**MCQ QUIZ APPLICATION**

**Python Mini Project Report**

Submitted in partial fulfillment of the requirements for

**Second Year of Engineering (Computer Engineering)**

by:

|  |  |
| --- | --- |
| Karishma Apparao Bairi | TU3F1920014 |
| Shweta Madhukar Kaware | TU3F1920005 |
| Ketki Milind Kulkarni | TU3F1920053 |

**Under the Guidance of**

**Prof. Nilesh Kulal**



**Department of Computer Engineering**

**TERNA ENGINEERING COLLEGE**

Nerul (W), Navi Mumbai 400706

**(University of Mumbai)**

(2020-21)

**Internal Approval Sheet**

****

**TERNA ENGINEERING COLLEGE, NERUL**

**Department of Computer Engineering**

Academic Year 2020-21

**CERTIFICATE**

This is to certify that the python mini project entitled **“MCQ Quiz Application”** is a bonafide work of

|  |  |
| --- | --- |
| Karishma Apparao Bairi | TU3F1920014 |
| Shweta Madhukar Kaware | TU3F1920005 |
| Ketki Milind Kulkarni | TU3F1920053 |
|  |  |

submitted to the University of Mumbai in partial fulfillment of the requirement for the Second Year of Engineering (Computer Engineering).

**Guide Head of Department Principal**

**Approval Sheet**

**Project Report Approval**

This Python Mini Project Report – entitled “**MCQ Quiz Application**” by following students is approved for the ***S.E. in "Computer Engineering"***.

**Submitted by:**

|  |  |
| --- | --- |
| Karishma Apparao Bairi | TU3F1920014 |
| Shweta Madhukar Kaware | TU3F1920005 |
| Ketki Milind Kulkarni | TU3F1920053 |
|  |  |

Examiners Name & Signature:

1.---------------------------------------------------------

2.----------------------------------------------------------

Date: ---------------------------------

Place: ---------------------------------

**Declaration**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

|  |  |  |
| --- | --- | --- |
| Karishma Bairi | TU3F1920014 | --------------------------- |
| Shweta Kaware | TU3F1920005 | --------------------------- |
| Ketki Kulkarni | TU3F1920053 | --------------------------- |
|  |  |  |

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Acknowledgement**

We would like to express our sincere gratitude towards our guide **Prof. Nilesh Kulal** for his help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Dr. Archana Mire (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks **to Dr. L. K. Ragha** our Principal for providing the encouragement and much support throughout our work.

|  |  |  |
| --- | --- | --- |
| Karishma Bairi | TU3F1920014 | --------------------------- |
| Shweta Kaware | TU3F1920005 | --------------------------- |
| Ketki Kulkarni | TU3F1920053 | --------------------------- |

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| Chapter 1 | Introduction | 08 |
|  | 1.1 Aim and Objectives of Project | 09 |
|  | 1.2 Future Scope of the Project | 09 |
|  | 1.3 Organization Of The Report | 10 |
| Chapter 2 | Problem Statement | 11 |
| Chapter 3 | Design and Implementation | 12 |
|  | 3.1 Source Code | 12 |
|  | 3.2 Output | 33 |
| Chapter 4  References | Conclusion | 37  38 |

**Abstract**

The purpose of MCQ Quiz Application is to automate the existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

MCQ Quiz Application, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

**CHAPTER - 1**

**Introduction**

The "MCQ Quiz Application" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. MCQ Quiz Application , as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the informations of Examinations, Students, Courses, Results, Papers. Every MCQ Quiz Application has different Students needs, therefore we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

**1.1 Aim and Objectives of the Project**

The main objective of the Project on MCQ Quiz Application is to manage the details of Students, Examinations, Marks. It manages all the information about Students and Results. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Students, Examinations, Results, Marks. It tracks all the details about the Marks.

Functionalities provided by MCQ Quiz Application are as follows:

• Provides the searching facilities based on various factors. Such as Students.

• MCQ Quiz Application also manages the Results details online, Papers details, Students.

• It tracks all the information of Examinations, Results, ect.

• Manage the information of Examinations.

• Shows the information and description of the Students, Marks.

• To increase efficiency of managing the Students, Examinations.

• It deals with monitoring the information.

• Manage the information of Students.

• Integration of all records of Papers.

**1.2 Future Scope of the Project**

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

• We can add printer in future.

• We can give more advance software for MCQ Quiz Application including more facilities.

• We will host the platform on online servers to make it accessible worldwide

• Integrate multiple load balancers to distribute the loads of the system

• Create the master and slave database structure to reduce the overload of the database queries • Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.

The above mentioned points are the enhancements which can be done to increase the applicability and usage of this project. Here we can maintain the records of Students and Examinations. Also, as it can be seen that now-a-days the players are versatile, i.e. so there is a scope for introducing a method to maintain the MCQ Quiz Application. Enhancements can be done to maintain all the Students, Examinations, Marks, Courses, Papers.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them.ln the last we would like to thanks all the persons involved in the development of the system directly or indirectly. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

**1.3 Organization of the Report**

Chapter 1 gives a brief overview about the aim for developing this project. The problem definition tells us about the expected outcome of the project for the application.

