PROJECT REPORT

On

Project Name: FITNESS CALCULATOR

Course Code: INT 213

Faculty: Navpreet

Rupal

B.TECH

in

COMPUTER SCIENCE AND ENGINEERING

Team Members:

S no	Name	Registration number	Roll No.
1	SACHIN PRAJAPATI	12114973	RK21 QTB 47
2	SATHI AKSHAY KUMAR REDDY	12115152	RK21 QTA 23
3	NARAYANA SRIKANTH	12112069	RK21 QTB 72

INTRODUCTION:

Project allocated to our team is Fitness Calculator. It is a Single User

This Mini project of "Fitness Calculator" is purely made in python, with a good user-friendly interface which lets the user to enter the various values required to be input in order to calculate how fit the person is.

The main window consists of all the entry fields that the user is required to fill in order to generate a report. This main window also has a button to generate report when all the entry box has the required values.

Program which generates report of the particular person with following labels:

- BMI
- Blood Pressure
- Pulse Rate
- RBC, WBC, Platelets Count
- Hemoglobin level
- Uric Acid level
- Cholesterol level

and categorize it as high, low and medium.

These different values are colored according to how harmful they are on the individual's health. This makes it easier for the user to view the report, moreover "tk" module of tkinter library is used to show these values as a progress making it easier for the person using the app.

LIBRARIES USED:

This project uses tkinter library which is used to make the G.U.I. Some modules of tkinter used are buttons, entry box, labels and progress bars.
All here are used to represent the report graphically to the user.

HARDWARE:

As a part of hardware, the user should have a laptop or desktop computer which is capable of handling python idle. It should also have minimum of 2GB ram for the easy compilation and smooth running of the program.

LOGIC USED:

Program used a bunch of if statements used to decide the output according to the input given by user.
This is the basic logic on which the program works.

The calculation of BMI and other values is done by using formulas and data collected from various websites is used to calculate the strength of a particular value in terms of High low or medium.

There are various functions and if statement which also control what color to change for the label when a certain value is entered.

BASIS OF CALCULATION MODULES:

$$BMI = \frac{weight\ (kg)}{height\ (m^2)}$$

BMI	Weight class
below 18.5	underweight
18.5 - 24.9	normal
25.0 - 29.9	overweight
30.0 and up	Very overweight

Blood Pressure Categories

BLOOD PRESSURE CATEGORY	SYSTOLIC mm Hg (upper number)		DIASTOLIC mm Hg (lower number)
NORMAL	LESS THAN 120	and	LESS THAN 80
ELEVATED	120 – 129	and	LESS THAN 80
HIGH BLOOD PRESSURE (HYPERTENSION) STAGE 1	130 – 139	or	80 – 89
HIGH BLOOD PRESSURE (HYPERTENSION) STAGE 2	140 OR HIGHER	or	90 OR HIGHER
HYPERTENSIVE CRISIS (consult your doctor immediately)	HIGHER THAN 180	and/or	HIGHER THAN 120

Age	Normal Pulse Rate
Neonate (<28 days)	100 - 205
Infant (1 month – 1 year)	100 - 190
Toddler (1 – 2 years)	98 - 140
Pre-school (3 – 5 years)	80 - 120
6 – 11 years	75 - 118
12 years – adult	60 – 100
Athlete	40 - 60

Table 5. Normal ranges of full blood count. (SI unit)

Parameter	Normal Range (Male)	Normal Range (Female)
Red Blood Cell (RBC)	$4.50 – 6.50 \times 10^{12}/L$	$3.80 – 5.80 \times 10^{12}/L$
White Blood Cell (WBC)	$3.60 – 11.0 \times 10^{9}/L$	$3.60 – 11.0 \times 10^{9}/L$
Platelet	$150 – 450 \times 10^{9}/L$	$150 – 450 \times 10^{9}/L$

