A REPORT OF SIX WEEKS INDUSTRIAL TRAINING

At

Think NEXT Technologies Private Limited

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD

OF THE DEGREE OF

**BACHELOR OF ENGINEERING**

(Computer Science & Engineering)



MAY-JUNE, 2019

**SUBMITTED BY:**

NAME:

UNIVERSITY UID:

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH UNIVERSITY GHARUAN, MOHALI

**CHANDIGARH UNIVERSITY, GHARUAN, MOHALI**

**CANDIDATE'S DECLARATION**

I “GURKIRAN SINGH GILL” hereby declare that I have undertaken six weeks industrial training at “ThinkNEXT Technologies Private Limited” during a period from \_\_\_\_\_\_ to \_\_\_\_\_\_\_ in partial fulfillment of requirements for the award of degree of B.E (COMPUTER SCIENCE & ENGINEERING) at CHANDIGARH UNIVERSITY GHARUAN, MOHALI. The work which is being presented in the training report submitted to the Department of Computer Science & Engineering at CHANDIGARH UNIVERSITY GHARUAN, MOHALI is an authentic record of training work.

Signature of the Student

 The six weeks industrial training Viva–Voce Examination of\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ has been held on \_\_\_\_\_\_\_\_\_\_\_\_ and accepted.

 Signature of Internal Examiner Signature of External Examiner

**ABSTRACT**

Owing to Harris and Benedict’s success, TDEE (Total Daily Energy Expenditure) has proved to be an indispensable tool to calculate energy consumption (in terms of calories) to gain, lose or maintain body weight. This simple formula (after extensive research and hard work) has curbed the fitness world’s caloric intake.

Fitness enthusiasts have benefited tremendously from this calculation technique, so why not put this technique at everyone’s disposal. This project aims to prove this vision true and a reality. Combining this complex technique and a comprehensible GUI now anyone can keep a track of caloric intake. Be it getting back into shape or being consistent. On your way to achieve a goal or getting help starting one, a simple platform to calculate your calories will help you in doing so.

Being versatile and a new language preferred by a majority of developers, Python was chosen for developing this project. An abundance of Python tools and libraries can be seen playing a part in this project. A special Python library Tkinter is used to enable the user in an effortless manner. Some of the features include monitoring weight, height, and physical goals.

**ACKNOWLEDGEMENT**

I would like to express my gratitude to Chandigarh University and the Computer Science Department for granting me this delightful opportunity to work on this project. I’d also like to thank Think NEXT Technologies Private Limited for facilitating the tools and training to be able to work on this project which has helped me not only to learn about the technologies used, but also helped me to do research about nutrition and health to get a better understanding for completing this project.

A special thanks to Rahul Kumar Sir (Think NEXT Technologies Python Instructor and trainer) for always being available at our disposal whenever needed. Not forgetting my training mates, thank you all for volunteering in the survey collected to attain activity levels and selective data.

I’d also like to use this opportunity to all those people who showed keen interest in the testing and development of this project to make it a great success.

A great thankyou to those people who didn’t have a clue about the workings of this project yet showered me with all the support I needed to put my best work in.

I hope that this project surpasses the quality required for the project.

**About ThinkNEXT Technologies Private Limited**

ThinkNEXT Technologies Private Limited is an education based company which takes partakes in training and equipping students with a wide variety of tools and technologies. With nine years of experience in the education domain, ThinkNEXT Technologies is rapidly excelling in its field. Working Closely with Universities, students and authorities has helped build the base of teaching provided by the company. In an era where getting hands on the latest technologies is essential, ThinkNEXT Technologies has put in an immense effort to help achieve that status.

The company provides Industrial training in popular programming languages such as Python, JAVA, C++ to major fields such as Big Data Analytics, Machine Learning, Digital Marketing and Web Development just to name a few.

A wide majority of the company’s focus is towards new companies/startups help get a kick start by helping them set up an online presence through websites, apps, and digital marketing.

