(r)
     Label(f1_frame7, text = u, bg='#AED6F1',font = ('Helvetica', 12, 'bold')).grid(row=5,
column=0, columnspan=2)
```

```
f1_frame7 = Frame(frame7, bg='#000000')
f1_frame7.pack(expand=True)

rlbl1 = Label(f1_frame7, text = 'Enter random test blood sugar: ',bg='#000000',fg='white',font = ('Helvetica', 16))
rlbl1.grid(row=0, column=0, sticky='e', pady=5)
re1 = Entry(f1_frame7)
re1.grid(row=0, column=1, pady=5)



rbtn = Button(f1_frame7, text='Find Cond.',font = ('Helvetica', 10,'bold'), width=20, command=randomtest)
rbtn.grid(row=2, column=0, columnspan=2, pady=16)


def f1_f7():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame7.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame7_2 = Frame(frame7, bg="#AED6F1")
frame7_2.pack(fill='x')

next5=Button(frame7_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f7)
next5.pack(side=RIGHT, padx=16, pady=20, ipadx=20)
```

```
############################################################

#FASTING
frame8 = Frame(root, bg='#000000')
fflbl2 = Label(frame8, text='FASTING SUGAR CHECKUP', bg='#000000', fg='cyan', font
= ('Helvetica', 20, 'bold'))
fflbl2.pack(pady=20)


def fastingcal():
  global f2
  f = float(fe1.get())

  f2 = fasting(f)
   Label(f1_frame8, text = f2, bg='#AED6F1',font = ('Helvetica', 12, 'bold')).grid(row=5,
column=0, columnspan=2)

f1_frame8 = Frame(frame8, bg='#000000')
f1_frame8.pack(expand=True)

flbl1 = Label(f1_frame8, text = 'Enter your Fasting sugar: ',bg='#000000',fg='white',font
= ('Helvetica', 16))
flbl1.grid(row=0, column=0, sticky='e', pady=5)
fe1 = Entry(f1_frame8)
fe1.grid(row=0, column=1, pady=5)



fbtn = Button(f1_frame8, text='Find Cond.',font = ('Helvetica', 10,'bold'), width=20,
command=fastingcal)
fbtn.grid(row=2, column=0, columnspan=2, pady=16)
```

```
def f1_f8():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame8.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame8_2 = Frame(frame8, bg="#AED6F1")
frame8_2.pack(fill='x')

next6=Button(frame8_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f8)
next6.pack(side=RIGHT, padx=16, pady=20, ipadx=20)

############################################################

#POST-PRANDIAL
frame9 = Frame(root, bg='#000000')
p_pplbl2 = Label(frame9, text='POST-PRANDIAL CHECKUP', bg='#000000', fg='cyan',
font = ('Helvetica', 20, 'bold'))
p_pplbl2.pack(pady=20)


def P_Pcal():
  global p_p1
  p_p11= float(p_pe1.get())
  p_p1 = p_p(p_p11)
        Label(f1_frame9, text = p_p1, bg='#AED6F1',font = ('Helvetica', 12,
'bold')).grid(row=5, column=0, columnspan=2)
```

```
f1_frame9 = Frame(frame9, bg='#000000')
f1_frame9.pack(expand=True)

p_plbl1 = Label(f1_frame9, text = 'Enter your post prandial sugar: ',bg='#000000',fg='white',font = ('Helvetica', 16))
p_plbl1.grid(row=0, column=0, sticky='e', pady=5)
p_pe1 = Entry(f1_frame9)
p_pe1.grid(row=0, column=1, pady=5)



p_pbtn = Button(f1_frame9, text='Find Cond.',font = ('Helvetica', 10,'bold'), width=20, command= P_Pcal)
p_pbtn.grid(row=2, column=0, columnspan=2, pady=16)


def f1_f9():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame9.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame9_2 = Frame(frame9, bg="#AED6F1")
frame9_2.pack(fill='x')

next7=Button(frame9_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f9)
next7.pack(side=RIGHT, padx=16, pady=20, ipadx=20)
```