THE CODE:

```
from tkinter
import *
import tkinter
import tkinter.messagebox

def
  main()
  :
  top1=
  Tk()
  top1.geometry('400x300')
  name=str(E1.get())
  L00 = Label(top1, text=name +" Your
      Medical Report is:",fg="brown",
      font="Times").grid(row=0,column=0)
  """_____BMI_____ """
  l12=Label(top1,text="BMI
  :",fg="red")
  l12.grid(row=1,column=0)
  E12=Entry(top1,bd=5)
  E12.grid(row=1,column=1
  ) weight1=int(E3.get())
  height1=float(E4.get())
  age1=int(E2.get())
  bmi1=weight1/(height1*
  height1)
  E12.insert(0,bmi1)
  E12['bg']='orange'


  """_____BP_____ """

  L11=Label(top1,text="BP(HIGH/MEDIUM/LOW):",fg="
```

```python
red") L11.grid(row=2,column=0)
E11=Entry(top1,bd=5)
E11.grid(row=2,column=1
)bplow1=int(E5.get())
if (bplow1<=90):
  E11.insert(0,"Low")
  E11['bg']='red'
elif  (bplow1>90 and
  bplow1<120):
  E11.insert(0,"Medium
  ") E11['bg']='orange'
else:
  E11.insert(0,"High")
  E11['bg']='green'


"""_____Pulserate_____"""

L22=Label(top1,text="Pulse
Rate(HIGH/MEDIUM/LOW):",fg="red")
L22.grid(row=3,column=0)
E22=Entry(top1,bd=5)
E22.grid(row=3,column=1
)
```

```python
pulserate1=int(E6.get())
if(pulserate1<60):
    E22.insert(0,"low")
    E22['bg']='red'
elif (pulserate1>60 and
    pulserate1<100):
    E22.insert(0,"Medium")
    E22['bg']='orange'
else:
    E22.insert(0,"High")
    E22['bg']='green'

"""_____RBC count_____"""

L33=Label(top1,text="RBC
Count(HIGH/MEDIUM/LOW):",fg="red")
L33.grid(row=5,column=0)
E33=Entry(top1,bd=5)
E33.grid(row=5,column=1
)rbccount1=int(E7.get())
if(rbccount1<475000):
    E33.insert(0,"Low")
    E33['bg']='red'
elif(rbccount1>475000 and
    rbccount1<610000):
    E33.insert(0,"Medium")
    E33['bg']='o
range'else:
    E33.insert(0,"High")
    E33['bg']='green'

"""_____WBC COUT_____"""

L44=Label(top1,text="WBC
Count(HIGH/MEDIUM/LOW):",fg="red")
L44.grid(row=6,column=0)
E44=Entry(top1,bd=5)
E44.grid(row=6,column=1
```

```python
)wbccount1=int(E8.get())
if(wbccount1<4000):
  E44.insert(0,"Low")
  E44['bg']='red'
elif(wbccount1>4000 and
  wbccount1<10000):
  E44.insert(0,"Medium")
  E44['bg']='o
range'else:
  E44.insert(0,"High")
  E44['bg']='green'

"""_____PLALATES_____"""
L55=Label(top1,text="Platelets(HIGH/MEDIUM/LOW):
",fg="red") L55.grid(row=7,column=0)
E55=Entry(top1,bd=5)
E55.grid(row=7,column=1
)platelets1=int(E9.get())
if(platelets1<150000):
  E55.insert(0,"Low")
```

```python
    E55['bg']='red'
elif (platelets1>150000 and
    platelets1<450000):
    E55.insert(0,"Medium")
    E55['bg']='o
range'else:
    E55.insert(0,"High")
    E55['bg']='green'
```

"""_____HEMOGLOBIN_____"""

```python
L66=Label(top1,text="HB(HIGH/MEDIUM/LOW):",fg="
red") L66.grid(row=8,column=0)
E66=Entry(top1,bd
=5)
E66.grid(row=8,col
umn=1)
hb1=int(E10.get())
if(hb1<12):
    E66.insert(0,"Low")
    E66['bg']='red'
elif(hb1>12 and
hb1<16):
    E66.insert(0,"Medium")
    E66['bg']='orange'
else:
    E66.insert(0,"High")
    E66['bg']='green'
```