Chapter 2 shows the Problem Statement.

Chapter 3 contains Design & Implementation i.e. Code of the Project and its Output.

Chapter 4 concludes the Project Report.

Lastly, it has list of References.

**CHAPTER - 2**

**Problem Statement**

MCQ QUIZ is an application developed to conduct an Online Quiz based on time constraints. MCQ Quiz Application is accessed by entering the username and password which is added to the database. Before start of the Quiz, the levels are displayed. After that number of questions to be answered and scoring methods are displayed . Quiz is started by displaying five questions with four options each based General Knowledge, Verbal Reasoning. If the answer is correct, score is incremented by one and no negative marks for wrong answers. If the time exceeds 10 minutes or all the questions are answered the quiz is stopped .Final score will be displayed and updated in the database with usemame.

**CHAPTER - 3**

**Design and Implementation**

**SOURCE CODE:**

import tkinter as tk

from tkinter import \*

import random

import sqlite3

import time

def loginPage(logdata):

sup.destroy()

global login

login = Tk()

user\_name = StringVar()

password = StringVar()

login\_canvas = Canvas(login,width=720,height=440,bg="blue")

login\_canvas.pack()

login\_frame = Frame(login\_canvas,bg="white")

login\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

heading = Label(login\_frame,text="Quiz App Login",fg="black",bg="white")

heading.config(font=('calibri 40'))

heading.place(relx=0.2,rely=0.1)

#USER NAME

ulabel = Label(login\_frame,text="Username",fg='black',bg='white')

ulabel.place(relx=0.21,rely=0.4)

uname = Entry(login\_frame,bg='#d3d3d3',fg='black',textvariable = user\_name)

uname.config(width=42)

uname.place(relx=0.31,rely=0.4)

#PASSWORD

plabel = Label(login\_frame,text="Password",fg='black',bg='white')

plabel.place(relx=0.215,rely=0.5)

pas = Entry(login\_frame,bg='#d3d3d3',fg='black',show="\*",textvariable = password)

pas.config(width=42)

pas.place(relx=0.31,rely=0.5)

def check():

for a,b,c,d in logdata:

if b == uname.get() and c == pas.get():

menu()

break

else:

error = Label(login\_frame,text="Wrong Username or Password!",fg='black',bg='white')

error.place(relx=0.37,rely=0.7)

#LOGIN BUTTON

log = Button(login\_frame,text='Login',padx=5,pady=5,width=5,command=check)

log.configure(width = 15,height=1, activebackground = "#33B5E5", relief = FLAT)

log.place(relx=0.4,rely=0.6)

login.mainloop()

def signUpPage():

root.destroy()

global sup

sup = Tk()

fname = StringVar()

uname = StringVar()

passW = StringVar()

country = StringVar()

sup\_canvas = Canvas(sup,width=720,height=440,bg="blue")

sup\_canvas.pack()

sup\_frame = Frame(sup\_canvas,bg="white")

sup\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

heading = Label(sup\_frame,text="Quiz App SignUp",fg="black",bg="white")

heading.config(font=('calibri 40'))

heading.place(relx=0.2,rely=0.1)

#full name

flabel = Label(sup\_frame,text="Full Name",fg='black',bg='white')

flabel.place(relx=0.21,rely=0.4)

fname = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = fname)

fname.config(width=42)

fname.place(relx=0.31,rely=0.4)

#username

ulabel = Label(sup\_frame,text="Username",fg='black',bg='white')

ulabel.place(relx=0.21,rely=0.5)

user = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = uname)

user.config(width=42)

user.place(relx=0.31,rely=0.5)

#password

plabel = Label(sup\_frame,text="Password",fg='black',bg='white')

plabel.place(relx=0.215,rely=0.6)

pas = Entry(sup\_frame,bg='#d3d3d3',fg='black',show="\*",textvariable = passW)

pas.config(width=42)

pas.place(relx=0.31,rely=0.6)

#country

clabel = Label(sup\_frame,text="Country",fg='black',bg='white')

clabel.place(relx=0.215,rely=0.7)

c = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = country)

c.config(width=42)

c.place(relx=0.31,rely=0.7)

def addUserToDataBase():

fullname = fname.get()

username = user.get()

password = pas.get()

country = c.get()

conn = sqlite3.connect('quiz.db')

create = conn.cursor()

create.execute('CREATE TABLE IF NOT EXISTS userSignUp(FULLNAME text, USERNAME text,PASSWORD text,COUNTRY text)')

create.execute("INSERT INTO userSignUp VALUES (?,?,?,?)",(fullname,username,password,country))

conn.commit()

create.execute('SELECT \* FROM userSignUp')

z=create.fetchall()

print(z)

# L2.config(text="Username is "+z[0][0]+"\nPassword is "+z[-1][1])

conn.close()

loginPage(z)

def gotoLogin():

conn = sqlite3.connect('quiz.db')

create = conn.cursor()

conn.commit()

create.execute('SELECT \* FROM userSignUp')

z=create.fetchall()

loginPage(z)

