****

**SYNOPSIS: *QUIZ MODEL BASED ON CATAGORIES WITH IMPROVEMENT SUGGESTIONS***

***GROUP 14 –***

**YouTube link =** <https://youtu.be/xqdlkUyc9FE>

**BY Vishnu chopra 11805967 K18XC 51**

**BY Tarun Kumar Sharma 11805996 K18XC 52**

**BY siddharth Singh 11803236 K18XC 50**

**BY Rajat Kumar 11813774 K18XC 53**

**INTRODUCTION:**

**We have designed an online quiz Application using PYTHON language and Graphical User Interface (GUI) (both backend and frontend approach is used). The purpose of the project is to build an application program to reduce the manual work for managing the result, student’s marks, managing the information of examination.**

**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 the other activities rather to concentrate on the record keeping.**

**NOTE- we are** going to use **Python 3.6** version.

## 1. What Is a Graphical User Interface (GUI)

**GUI** is a desktop app which helps you to interact with the computers. They are used to perform different tasks in the desktops, laptops, other electronic devices, etc.., here, we mainly talking about the laptops and desktops.

## 2. What Is Tkinter

**Tkinter** is an inbuilt **Python** module used to create simple **GUI** apps. It is the most commonly used module for **GUI** apps in the **Python**.

* **Button**: - **Button** widget is used to place the buttons in the **tkinter**.
* **Canvas**: - **Canvas** is used to draw shapes in your **GUI**.
* **Check button**: - **Check button** is used to create the check buttons in your application. You can select more than one option at a time.
* **Radio button**- It is used to select one option in your application out of other options.

**Objectives**

1. The main objective of “Quiz” is to facilitate a user friendly environment for all users and reduces the manual effort.
2. In past days quiz is conducted manually but in further resolution of the technology we are able to generate the score and pose the queries automatically.
3. The functional requirements include creating users that are going to participate in the quiz, automatic score and report generation and administrative tasks like add, delete, update for admin privilege users.
4. In this application, all the permissions lies with the **administrator** i.e., specifying the details of the quiz with checking result will show to interviewee or not, addition of question and answers, marks for each question, Set timer for each quiz and generate report with score for each quiz
5. We use **Radio Button:**To select only one option by displaying a number of options to a user as toggle buttons.
6. We use **Submit Button:**To submit the test which we performed.
7. We can use ***Exit Button*** to quit the interface.(destroy the window)
8. We can add images and in the **GUI**.

**METHODOLOGIES:-**

We will use any python programming platform for coding and building the project.

1. For this we need some python **standard libraries** like **tkinter .GUI** is a desktop app which helps you to interact with the computers.
2. We have used “import random”

* **Label** is used to insert some objects into the **window**. Here, we are adding a **Label** with some text.
* **pack()** attribute of the widget is used to display the **widget** in a size it requires. Pack() method includes:

1. Pady
2. padx
3. anchor
4. side etc.

* Finally, the **mainloop()** method to display the **window** until you manually close it.
* **Scale:**It is used to provide a graphical slider that allows to select any value from that scale.
* **Destroy-** We have used this method to destroy the objects (or the widgets) and the main Window to end the quiz

**Backend work:**

1) A **List** of 10 questions (index from 0 to 9) in a variable named as **question**

A list of 10 multiple choices (index from 0 to 9) in a variable names as **answer\_choice**.

A list of 10 answers ( index from 0 to 9) ->

answer\_keys=[3,3,0,3,2,2,1,0,0,2] respectively .