"""_____URIC ACID_____"""
```python
L77=Label(top1,text="Uric
Acid(HIGH/MEDIUM/LOW):",fg="red")
L77.grid(row=9,column=0)
E77=Entry(top1,bd=5)
E77.grid(row=9,column=1
)
uricacid1=int(E50.get())
```

```python
if(uricacid1<4):
  E77.insert(0,"Low")
  E77['bg']='red'
elif(uricacid1>4 and
  uricacid1<7):
  E77.insert(0,"Mediu
  m")
  E77['bg']='orange'
else:
  E77.insert(0,"High")
  E77['bg']='green'

"""_____CHOLESTROL_____"""
L88=Label(top1,text="Cholestrol(HIGH/MEDIUM/LOW
):",fg="red") L88.grid(row=10,column=0)
E88=Entry(top1,bd=5)
E88.grid(row=10,column=
1)
cholestrol1=int(E51.get())
if(cholestrol1<40):
  E88['bg']='red'
  E88.insert(0,"low")
elif(cholestrol1>40 and
cholestrol1<50):
```

```python
        E88.insert(0,"Medium")
        E88['bg']='orange'
    else:
        E88.insert(0,"High")
        E88['bg']='green'


    top1.mainl

oop()top =

Tk()

L0 = Label(top, text="Fitness
        Calculator", fg="purple",
        font="Times").grid(row=
        0,column=1)

L1=Label(top,text=
        "Name",
        fg="red",
        font="Times").grid(row=
1,column=0)E1=
Entry(top,bd=5)
E1.grid(row=1,column=1)

L2=Label(top,text
        ="age ",
        fg="red",
        font="Times").grid(row=2,
column=0)E2=
Entry(top,bd=5)
E2.grid(row=2,column=1)
l3=Label(top,text="weight (in
Kg)",fg="red",font="Times").grid(row=3,column=0)E3=
Entry(top,bd=5)
E3.grid(row=3,column=1)
```

```python
l4=Label(top,text="Height (in
M)",fg="red",font="Times").grid(row=4,column=0)
E4=Entry(top,bd=5)
E4.grid(row=4,column=1)
l5=Label(top,text="Bp (0-
120)",fg="red",font="Times").grid(row=5,column=0)
E5=Entry(top,bd=5)
E5.grid(row=5,column=1)
L6=Label(top,text="Pulser
ate (0-100)",
  fg="red",
  font="Times").grid(row=6,
  column=0)
E6= Entry(top,bd=5)
E6.grid(row=6,column=1)
L7=Label(top,text="RBCcount (
    310000-610000)",fg="red",
    font="Times").grid(row=7,col
    umn=0)
E7 = Entry(top,bd=5)
E7.grid(row=7,column=1)
L8=Label(top,text="WBCcount
(2000-10000)",
    fg="red",
    font="Times").grid(row=8,
    column=0)
E8= Entry(top,bd=5)
E8.grid(row=8,column=1)
L9=Label(top,text="Platelets
    (150000-615000)",fg="red",
    font="Times").grid(row=10,c
    olumn=0)
```

```python
E9 = Entry(top,bd=5)
E9.grid(row=10,column=1)
L10=Label(top,text="HEMOG
LOBIN(0-16)",
    fg="red",
    font="Times").grid(row=11,col
    umn=0)
E10 = Entry(top,bd=5)
E10.grid(row=11,column=1)

L50=Label(top,text="URIC ACID
    (0-7)", fg="red",
    font="Times").grid(row=12,
    column=0)
E50 = Entry(top,bd=5)
E50.grid(row=12,column=1)
L51=Label(top,text="CHOLES
TROL(40-55)",
    fg="red",
    font="Times").grid(row=13,col
    umn=0)
E51 = Entry(top,bd=5)
E51.grid(row=13,column=1)

def
  clear_textbo
  x():
  E1.delete(0,
  END)
  E2.delete(0,
  END)
  E3.delete(0,
  END)
  E4.delete(0,
  END)
  E5.delete(0,
  END)
  E6.delete(0,
```

```python
          END)
  E7.delete(0,
  END)
  E8.delete(0,
  END)
  E9.delete(0,
  END)
  E10.delete(0
  , END)
  E50.delete(0
  , END)
  E51.delete(0
  , END)
def
  close_windo
  w ():
  top.destroy(
  )
D=Label(top,text="Developed By:-
",fg='green')
D.grid(row=19,column=0)
D1=Label(top,text="Nishitha
Neelam",fg='brown')
D1.grid(row=19,column=1)
D2=Label(top,text="Ashutosh
Shukla",fg='red')
D2.grid(row=19,column=2)
D2=Label(top,text="Ashish
Yadav",fg='purple')
D2.grid(row=19,column=3)
B14=Button(top, text='RESET ALL
ENTRIES',command=clear_textbox,bg="red")
B14.grid(row=14,column=0)

B15=Button (top,text =
"EXIT",command=close_window,bg="yellow")
B15.grid(row=14,column=2)
```

```
B1=Button(top, text='Generate
Report',command=main,bg="orange")
B1.grid(row=13,column=2)
top.mainloop()
```

SCREEN SHOTS:

Fitness Calculator

Fitness			
Name :	Sachin Prajapati	Age :	18
Gender :	⦿ Male		○ Female
Weight (Kg) :	50		
Height (Mts) :	1.72		
BP Low (Systolic) mm/Hg :	120		
BP High (Diastolic) mm/Hg :	85		
Pulse Rate :	73		
RBC Count (trillion Cells/L) :	4.50		
WBC Count (billions cells/L) :	4		
Platelets :	155		
HB :	14		
Uric Acid (mg/dL) :	5		
Cholestrol (mg/dL) :	180		

OBJECTIVE:

Creating worldwide health platform to store and generate report of a person's health. If someone's health goes bad on this platforms, different health organizations aka Hospitals can tap that person and help him to get to know the symptoms.

As the Future, the young generation of the world we have to take care of our health. Knowing the disease faster makes us having more chances of curing it more efficiently.

CONCLUSION:

Being Mental, Emotional, Physically Fit and aware is extremely important in this age. Pollution, sickness is one of the main causes of death today.

Reaching and maintaining a healthy body is important for overall health and can help you prevent and control many diseases and conditions. If your levels are LOW or HIGH, you are at high risk of developing serious health problems, including heart disease, high blood pressure, type 2 diabetes, gallstones, breathing problems, and certain cancers. That is why maintaining a healthy body is so important: It helps you lower your risk for developing these problems, helps you feel good about yourself, and gives you more energy to enjoy life.

REFERENCES:

WWW.GOOGLE.COM
WWW.YOUTUBE.COM
WWW.GITHUB.COM