```
############################################################

#Hb1Ac
frame16 = Frame(root, bg='#000000')
Hb1Acclbl2 = Label(frame16, text='Hb1Ac1 CHECKUP', bg='#000000', fg='cyan', font =
('Helvetica', 20, 'bold'))
Hb1Acclbl2.pack(pady=20)


def Hb1Accal():
  global hb1ac1
  hb1ac= float(hb1ace1.get())
  hb1ac1 = Hb1Ac(hb1ac)
      Label(f1_frame16, text = hb1ac1, bg='#AED6F1',font = ('Helvetica', 12,
'bold')).grid(row=5, column=0, columnspan=2)

f1_frame16 = Frame(frame16, bg='#000000')
f1_frame16.pack(expand=True)

hb1aclbl1 = Label(f1_frame16, text = 'Enter your Hb1Ac sugar:
',bg='#000000',fg='white',font = ('Helvetica', 16))
hb1aclbl1.grid(row=0, column=0, sticky='e', pady=5)
hb1ace1 = Entry(f1_frame16)
hb1ace1.grid(row=0, column=1, pady=5)



hb1acbtn = Button(f1_frame16, text='Find Cond.',font = ('Helvetica', 10,'bold'), width=20,
command=Hb1Accal)
hb1acbtn.grid(row=2, column=0, columnspan=2, pady=16)
```

```
def f1_f16():
    condition = m.askyesno('Fitness', 'Do you want to continue ?')
    if condition:
        frame16.forget()
        frame_choice.pack(fill='both', expand=1)
    else:
        reportgen()



frame16_2 = Frame(frame16, bg="#AED6F1")
frame16_2.pack(fill='x')

next8=Button(frame16_2,text="Next",font = ('Helvetica', 10,'bold'),command=f1_f16)
next8.pack(side=RIGHT, padx=16, pady=20, ipadx=20)



################################################################################
"Choice"

def selection():
    choice = var1.get()
    if choice == 'BMI':
        frame_choice.forget()
        frame2.pack(fill='both', expand=1)

    elif choice == 'Blood Pressure':
        frame_choice.forget()
        frame3.pack(fill='both', expand=1)
    elif choice == 'Calorie':
        frame_choice.forget()
        frame4.pack(fill='both', expand=1)
```

```
    elif choice == 'Heart':
      frame_choice.forget()
      frame5.pack(fill='both', expand=1)
    elif choice == 'Pulse':
      frame_choice.forget()
      frame6.pack(fill='both', expand=1)
    elif choice == 'Random':
      frame_choice.forget()
      frame7.pack(fill='both', expand=1)
    elif choice == 'Fasting':

       frame_choice.forget()
       frame8.pack(fill='both', expand=1)
    elif choice == 'P_P':

       frame_choice.forget()
       frame9.pack(fill='both', expand=1)
    elif choice == 'Hb1Ac':

       frame_choice.forget()
       frame16.pack(fill='both', expand=1)


frame_choice = Frame(root, bg='#000000')
frame_choice.pack(fill='both', expand=1)
lbl2_frame_choice = Label(frame_choice, bg='#000000',fg="yellow", text=' F I T N E S S
C H E C K U P', font = ( "Helvetica",24, 'bold italic underline'))
lbl2_frame_choice.pack(pady=20)

lbl1_frame_choice = Label(frame_choice, bg='#000000',fg='#00ffff' ,text='What  do  you
want  to  check  ?', font = ('Helvetica', 18, 'bold'))
lbl1_frame_choice.pack(pady=100)
```