#signup BUTTON

sp = Button(sup\_frame,text='SignUp',padx=5,pady=5,width=5,command = addUserToDataBase,bg='green')

sp.configure(width = 15,height=1, activebackground = "#33B5E5", relief = FLAT)

sp.place(relx=0.4,rely=0.8)

log = Button(sup\_frame,text='Already have a Account?',padx=5,pady=5,width=5,command = gotoLogin,bg="white",fg='blue')

log.configure(width = 16,height=1, activebackground = "#33B5E5", relief = FLAT)

log.place(relx=0.4,rely=0.9)

sup.mainloop()

def menu():

login.destroy()

global menu

menu = Tk()

menu\_canvas = Canvas(menu,width=720,height=440,bg="blue")

menu\_canvas.pack()

menu\_frame = Frame(menu\_canvas,bg="white")

menu\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

wel = Label(menu\_canvas,text=' W E L C O M E T O Q U I Z S T A T I O N ',fg="white",bg="#101357")

wel.config(font=('Broadway 22'))

wel.place(relx=0.1,rely=0.02)

level = Label(menu\_frame,text='Select your Difficulty Level !!',bg="white",font="calibri 18")

level.place(relx=0.25,rely=0.3)

var = IntVar()

easyR = Radiobutton(menu\_frame,text='Easy',bg="white",font="calibri 16",value=1,variable = var)

easyR.place(relx=0.25,rely=0.4)

mediumR = Radiobutton(menu\_frame,text='Medium',bg="white",font="calibri 16",value=2,variable = var)

mediumR.place(relx=0.25,rely=0.5)

hardR = Radiobutton(menu\_frame,text='Hard',bg="white",font="calibri 16",value=3,variable = var)

hardR.place(relx=0.25,rely=0.6)

def navigate():

x = var.get()

print(x)

if x == 1:

menu.destroy()

easy()

elif x == 2:

menu.destroy()

medium()

elif x == 3:

menu.destroy()

difficult()

else:

pass

letsgo = Button(menu\_frame,text="Let's Go",bg="white",font="calibri 12",command=navigate)

letsgo.place(relx=0.25,rely=0.8)

menu.mainloop()

def easy():

global e

e = Tk()

easy\_canvas = Canvas(e,width=720,height=440,bg="#101357")

easy\_canvas.pack()

easy\_frame = Frame(easy\_canvas,bg="white")

easy\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(10, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

easy\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

easyQ = [

[

"Which of the following is the capital of Arunachal Pradesh?" ,

"Itanagar",

"Dispur",

"Imphal",

"Panaji"

] ,

[

"Which one among the following radiation carries maximum energy?" ,

"Ultraviolet rays",

"Gamma rays",

"X-rays",

"Infra-red rays"

],

[

"The metal whose salts are sensitive to light is \_\_\_\_" ,

"Silver",

"Zinc",

"Copper",

"Gold"

],

[

"Who is the father of geometry?" ,

"Aristotle",

"Euclid",

"Pythagoras",

"Kepler"

],

[

"The World's Largest desert is \_\_\_\_?" ,

"Thar",

"Kalahari",

"Sahara",

"Sonoran"

]

]

answer = [

"Itanagar",

"Gamma rays",

"Silver",

"Euclid",

"Sahara"

]

li = ['',0,1,2,3,4]

x = random.choice(li[1:])

ques = Label(easy\_frame,text =easyQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(easy\_frame,text=easyQ[x][1],font="calibri 10",value=easyQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(easy\_frame,text=easyQ[x][2],font="calibri 10",value=easyQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(easy\_frame,text=easyQ[x][3],font="calibri 10",value=easyQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(easy\_frame,text=easyQ[x][4],font="calibri 10",value=easyQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(e)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

if len(li) == 1:

e.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =easyQ[x][0])

a.configure(text=easyQ[x][1],value=easyQ[x][1])

b.configure(text=easyQ[x][2],value=easyQ[x][2])

c.configure(text=easyQ[x][3],value=easyQ[x][3])

d.configure(text=easyQ[x][4],value=easyQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(easy\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(easy\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

e.mainloop()

def medium():

global m

m = Tk()

med\_canvas = Canvas(m,width=720,height=440,bg="#101357")

med\_canvas.pack()

med\_frame = Frame(med\_canvas,bg="white")

med\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(10, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

med\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

mediumQ = [

[

"Which soil is suitable for agriculture?",

"Red soil",

"Sand",

"Black soil",

"Peaty soil"

],

[

"The device used for measuring altitudes is?",

"Altimeter",

"Ammeter",

"Audiometer",

"Galvanometer"

],

[

"The first chairman of the Atomic Energy Commission was \_\_\_\_\_\_\_\_\_?",

"Dr.C.V.Raman",

"Dr.H.J.Bhabha",

"Dr.A.P.J.Abdul Kalam",

"Dr.Vikram Sarabhai"

],

[

"D.D.T was invented by \_\_\_\_",

"Mosley",

"Rudeolf",

"Karl Benz",

"Dalton"

],

[

"Fathometer is used to measure \_\_\_\_\_",

"Earthquakes",

"Rainfall",

"Ocean depth",

"Sound intensity"

],

]

answer = [

"Peaty soil",

"Altimeter",

"Dr.H.J.Bhabha",

"Mosley",

"Ocean depth"