Internship positions are also available for students looking to get their hands on the latest training methods being taught.

**CONTENTS**

**Topic Page Number**

**Training Certificate i**

**Candidate’s Declaration ii**

**Abstract iii**

**Acknowledgement iv**

**About ThinkNEXT Technologies Private Limited v**

1. **Introduction 1-4**
   1. **Training Background 1**
   2. **Python 2**
   3. **Tkinter 2**
   4. **TDEE 3**
   5. **Software/Hardware Requirements 4**
2. **Training Work Undertaken 5-6**
   1. **Brainstorming 5**
   2. **Research 5**
   3. **Blueprint 5**
   4. **Coding, GUI, and Binding 6**
   5. **Debugging 6**
3. **Results and Discussion 7-14**
   1. **Figure 1.1 8**
   2. **Figure 1.2 9**
   3. **Source code 10-14**
4. **Conclusion and Future scope 15**

**Bibliography 16**

**INTRODUCTION**

1. **Training Background**

The industrial training partaken has revolved around the popular programming languages called Python. Learning the basics and the structures of a Python program kick started the process of becoming familiar with the language. The topics learned during the training include: *statements, variables, functions, data types, operators, control flow* and *classes* along with a few arbitrary topics.

By setting up a strong base about the language, the training commenced with learning of the Python library called Tkinter (which is used for the development of this project) used for creating interactive GUIs. The library uses features similar to HTML and is quite easy to learn and understand. The implementation can prove to be a bit difficult, however with consistent use can be gotten accustomed to. Some of the topics covered during the training include: *basics, frames, labels, packing methods, buttons, entries, menu bars, scaling* and most importantly *binding functions*.

In order to make a project worth the work, one has to do extensive research about it, which is exactly what was proposed to me. Looking at the facts, calculations, and the tedious amount of hard work done by the people in the field not only excited me but motivates me till date to do my own. Starting out for the project was a bit of a challenge at first, however, I thought to myself, *“why not make a project on a topic which I’m genuinely interested in?”*. That become my fortitude and I started researching about a convenient way to calculate the number of calories and macros needed from a basal point of view. That’s when I came across TDEE, which is basically a convenient method developed by Harris and Benedict to calculate the *‘Total Daily Energy Expenditure’* in terms of calories.

1. **Python**

Python was created in 1991 by Guido van Rossum. The sole purpose for the creation of Python was to enable anyone to easily code which is why its syntax is one of the easiest to learn and use. It’s an open-source, high-level language which can be quite extensively used and supports multiple programming paradigms such as procedural, object-oriented and functional programming. Python is also an interpreted language which makes it a little easier to debug, however they are slower than compiled programs. Being versatile in many fields, this proves to be the ideal language to start for a beginner.

A few of its perks are:

* It’s easy to learn and code.
* It’s constantly upgrading.
* It easily integrates with other languages and frameworks.
* Increases productivity and efficiency with its flexible and scalable nature.

Some of the fields which use Python are *machine learning, data mining, code web scraping,* and *web development* just to name a few.

1. **Tkinter**

Tkinter is the standard library in Python for making a GUI. It’s the most favored library for making a GUI in Python. Due to its high compatibility with Python it performs extremely fast for outputs.

A few steps to create a GUI using Tkinter are:

* Importing the Tkinter module.
* Creating the main window for the GUI.
* Adding Widgets for increasing functionality.
* .Enter the main event loop to take action against each event triggered by the user.

We can add widgets to the main window or a specific frame for increasing the functionality such as: buttons, frames, canvas, menu, labels, entries, scale, text and a spin box.

1. **TDEE**

A basic knowledge has been established for weight loss and muscle gain. If one wants to take part in weight loss, they should create a caloric deficit and for gaining muscle mass one should create caloric surplus.

For creating these conditions, one needs to monitor their nutrition progressively and there are many methods to do so. One such method is TDEE, which stands for *‘Total Daily Energy Expenditure’.* This terms dictates the amount of calories required for a person at a specific activity level to maintain their current body weight.