```
f1_frame_choice = Frame(frame_choice, bg='#AED6F1')
f1_frame_choice.pack(expand=True)


var1 = StringVar()
var1.set('BMI')

r1 = Radiobutton(f1_frame_choice,bg='#000000', fg="#000fff000", text = 'BMI Level',font = ('Helvetica', 16), variable = var1, value = 'BMI')
r2 = Radiobutton(f1_frame_choice,bg='#000000', fg="#000fff000", text = 'Blood Pressure',font = ('Helvetica', 16), variable = var1, value = 'Blood Pressure')
r3 = Radiobutton(f1_frame_choice,bg='#000000', fg="magenta", text = 'Calorie Burnt',font = ('Helvetica', 16), variable = var1, value = 'Calorie')
r4 = Radiobutton(f1_frame_choice,bg='#000000', fg="magenta", text = 'Heart Beat',font = ('Helvetica', 16), variable = var1, value = 'Heart')
r5 = Radiobutton(f1_frame_choice,bg='#000000', fg="#000fff000", text = 'Pulse Rate',font = ('Helvetica', 16), variable = var1, value = 'Pulse')
r6 = Radiobutton(f1_frame_choice,bg='#000000', fg="#000fff000", text = 'Random Sugar Test',font = ('Helvetica', 16), variable = var1, value = 'Random')
r7 = Radiobutton(f1_frame_choice,bg='#000000', fg="magenta", text = 'Fasting Sugar Test',font = ('Helvetica', 16), variable = var1, value = 'Fasting')
r8 = Radiobutton(f1_frame_choice,bg='#000000', fg="magenta", text = 'Post Prandial Sugar Test',font = ('Helvetica', 16), variable = var1, value = 'P_P')
r9 = Radiobutton(f1_frame_choice,bg='#000000', fg="#000fff000", text = 'Hb1Ac Sugar Test',font = ('Helvetica', 16), variable = var1, value = 'Hb1Ac')

r1.grid(row=0, column=0, sticky='w', padx=25, pady=10)
r2.grid(row=0, column=1, sticky='w', padx=25, pady=10)
r3.grid(row=1, column=0, sticky='w', padx=25, pady=10)
r4.grid(row=1, column=1, sticky='w', padx=25, pady=10)
r5.grid(row=2, column=0, sticky='w', padx=25, pady=10)
r6.grid(row=2, column=1, sticky='w', padx=25, pady=10)
r7.grid(row=3, column=0, sticky='w', padx=25, pady=10)
```

```
r8.grid(row=3, column=1, sticky='w', padx=25, pady=10)
r9.grid(row=4, column=0, sticky='w', padx=25, pady=10)

frame_choice_end = Frame(frame_choice, bg='#AED6F1')
frame_choice_end.pack(fill='x')

btn_choice = Button(frame_choice_end, text = 'Next',font = ('Helvetica', 10,'bold'), command=selection,width=12)
btn_choice.pack(side=RIGHT,pady=10,padx=10)
################################################################################
"Report"

p="---"
k="---"
pu="---"
u="---"
c="---"
f2="---"
p_p1="---"
hb1ac1="---"
heartbeat1="---"
def reportgen():


  root.geometry('600x750')

  frame1.forget()
  frame2.forget()
  frame3.forget()
  frame4.forget()
  frame5.forget()
  frame6.forget()
  frame7.forget()
```

```
frame8.forget()
frame9.forget()
frame16.forget()

frame_report = Frame(root)
frame_report.pack(fill='both', expand=1)
lbl_frame_report = Label(frame_report, text = 'Fitness Report', bg="#AED6F1",font = ('Helvetica', 14,'bold'))
lbl_frame_report.pack(fill='x')

f_bmi_report = LabelFrame(frame_report, text = 'BMI Report',font = ('Helvetica', 12))
f_bmi_report.pack(fill='x', padx=16, pady = 16)
bmi_report_lbl = Label(f_bmi_report, text = k,font = ('Helvetica', 11))
bmi_report_lbl.pack()

f_bp_report = LabelFrame(frame_report, text = 'Blood Pressure Report',font = ('Helvetica', 12))
f_bp_report.pack(fill='x', padx=16, pady = 16)
bp_report_lbl = Label(f_bp_report, text = p,font = ('Helvetica', 11))
bp_report_lbl.pack()