]

li = ['',0,1,2,3,4]

x = random.choice(li[1:])

ques = Label(med\_frame,text =mediumQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(med\_frame,text=mediumQ[x][1],font="calibri 10",value=mediumQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(med\_frame,text=mediumQ[x][2],font="calibri 10",value=mediumQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(med\_frame,text=mediumQ[x][3],font="calibri 10",value=mediumQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(med\_frame,text=mediumQ[x][4],font="calibri 10",value=mediumQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(m)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

if len(li) == 1:

m.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =mediumQ[x][0])

a.configure(text=mediumQ[x][1],value=mediumQ[x][1])

b.configure(text=mediumQ[x][2],value=mediumQ[x][2])

c.configure(text=mediumQ[x][3],value=mediumQ[x][3])

d.configure(text=mediumQ[x][4],value=mediumQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(med\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(med\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

m.mainloop()

def difficult():

global h

h = Tk()

hard\_canvas = Canvas(h,width=720,height=440,bg="#101357")

hard\_canvas.pack()

hard\_frame = Frame(hard\_canvas,bg="white")

hard\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(10, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

hard\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

hardQ = [

[

"Name the currency used in Japan.",

"Taka",

"Dinar",

"Ngultrum",

"Yen"

],

[

"Which colors must be mixed together to make green?",

"Orange & Blue",

"Red & Blue",

"Blue & Yellow",

"Black & Yellow"

],

[

"How many dots are on one six-sided die",

"17",

"19",

"20",

"21"

],

[

"What is the day after Christmas commonly known as?",

"Caroling Day",

"Boxing Day",

"Prayer Day",

"Shopping Day"

],

[

"Who wrote Hamlet?",

"Walt Whitman",

"Leonardo da Vinci",

"Franz Kafka",

"William Shakespeare"

]

]

answer = [

"Yen",

"Blue & Yellow",

"21",

"Boxing Day",

"William Shakespeare"

]

li = ['',0,1,2,3,4]

x = random.choice(li[1:])

ques = Label(hard\_frame,text =hardQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(hard\_frame,text=hardQ[x][1],font="calibri 10",value=hardQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(hard\_frame,text=hardQ[x][2],font="calibri 10",value=hardQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(hard\_frame,text=hardQ[x][3],font="calibri 10",value=hardQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(hard\_frame,text=hardQ[x][4],font="calibri 10",value=hardQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(h)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

if len(li) == 1:

h.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =hardQ[x][0])

a.configure(text=hardQ[x][1],value=hardQ[x][1])

b.configure(text=hardQ[x][2],value=hardQ[x][2])

c.configure(text=hardQ[x][3],value=hardQ[x][3])

d.configure(text=hardQ[x][4],value=hardQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(hard\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(hard\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

h.mainloop()

def showMark(mark):

global sh

sh = Tk()

show\_canvas = Canvas(sh,width=720,height=440,bg="#101357")

show\_canvas.pack()

show\_frame = Frame(show\_canvas,bg="white")

show\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

st = "Your score is "+str(mark)

mlabel = Label(show\_canvas,text=st,fg="black")

mlabel.place(relx=0.5,rely=0.2,anchor=CENTER)

sh.mainloop()

def start():

global root

root = Tk()

canvas = Canvas(root,width = 720,height = 440)

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

img = PhotoImage(file=r"C:\Users\Siddharth\Downloads\SIMPLE\_QUIZ\_APP\_IN\_PYTHON\_WITH\_SOURCE\_CODE (1)\Quiz-App\images\back.png")

canvas.create\_image(50,10,image=img,anchor=NW)

button = Button(root, text='Start',command = signUpPage)