One crucial aspect for Calculating TDEE is BMR, which stands for Basal Metabolic Rate. This term identifies the caloric intake for a person for their body’s maintenance at rest, which is also known as metabolism. Now, to calculate the TDEE of a person, we need to find out their activity level, which can be categorized into 4 or 5 types. Depending upon their current activity level, a TDEE is calculated by multiplying BMR with the value associated with the activity level.

It’s advisable to check your TDEE once every month while monitoring caloric intake. For weight loss, one can subtract 500 calories from their TDEE value to create caloric deficit. Also, for creating caloric surplus, one can add 500 calories to their TDEE value.

1. **Software/Hardware Requirements**

The minimum requirements for creating and executing this project were:

- An IDE to develop the program.

- An internet connection or written sources for research.

- A computer with a minimum i3 processor.

- Copious amounts of coffee.

- Finally, a source of power such as a will to learn and keep you going.

**TRAINING WORK UNDERTAKEN**

**Brainstorming**

Mid-way through the second week of our training, I started brainstorming about the possible ideas suitable for making a project. The reason for me to make this project is solving a basic problem which I personally face. Although this project isn’t that advanced, however it takes care of the basic functionality needed. The particular reason for choosing to work on this was to find something that I not only need/have to create, but would also love working on developing into a good tool in the future for my personal use.

**Research**

Despite the simplicity of the calculator, an extensive amount of research was needed to deeply understand the building block of this calculation method. In order to do that, copious amounts of online sources have been visited. The research also helped in choosing the suitable framework for creating the GUI and the language for coding.

**Blueprint**

After the research was over, it was time to move onto deciding what functionality the project would anchor. That was achieved by creating a rough idea of how the application would look like and perform. The binding of the code with the widgets was also taken care of by using test designs, since hard coded programs have to be customized to cater to the needs of the GUI.

**Code, GUI and binding.**

It took approximately 31.6 hours to code the entire project and 2 weeks for research. The coding was a bit tricky to implement, due to the fact that Python doesn’t use switch cases unlike C/C++, which only meant that if-else would be the best option other than a dictionary. The structure of the GUI constitutes of the main window with frames and widgets, with an infinite loop to keep the window open. The most difficult part as a beginner was finding out a way to effectively bind the GUI with the functions to perform the calculations. The code is divided into five parts:

- The first part includes importing the Tkinter library as well as setting up the main window with two frames.

- In the second part, the formulae to construct the TDEE is coded using basic Python statements.

- Further, we have the first info section which includes setting labels and entries for the input to calculate the TDEE.

- The second info section constitutes of the activity level and the final TDEE output.

- Finally, the last section is constructed of the buttons at the bottom of the GUI.

**Debugging**

Debugging was done to take care of function calls and invalid function parameters occurring due to different data types. By fixing this issue, the program functions effectively.

**RESULTS AND DISCUSSION**

The calculator proves to be a vital tool for monitoring a wide variety of progress made through nutrition. These include:

- Weight

- Height

- Caloric intake

More input available could definitely help evolving the prospect of calculation to obtain accurate results by taking into account other factors such as:

- Body fat

- BMI

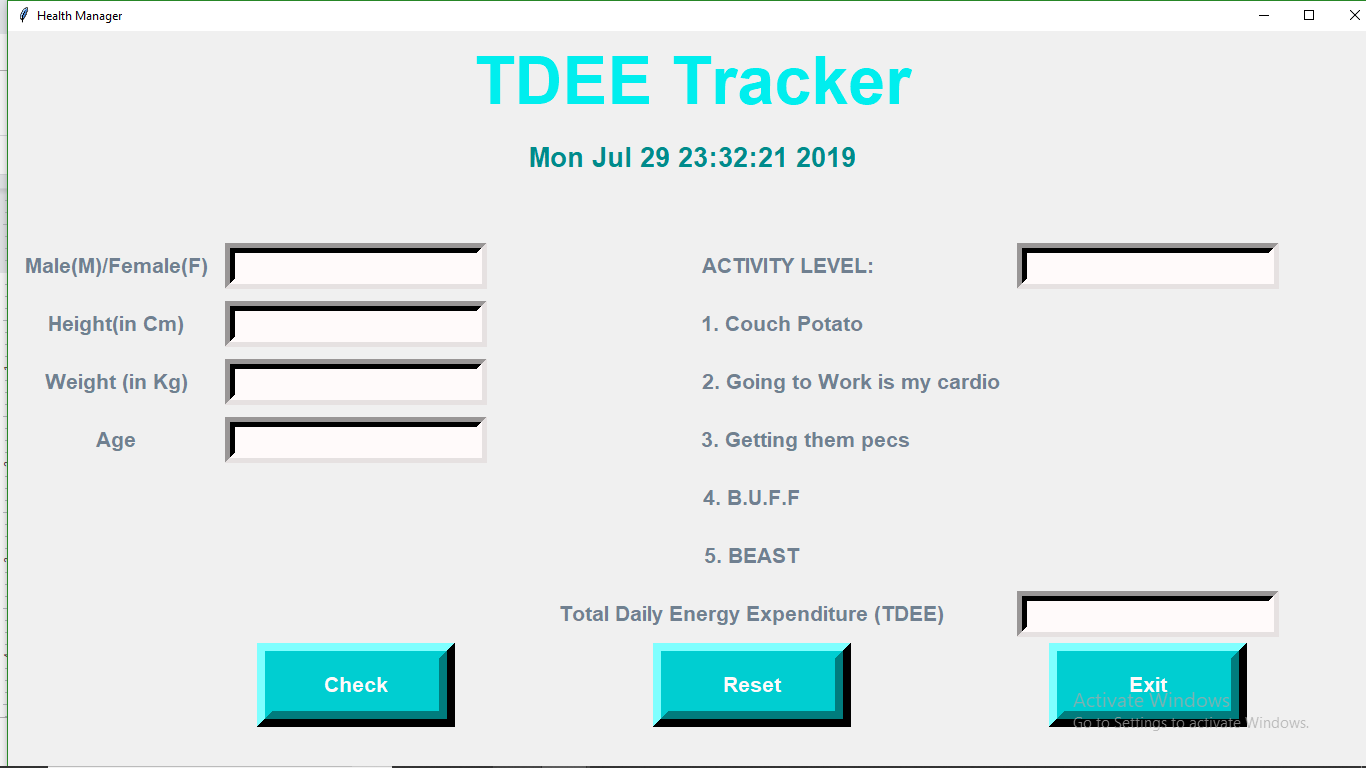
- Bone density

- Metabolic rate

- Nutrient dense foods.