f_cal_report = LabelFrame(frame_report, text = 'Calorie Report',font = ('Helvetica', 12))
f_cal_report.pack(fill='x', padx=16, pady = 16)
cal_report_lbl = Label(f_cal_report, text = c,font = ('Helvetica', 11))
cal_report_lbl.pack()

f_hb_report = LabelFrame(frame_report, text = 'Heart Beat Report',font = ('Helvetica', 12))
f_hb_report.pack(fill='x', padx=16, pady = 16)
hb_report_lbl = Label(f_hb_report, text = heartbeat1,font = ('Helvetica', 11))
hb_report_lbl.pack()

f_pulse_report = LabelFrame(frame_report, text = 'Pulse Report',font = ('Helvetica', 12))
```

```
 f_pulse_report.pack(fill='x', padx=16, pady = 16)
 pulse_report_lbl = Label(f_pulse_report, text = pu,font = ('Helvetica', 11))
 pulse_report_lbl.pack()

 f_r_report = LabelFrame(frame_report, text = 'Random Sugar Report',font = ('Helvetica',
12))
 f_r_report.pack(fill='x', padx=16, pady = 16)
 r_report_lbl = Label(f_r_report, text = u,font = ('Helvetica', 11))
 r_report_lbl.pack()

 f_fasting_report = LabelFrame(frame_report, text = 'Fasting Report',font = ('Helvetica',
12))
 f_fasting_report.pack(fill='x', padx=16, pady = 16)
 fasting_report_lbl = Label(f_fasting_report, text = f2,font = ('Helvetica', 11))
 fasting_report_lbl.pack()

  f_pp_report = LabelFrame(frame_report, text = 'Pst Prandial Sugar Report',font =
('Helvetica', 12))
 f_pp_report.pack(fill='x', padx=16, pady = 16)
 pp_report_lbl = Label(f_pp_report, text = p_p1,font = ('Helvetica', 11))
 pp_report_lbl.pack()

  f_Hb1Ac_report = LabelFrame(frame_report, text = 'Hb1Ac Sugar Report',font =
('Helvetica', 12))
 f_Hb1Ac_report.pack(fill='x', padx=16, pady = 16)
 Hb1Ac_report_lbl = Label(f_Hb1Ac_report, text = hb1ac1,font = ('Helvetica', 11))
 Hb1Ac_report_lbl.pack()