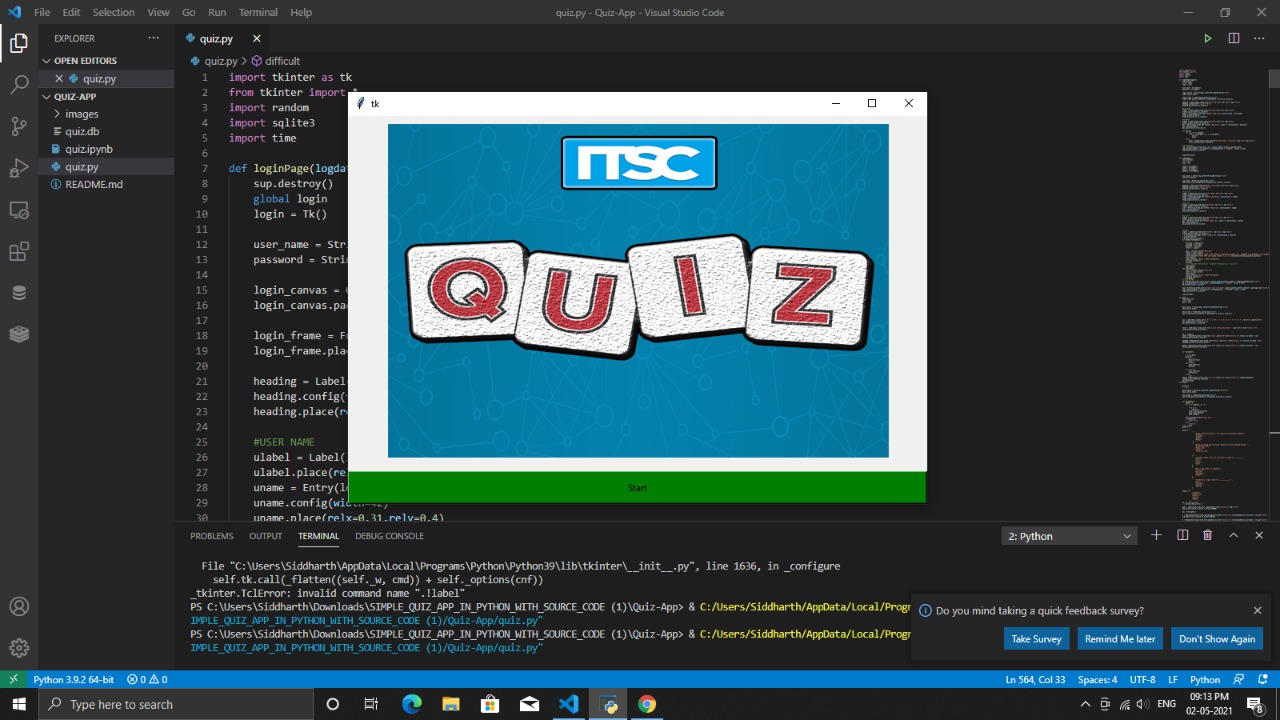
button.configure(width = 102,height=2, activebackground = "#33B5E5", bg ='green', relief = RAISED)

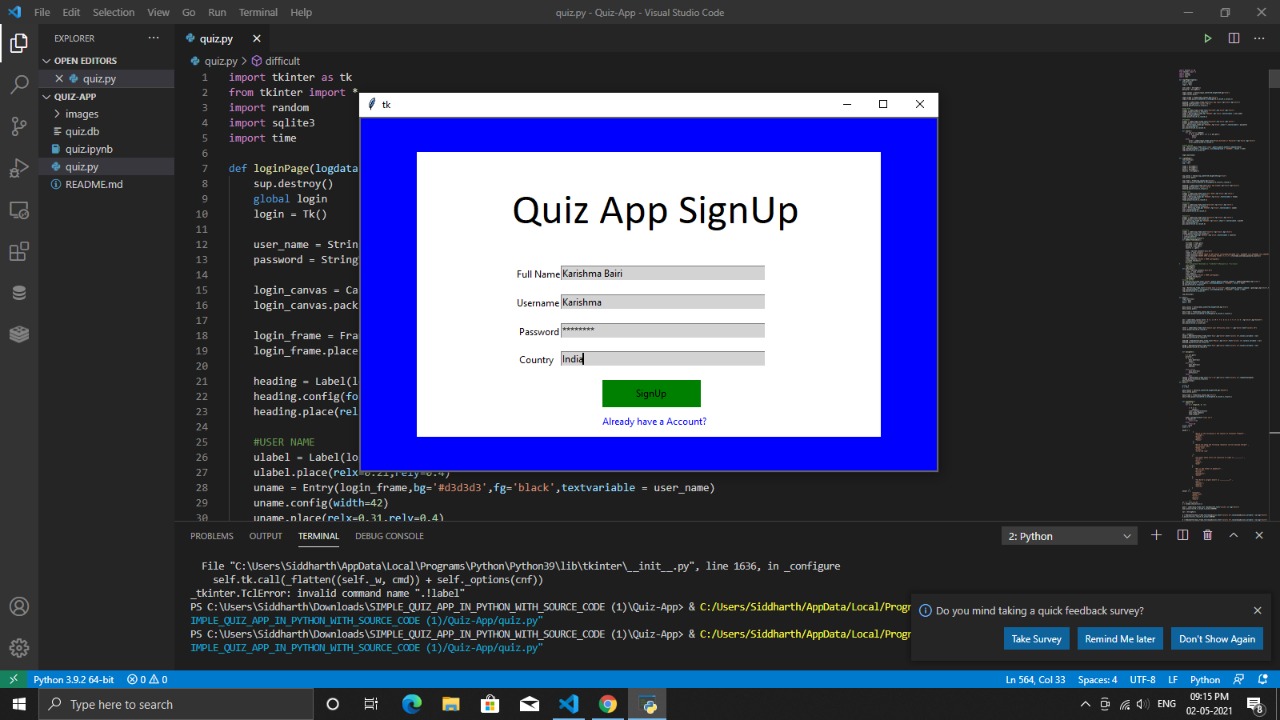
button.grid(column = 0 , row = 2)