2) The following code shows that to get the random question every time from the list of questions we need to use **randint()** method.

indexes=[]

**def** generate():

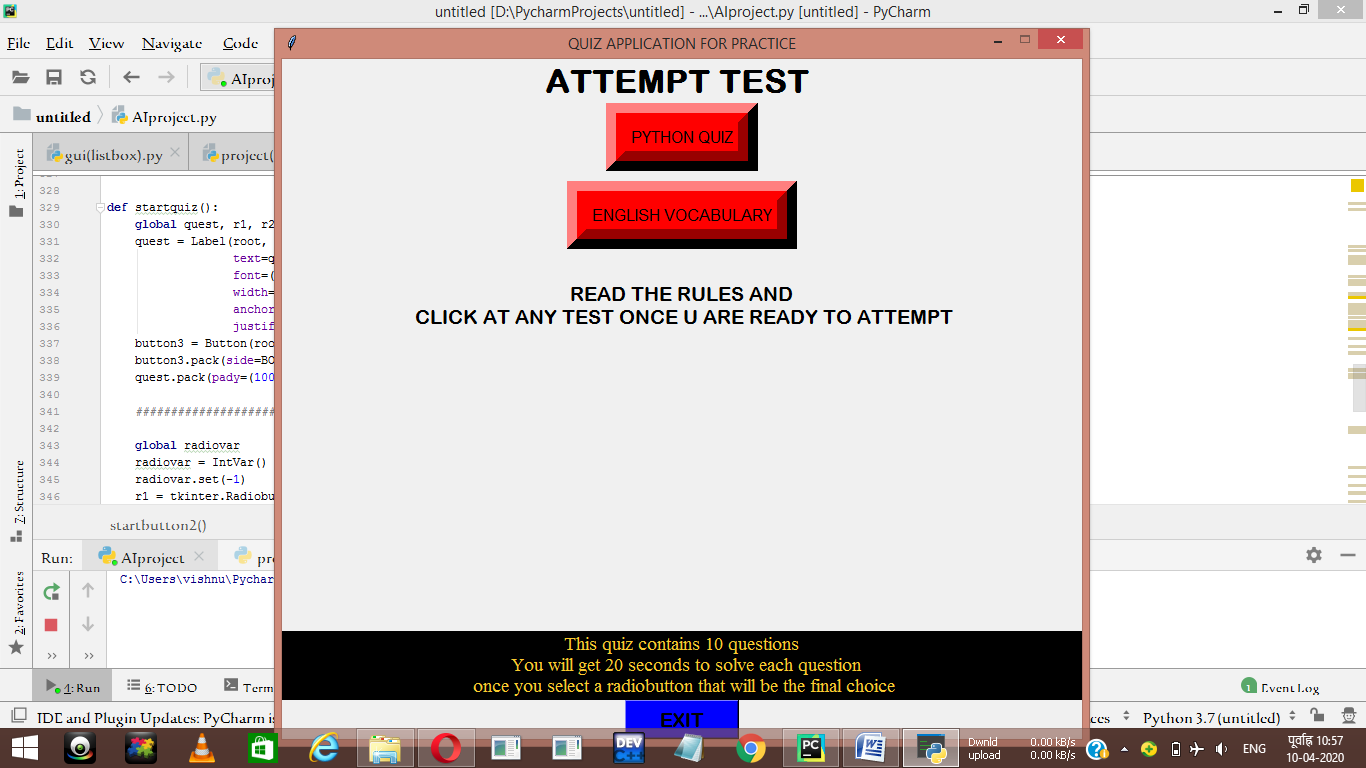
**global** indexes*#* **while**(len(indexes)<5):*# because we need to select 5 question* x=random.randint(0,9)  
 indexes.append(x)  
 **if** x **in** indexes:  
 **continue** *### continue the loop* **else**:  
 indexes.append(x)