################################################################################
root.mainloop()
```

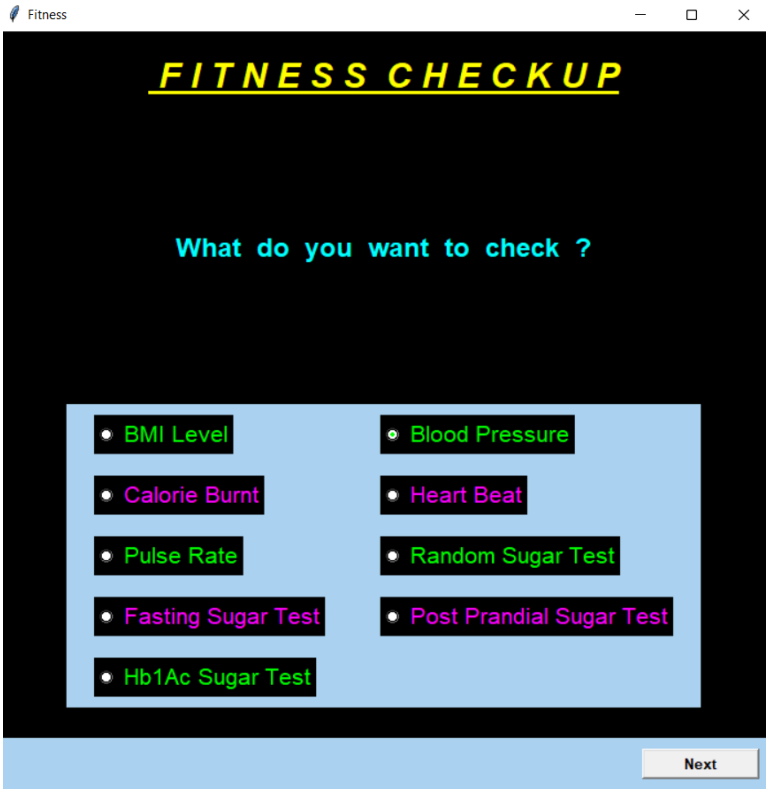
Testing

There are 9 test cases, among these there are 6 major cases among which we have 4 sub cases belonging to sugar test:

1. *BMI test*
2. *Blood Pressure test*
3. *Calorie Burn test*
4. *Heart Beat test*
5. *Pulse Rate test*
6. *Sugar test*
    - 6.1. *Random sugar test*
    - 6.2. *Fasting sugar test*
    - 6.3. *Post Prandial sugar test*
    - 6.4. *Hb1Ac sugar test*

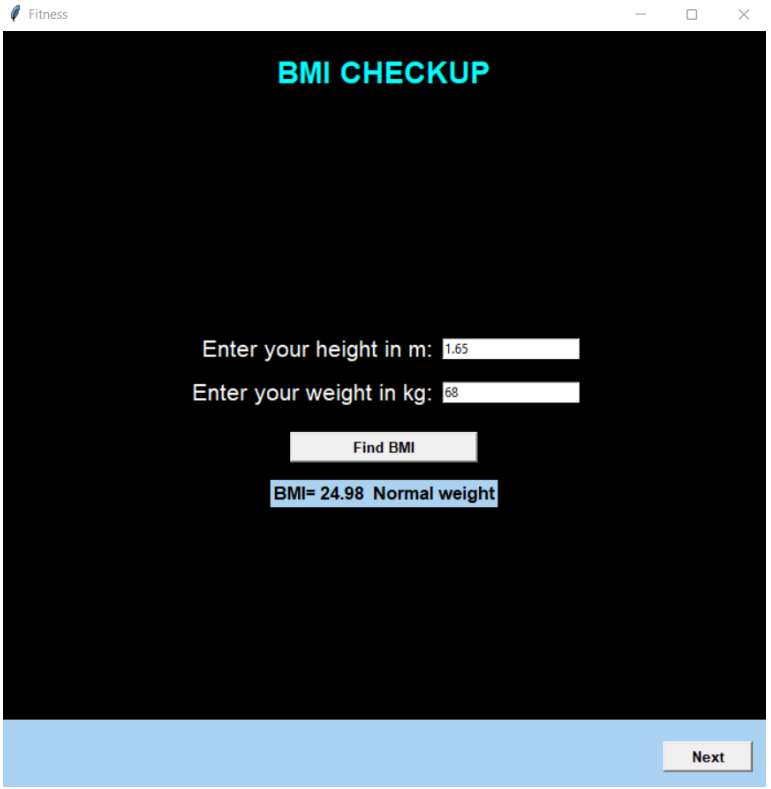
TESTS	***RESULTS***
BMI Test	*It reports the weight condition of a body.*
BP Test	*It reports the BP's stability of a body.*
Calorie Burn Test	*It reports the calorie burn for a specific time in the body.*
Heart Beat Test	*It reports the heart condition of a body.*
Pulse Rate Test	*It reports the pulse condition of the body.*
Random sugar test	*It reports the sugar level condition in the current time of a body.*