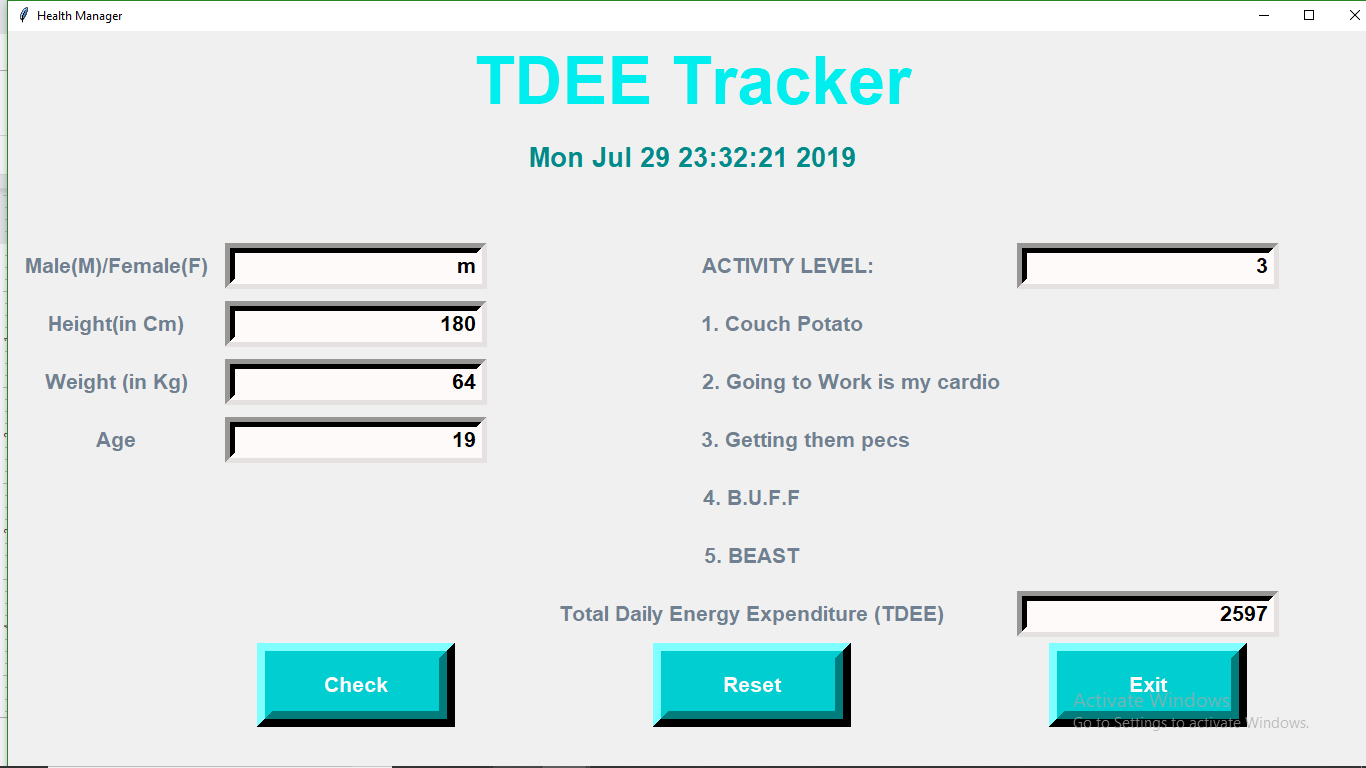
By using Python and its valuable GUIs, one can develop applications and programs with variant complexities and versatile functionalities. Creating projects like these, not only helps to polish programming skills, but also helps in evolving problem solving thinking by working on bugs and problems faced while creation of the program in a real-time environment.

The given pictures help get the basic idea of how a GUI can look for similar programs:



**Figure : 1.1**

**Sample input:** Male, 180 cm (height), 64 kg (weight), 19 years, Activity Level = 3.



**Figure : 1.2**

**SOURCE CODE :**

from tkinter import\*

import time

import datetime

root=Tk()

root.geometry("1600x8000")

root.title("Health Manager")

Tops=Frame(root, width=1600,relief=SUNKEN)

Tops.pack(side=TOP)

f1=Frame(root,width=800,height=700,relief=SUNKEN)

f1.pack(side=LEFT)

localtime=time.asctime(time.localtime(time.time()))

lblInfo=Label(Tops,font=('arial',50,'bold'),text="TDEE Tracker",fg="cyan2",bd=10,anchor='w')

lblInfo.grid(row=0,column=0)

lblInfo=Label(Tops,font=('arial',20,'bold'),text=localtime,fg="cyan4",bd=10,anchor='w')

lblInfo.grid(row=1,column=0)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

def bmr():

g = str(gender.get())

h = int(height.get())

w = int(weight.get())

a = int(age.get())