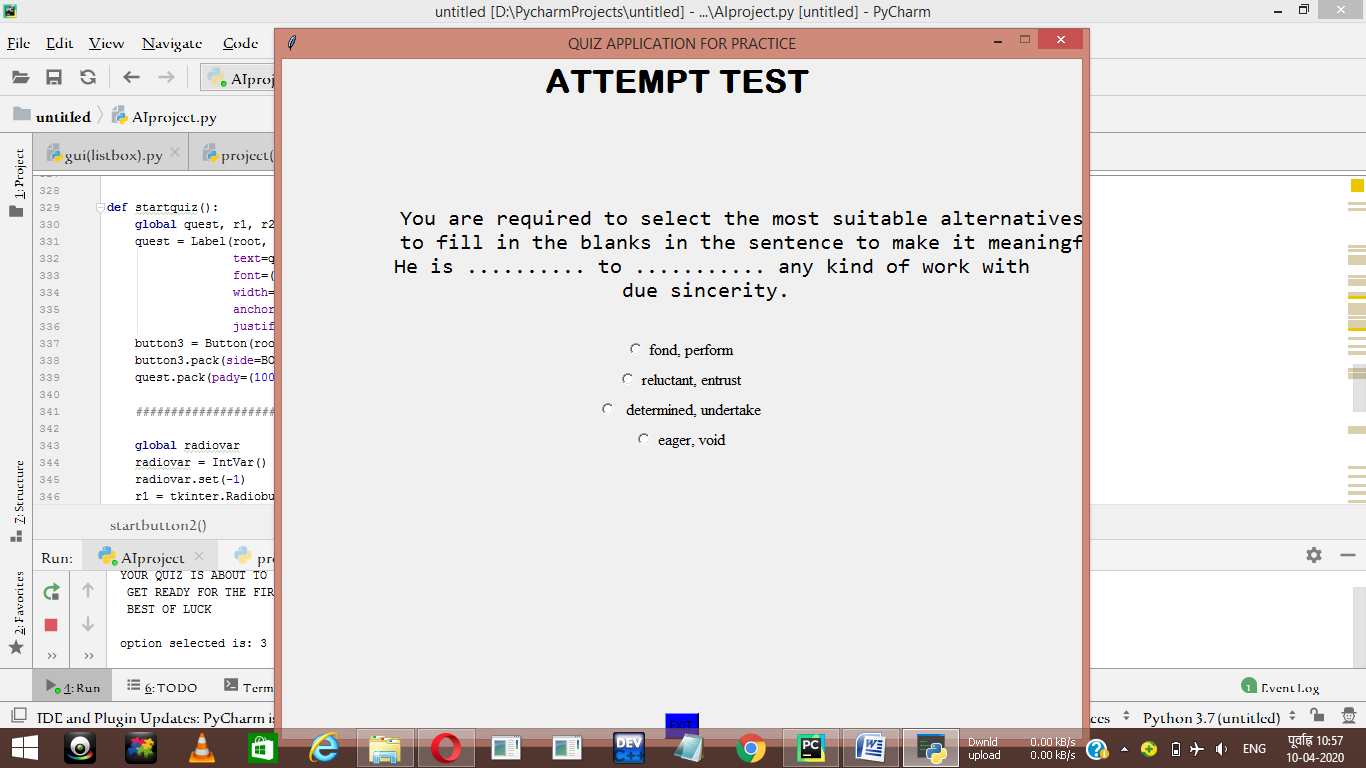
3) To show the question randomly we used the above method and now in the Label **text** will contain the **question[indexes[0]]**