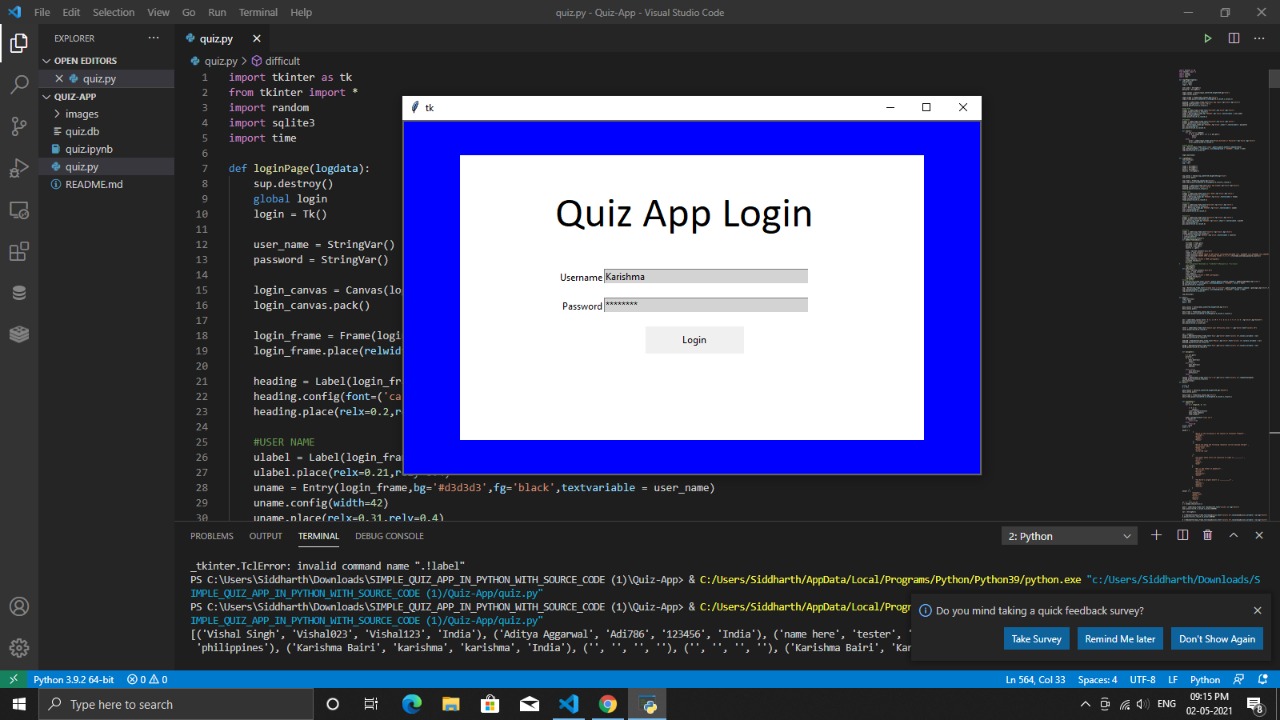
root.mainloop()

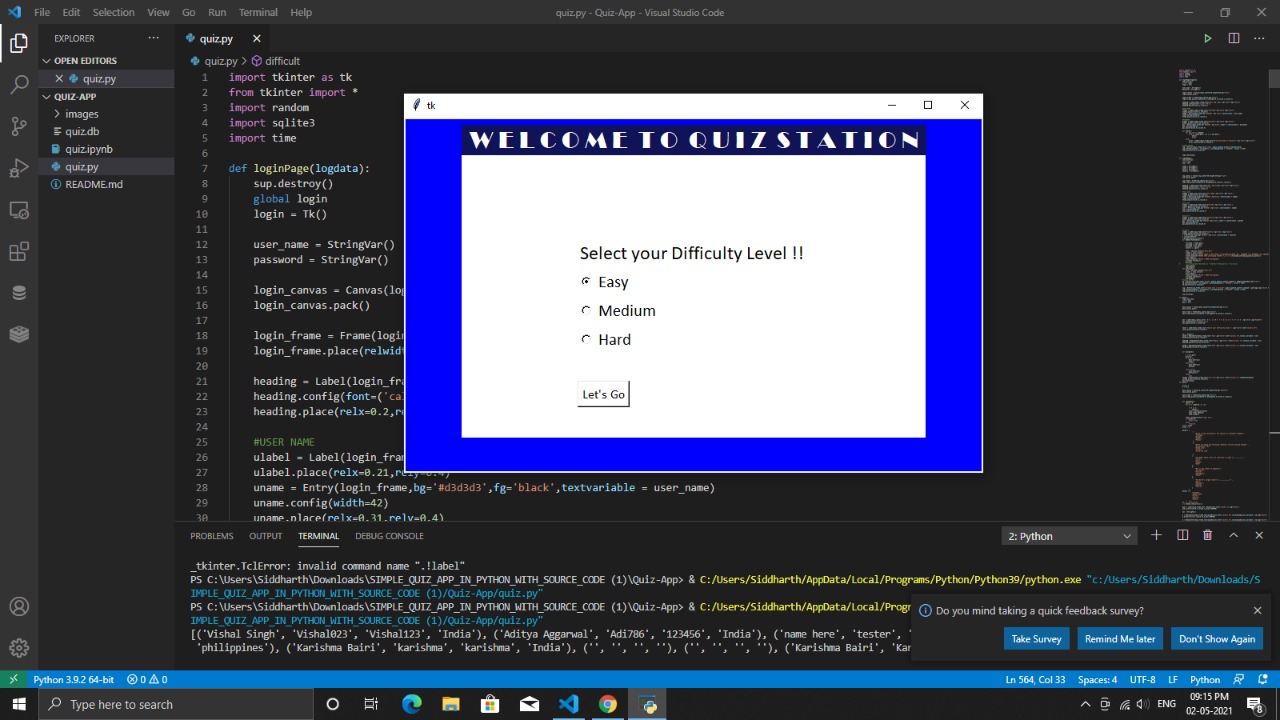
if \_name=='main\_':