H = h \* 6.25

W = w \* 9.99

A = a \* 4.92

if g == 'm' or g == 'M':

bmr = H + W - A + 5

elif g == 'f' or g == 'F':

bmr = H + W - A - 161

tdee = aLevel(bmr,g)

return tdee;

def aLevel(bmr,g):

aLevel = int(level.get())

if g == 'm' or g == 'M' :

if aLevel == 1:

tdee = 1.2 \* bmr

elif aLevel == 2:

tdee = 1.375 \* bmr

elif aLevel == 3:

tdee = 1.55 \* bmr

elif aLevel == 4:

tdee = 1.725 \* bmr

elif aLevel == 5:

tdee = 1.9 \* bmr

if g == 'f' or g == 'F':

if aLevel == 1:

tdee = 1.1 \* bmr

elif aLevel == 2:

tdee = 1.275 \* bmr

elif aLevel == 3:

tdee = 1.35 \* bmr

elif aLevel == 4:

tdee = 1.525 \* bmr

elif aLevel == 5:

tdee = 1.725 \* bmr

return(int(tdee))

def Ref():

cal = bmr()

Total.set(cal)

def Reset():

gender.set("")

height.set("")

weight.set("")

age.set("")

level.set("")

Total.set("")

def qExit():

root.destroy()

#====================================Info1==========================================

gender = StringVar()

height=StringVar()

weight=StringVar()

age=StringVar()

level=StringVar()

Total=StringVar()

lblgender= Label(f1, font=('arial', 16, 'bold'),text="Male(M)/Female(F)",bd=16,fg="slate gray",anchor="w")

lblgender.grid(row=0, column=0)

txtgender=Entry(f1, font=('arial',16,'bold'),textvariable=gender,bd=10,insertwidth=4,bg="snow",justify='right')

txtgender.grid(row=0,column=1)

lblheight= Label(f1, font=('arial', 16, 'bold'),text="Height(in Cm)",bd=16,fg="slate gray",)

lblheight.grid(row=1, column=0)

txtheight=Entry(f1, font=('arial',16,'bold'),textvariable=height,bd=10,insertwidth=4,bg="snow",justify='right')

txtheight.grid(row=1,column=1)

lblweight= Label(f1, font=('arial', 16, 'bold'),text="Weight (in Kg)",bd=16,fg="slate gray",anchor="w")

lblweight.grid(row=2, column=0)

txtweight=Entry(f1, font=('arial',16,'bold'),textvariable=weight,bd=10,insertwidth=4,bg="snow",justify='right')

txtweight.grid(row=2,column=1)

lblage= Label(f1, font=('arial', 16, 'bold'),text="Age",bd=16,fg="slate gray",anchor="w")

lblage.grid(row=3, column=0)

txtage=Entry(f1, font=('arial',16,'bold'),textvariable=age,bd=10,insertwidth=4,bg="snow",justify='right')

txtage.grid(row=3,column=1)

lbl8= Label(f1, font=('arial', 16, 'bold'),text="",bd=16,anchor="w")

lbl8.grid(row=4, column=0)

lbl9= Label(f1, font=('arial', 16, 'bold'),text="",bd=16,anchor="w")

lbl9.grid(row=5, column=0)

#============================= INFO 2===============================================

lbl= Label(f1, font=('arial', 16, 'bold'),text=" ACTIVITY LEVEL:",bd=16,fg="slate gray",anchor="w")

lbl.grid(row=0, column=2)

txtlevel=Entry(f1, font=('arial',16,'bold'),textvariable=level,bd=10,insertwidth=4,bg="snow",justify='right')

txtlevel.grid(row=0,column=3)

lbl1= Label(f1, font=('arial', 16, 'bold'),text=" 1. Couch Potato",bd=16,fg="slate gray",anchor="w")

lbl1.grid(row=1, column=2)

lbl2= Label(f1, font=('arial', 16, 'bold'),text=" 2. Going to Work is my cardio",bd=16,fg="slate gray",anchor="w")

lbl2.grid(row=2, column=2)