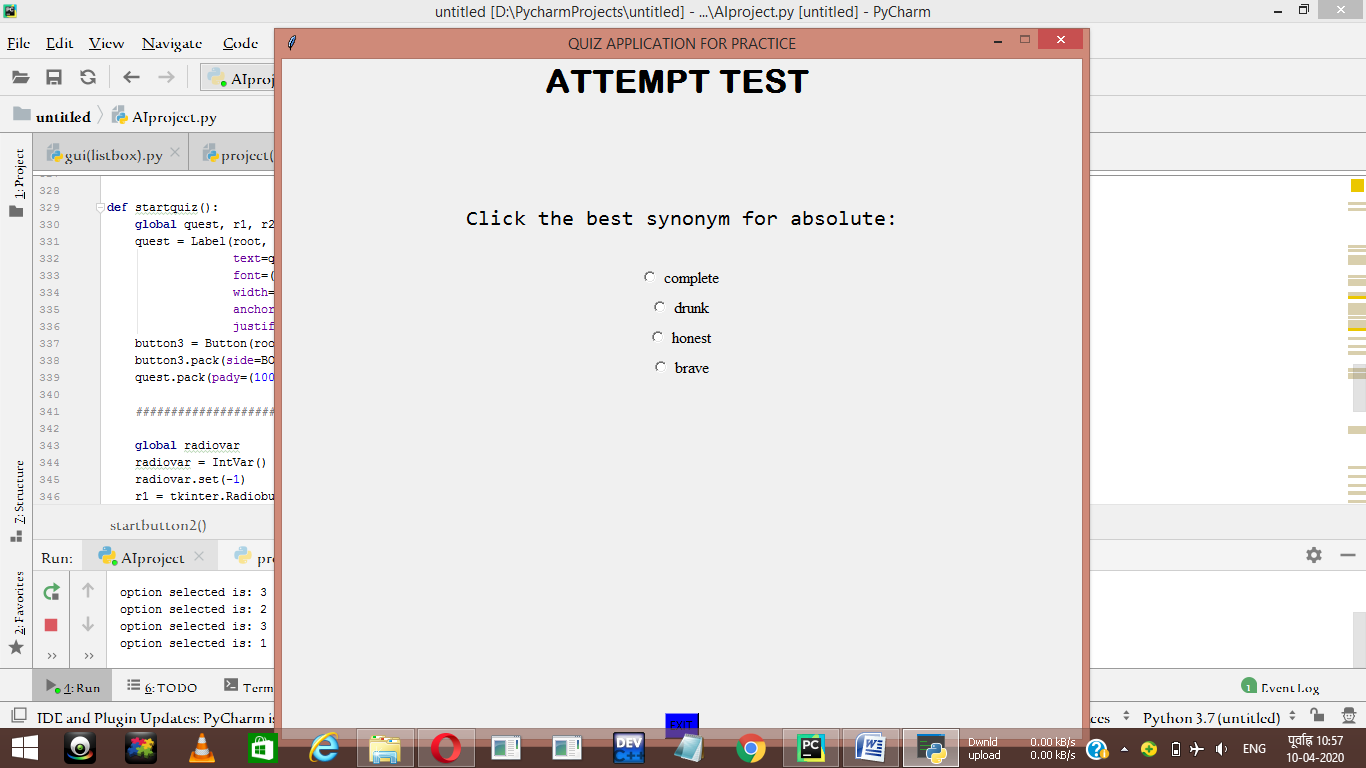
**def** startquiz():  
 **global** quest,r1,r2,r3,r4  
 quest=Label(root,  
 text=question[indexes[0]],  
 font=(**'Consolas'**,16),  
 width=500,  
 anchor=**"center"**,  
 justify=**"center"**)  
 quest.pack(pady=(100,30))