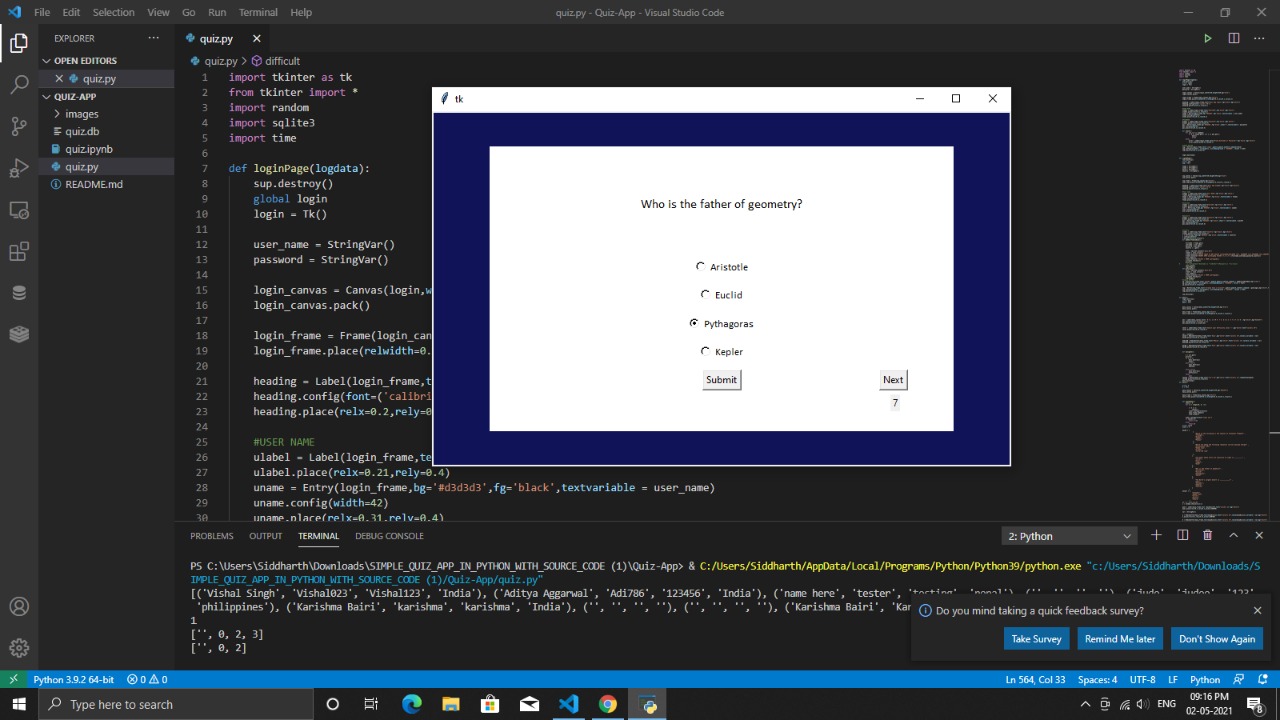
start()