lbl3= Label(f1, font=('arial', 16, 'bold'),text=" 3. Getting them pecs",bd=16,fg="slate gray",anchor="w")

lbl3.grid(row=3, column=2)

lbl4= Label(f1, font=('arial', 16, 'bold'),text="4. B.U.F.F",bd=16,fg="slate gray",anchor="w")

lbl4.grid(row=4, column=2)

lbl5= Label(f1, font=('arial', 16, 'bold'),text="5. BEAST",bd=16,fg="slate gray",anchor="w")

lbl5.grid(row=5, column=2)

lblTotalCost= Label(f1, font=('arial', 16, 'bold'),text="Total Daily Energy Expenditure (TDEE)",bd=16,fg="slate gray",anchor="w")

lblTotalCost.grid(row=6, column=2)

txtTotalCost=Entry(f1, font=('arial',16,'bold'),textvariable=Total,bd=10,insertwidth=4,bg="snow",justify='right')

txtTotalCost.grid(row=6,column=3)

#==========================================Buttons==================================btnTotal=Button(f1,padx=16,pady=8,bd=16,fg="snow",font=('arial',16,'bold'),width=10,text="Check",bg="dark turquoise",command=Ref).grid(row=9,column=1)

btnReset=Button(f1,padx=16,pady=8,bd=16,fg="snow",font=('arial',16,'bold'),width=10,text="Reset",bg="dark turquoise",command=Reset).grid(row=9,column=2)

btnExit=Button(f1,padx=16,pady=8,bd=16,fg="snow",font=('arial',16,'bold'),width=10,text="Exit",bg="dark turquoise",command=qExit).grid(row=9,column=3)

root.mainloop()

**CONCLUSION AND FUTURE SCOPE**

The project, implements the use of Python and Tkinter library along with binding and indenting the code. The calculator fulfills the requirement for calculating calories on a basic level. Each button and entry performs well. A full-fledged implementation of Harris-Benedict Formula can be observed in this project. The resources used and learned from have proved to play a vital role in the development of this project.

Modifications and updating can be and should be done to any piece of software to keep it up to date and more importantly, help in aiding to the user’s requirements and needs. In prospect of that aspect, a macro calculator can be integrated into this calculator for a hands on experience to help maintain caloric intake and nutrition. Other frameworks such as WxPython, Kivy and PyGUI can be used to increase the amount of control over the widgets and data.

Since a majority of the beginners are stuck between ‘*bro science’* and a lack of resources and knowledge, the final aim or goal intended for this project is to develop a fitness app to help monitor macros, tdee, and also give basic meal plans and workout for beginners.

A clean and interactive GUI can be constructed through a Google developed framework called *FLUTTER* which will enable both IOS and Android app development simultaneously by using the versatile and widely growing language named *‘DART’*.

**BIBLIOGRAPHY**

- About ThinkNEXT:

<http://thinknext.co.in/about-us/>

<http://thinknext.co.in/services/>

- Introduction:

- About python:

<https://www.python.org/about/>

<https://en.wikipedia.org/wiki/Python_(programming_language)>

<https://www.analyticsindiamag.com/python-swiss-army-knife-coding-thanks-versatility/>

- Tkinter:

<https://www.youtube.com/watch?v=RJB1Ek2Ko_Y&list=PL6gx4Cwl9DGBwibXFtPtflztSNPGuIB_d>

<https://www.geeksforgeeks.org/python-gui-tkinter/>

<https://www.quora.com/What-are-the-best-ways-to-make-GUIs-in-Python>

- TDEE:

- Formulae:

<https://superfastdiet.com/what-is-tdee/>

<https://www.sailrabbit.com/bmr/>

- Calculators:

<https://www.bodybuilding.com/fun/calculate-your-total-daily-energy-expenditure-tdee.html>

<https://tdeecalculator.net>

<https://www.freedieting.com/nutrient-calculator>

<http://www.exercise4weightloss.com/bmr-calculator.html>