Fasting sugar test	*It reports the sugar level condition after a fasting period of the body.*
Post Prandial sugar test	*It reports the sugar level condition after a specific time of heavy meal of the body.*
Hb1Ac sugar test	*It reports the sugar level condition for the past 2 to 3 months of a body.*

Result and Analysis



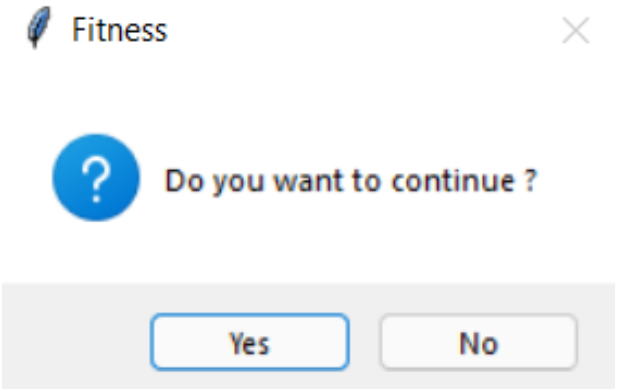
Main Page

(It will be appearing initially.)



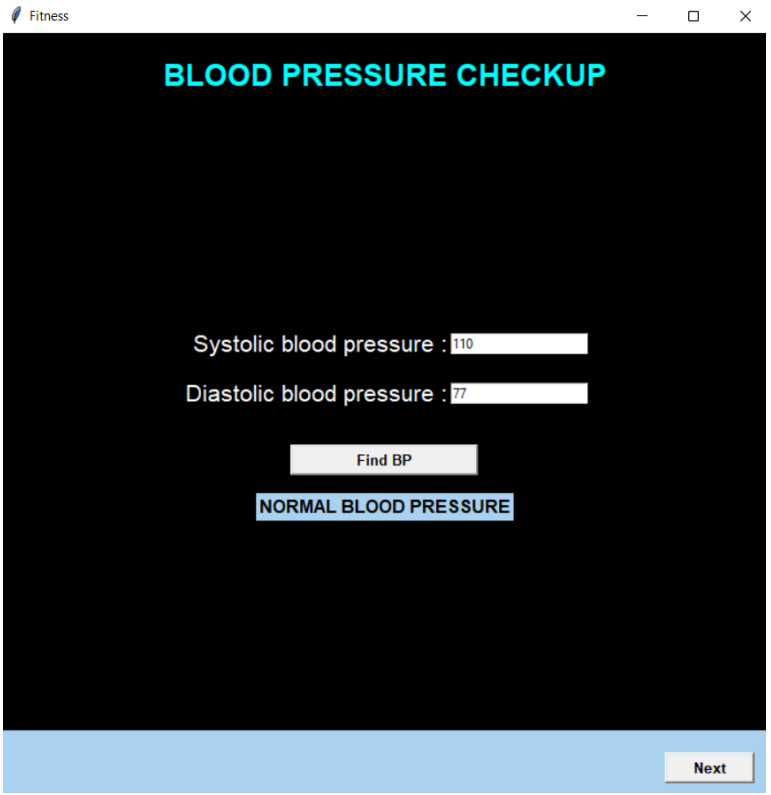
BMI Test

(It will appear on selecting "BMI Level" in the Main Page.)



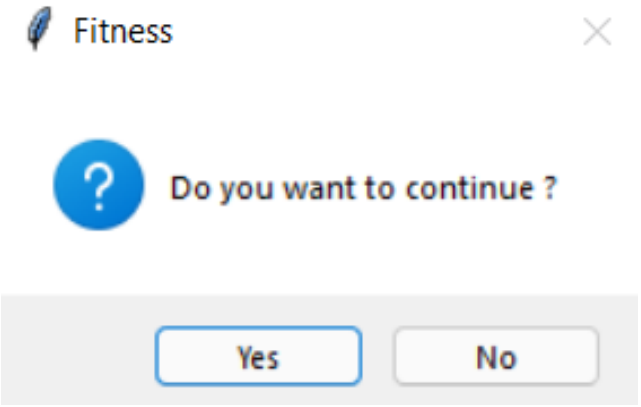
Decision Box

(It will appear by Clicking the "Next" button.)



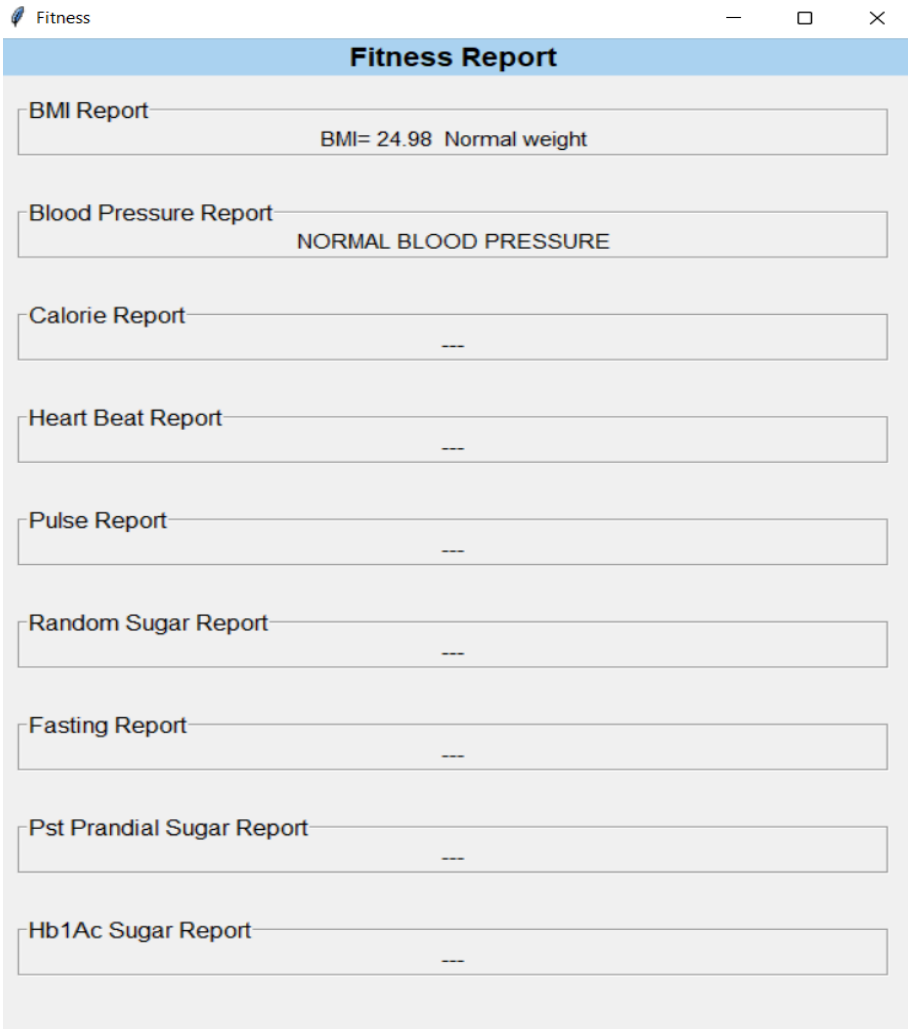
BP Test

(It will appear by clicking on "Yes" in Decision Box and selecting "Blood Pressure" in the Main Page.)



Decision Box

(It will appear by Clicking the "Next" button.)



Final Report

(It will appear by clicking on "No" in the Decision Box.)

Conclusion & Future Enhancement

The project module reports fitness, we can input more than one parameters and produce a report. With the help of this module individuals can assess their health in no time and avoid unnecessary panic.

The code has undergone various testings increasing the readability and making it versatile.The code can be extended to enhance automation. It provides flexibility and can be incorporated in different modules.

This module seeks numerous implementations:

- *Scope of this project can be extended within hospital setups. providing a module as such for hospital examinations to diagnose patients initially and maintain records.*

- *The code can also be used within weight reduction and body building applications wherein a host can monitor the fluctuation in any parameter and provide diets or exercises assessing the same.*

References

- *https://www.geeksforgeeks.org/python-tkinter-tutorial/*
- *https://www.codemy.com/*
- *https://www.programiz.com/python-programming*