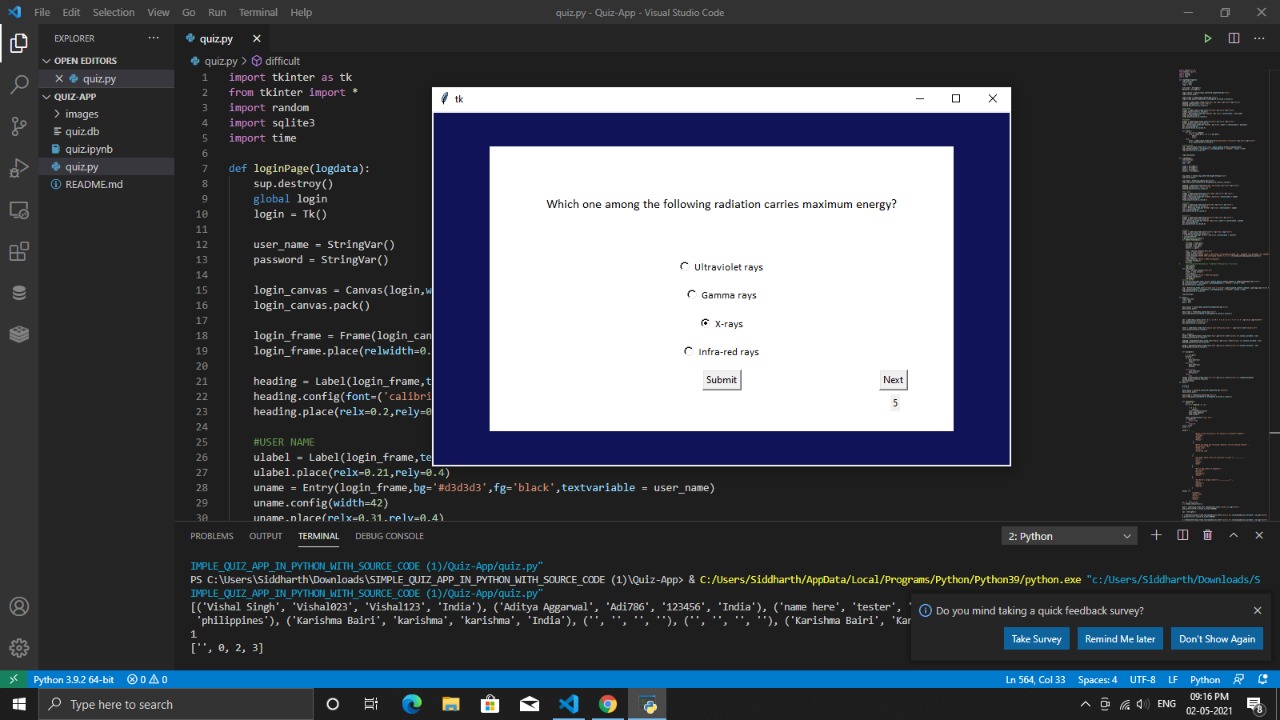
**OUTPUT:**

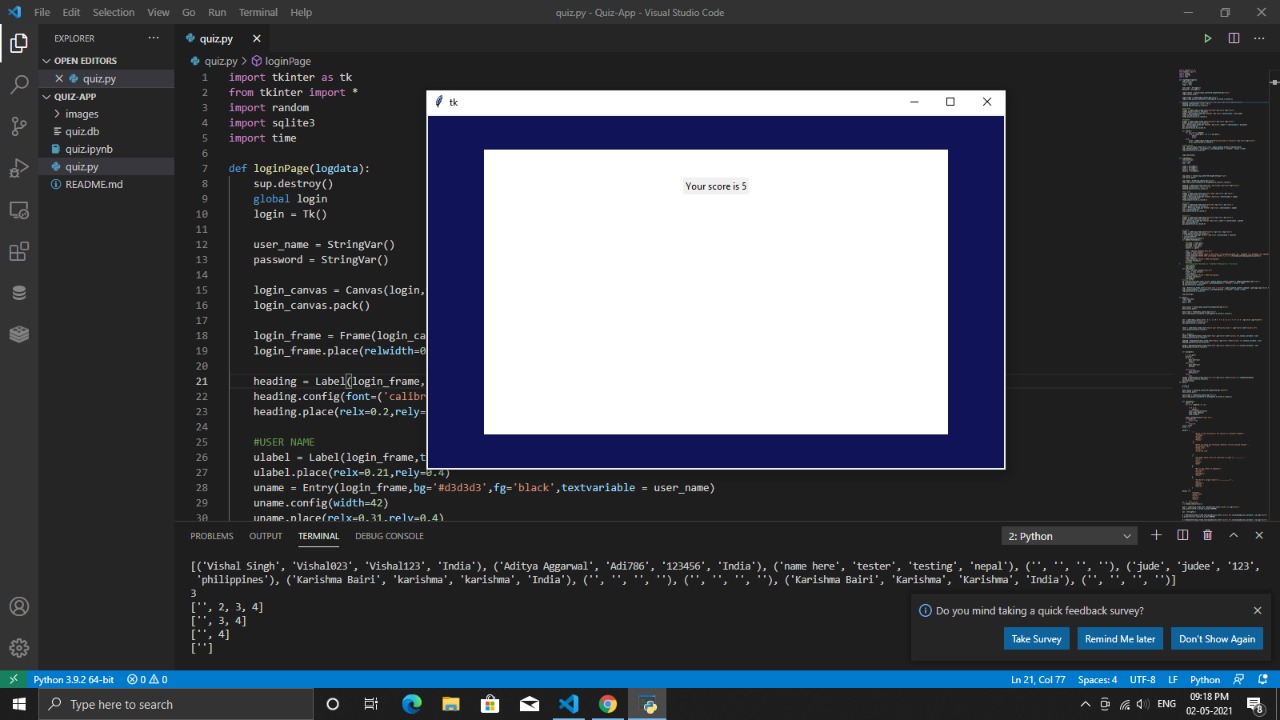












**CHAPTER - 4**

**Conclusion**

In this MCQ Quiz Application we have focused on an automated system, which replaces the manual system. But the interesting thing is that the thesis is not just the MCQ Quiz application; it has its own intelligent capability. This was the the actual target feature for our thesis. Basically, we have tried to introduce the evaluation method of the student - performance in this application. Here, the questions are appeared according to the levels selected by the student. We have implemented this in our thesis and it works successfully in our system. In this system, an educational institute can make their examination procedure automated. This system is basically designed for educational institute but can also be developed for other examination systems like job interview, quiz contest and for other criterias.

**References**

1. <https://partheniumprojects.com/online-quiz-management-system/>
2. <https://www.scribd.com/document/334841504/Synopsis-of-MCQ-Quiz-Application>
3. <https://www.researchgate.net/publication/303896969_Quizzy_Quiz_Application_development_using_Android_Plartform>