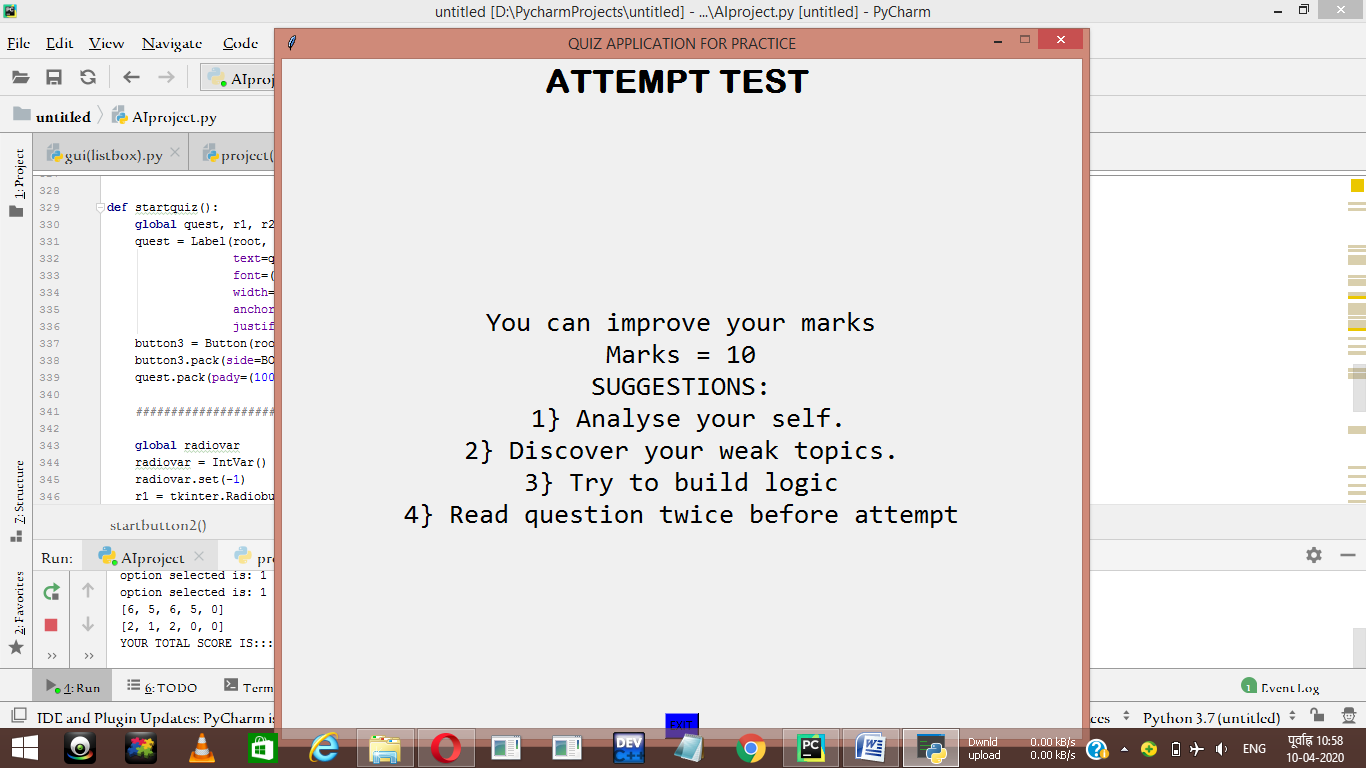
4) Same we will do in the **radio button**, example:(see the highlighted part)

radiovar=IntVar()  
radiovar.set(-1)  
r1=tkinter.Radiobutton(  
 root,  
 text=answer\_choice[indexes[0]][0],  
 padx=50,  
 font=(**"Times"**,12),  
 value=0,  
 variable=radiovar,  
 command=selected,  
  
)  
r1.pack(anchor=tkinter.CENTER)

****

****

****



**# After you attempt all your questions, you will get the marks.**

**LIMITATIONS:**

**A]** Off-line reports of results, marks, courses cannot be generated due to batch mode execution.

**B**] The transaction are executed in off-line mode, hence on-line data for results, paper capture and modification is not possible.

**C**] Excel export has not been developed for students, examination due to some criticality.

|  |
| --- |
| BIBLOGRAPHY   * 1. Introduction To Programming Using Python,by DANIEL LIANG   2. YOUTUBE   3. Stackoverflow.com |

**CONTRIBUTION:**

VISHNU CHOPRA (11805967) => PYTHON CODING

TARUN KUMAR (11805996) => VIDEO MAKING/RECORDING

RAJAT KUMAR (**11813774) => POWER POINT PRESENTATION**

**SIDDHARTH SINGH (11803236) => Helped in GUI part and Report making**

**YouTube link =** <https://youtu.be/xqdlkUyc9FE>