AIR FORCE SCHOOL AGRA

CBSE PRACTICAL FILE

COMPUTER SCIENCE

SUBMMITED BY: ABHIMANYU SINGH FAUZDAR

CLASS & SEC: XII -A

ROLL NO: 02

SUBMMITED TO: MR. NITIN SHARMA

INDEX

1. CERTIFICATE
2. AKNOWLEDGMENT
3. DECLARATION
4. INTRODUCTION
5. AIM
6. CODING AND O/P SS
7. REQUIREMENTS
8. BIBLIOGRAPHY

**CERTIFICATE OF COMPLETION**

This is to certify that the project work on multiple choice questions quiz has been completed by ABHIMANYU SINGH FAUZDAR of Class-XII Section A of Air Force School Agra . The above mentioned project work has been completed under my(Mr. Nitin Sharma) guidance during the academic year 2024-25 .

**Signature of Teacher :**

**ACKNOWLEDGEMENT**

I would like to extend my sincere and heartfelt obligation towards all those who have helped me in making this project . Without their active guidance , help , cooperation and encouragement , I would not have been able to present the project on time . I am extremely thankful and pay my sincere gratitude to my teacher of COMPUTER SCIENCE **MR. NITIN SHARMA** for his valuable guidance and support for completion of this project . I also acknowledge with a deep sense of reverence , my gratitude towards my parents , other faculty members of the school and friends for their valuable suggestions given to me in completing the project.

**Name : ABHIMANYU SINGH FAUZDAR**

DECLARATION

I am hereby declare that the project work on:-

MULTIPLE CHOICE QUESTION QUIZ

is prepared by:-

ABHIMANYU SINGH FAUZDAR

Under the supervision of MR. NITIN SHARMA (CS teacher)

**INTRODUCTION**

**Introduction to the MCQ Quiz Project**

This project aims to create a robust and interactive Multiple Choice Question (MCQ) quiz application designed using Python and MySQL. The application allows users to engage in quizzes, test their knowledge, and track their performance in various subjects. Python serves as the primary programming language for developing the application’s core logic and user interface, providing seamless interaction and functionality. MySQL is employed as the backend database to store and manage quiz questions, answers, user data, and scores, ensuring efficient data retrieval and persistence.

The MCQ quiz project is designed to be user-friendly and flexible, with features such as randomized question order, multiple quiz categories, and the ability to store user scores for future reference. The integration of Python with MySQL enables the system to handle large sets of questions and provides a scalable solution that can easily be expanded with new features, such as user authentication or real-time leaderboards. This project demonstrates the application of Python for backend development and MySQL for database management, making it an ideal showcase for implementing database-driven applications in a real-world context.

By combining these technologies, this project offers an interactive platform for users to test their knowledge while showcasing key concepts in software development, database management, and full-stack programming.

This project aims to create a dynamic and interactive Multiple Choice Question (MCQ) quiz application, built using Python and MySQL, with an emphasis on seamless connectivity between the frontend and backend. The application allows users to take quizzes, evaluate their performance, and track scores in various subjects. Python is used for implementing the core logic, user interface, and handling interactions, while MySQL serves as the backend database, storing questions, answers, user information, and scores efficiently.

The key feature of this project is the robust **connectivity** between Python and MySQL, allowing the application to seamlessly retrieve and store quiz data. Using Python's MySQL connector, the app establishes a connection to the MySQL database, enabling smooth data transactions such as fetching questions, validating answers, and saving user scores in real time. This connection ensures that the data remains consistent and easily accessible, even when handling a large number of quiz attempts or questions.

By leveraging Python's ability to interact with MySQL, the system can deliver dynamic quizzes by selecting random questions, categorizing them based on subjects, and adapting to user inputs. The application also supports user tracking and persistence, meaning users can have their scores stored and accessed across different sessions.

This project highlights not only the application of Python for logic and UI development but also the integration of databases through MySQL to create a fully connected and scalable system. By understanding the connectivity between the frontend and backend, this project provides a practical example of how modern applications manage data flow and offer an interactive, user-centered experience.

AIM

* Write a program on the topic multiple choice question quiz

using python and MySql - connectivity

CODING

import sys

import random

import mysql.connector

from mysql.connector import Error

import warnings

warnings.filterwarnings("ignore")

def create\_server\_connection(host\_name, user\_name, user\_password):

connection = None

try:

connection = mysql.connector.connect(

host = host\_name,

user = user\_name,

passwd = user\_password

)

print("Your server is running!!!")

except Error as err:

print(f"Error: '{err}'")

return connection

pw = 'abhi'

db = 'quiz'

**# Making connection**

connection = create\_server\_connection("localhost", "root", pw)

**# Database creation**

def create\_database(connection, query):

cursor = connection.cursor()

try:

cursor.execute(query)

print("Database Created Successfully.")

except Error as err:

print(f"Error: '{err}'")

create\_database\_query = "Create database quiz"

create\_database(connection, create\_database\_query)

**# Database connection**

def create\_db\_connection(host\_name, user\_name, user\_password, db\_name):

connection = None

try:

connection = mysql.connector.connect(

host = host\_name,

user = user\_name,

passwd = user\_password,

database = db\_name

)

print("MySQL Database Connection Successful")

except Error as err:

print(f"Error: '{err}'")

return connection

**# Executing sql queries**

def execute\_query(connection, query):

cursor = connection.cursor()

try:

cursor.execute(query)

connection.commit()

print("Query Executed Successful.")

except Error as err:

print(f"Error: '{err}'")

create\_quiz\_table = '''

create table question(

question\_id int primary key,

question varchar(1000) not null,

option\_1 varchar(100) not null,

option\_2 varchar(100) not null,

option\_3 varchar(100) not null,

option\_4 varchar(100) not null,

answer varchar(100) not null

)

'''

connection = create\_db\_connection("localhost", "root", pw, db)

execute\_query(connection, create\_quiz\_table)

create\_user\_table = '''

create table users(

user\_id int primary key,

name varchar(50) not null,

score int not null

)

'''

connection = create\_db\_connection("localhost", "root", pw, db)

execute\_query(connection, create\_user\_table)

**# Question section**

def Question():

key='Y'

while key=='Y' or key=='y':

print("Welcome to Question Portal")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

question=input("Enter the question: ")

option\_1=input("Enter the option 1: ")

option\_2=input("Enter the option 2: ")

option\_3=input("Enter the option 3: ")

option\_4=input("Enter the option 4: ")

answer=0

while answer==0:

option=int(input("Which option is correct answer (1,2,3,4): "))

if option==1:

answer=option\_1

elif option==2:

answer=option\_2

elif option==3:

answer=option\_3

elif option==4:

answer=option\_4

else:

print("Please choose the correct option as answer")

mycursor.execute("Select \* from question")

data=mycursor.fetchall()

question\_id=(mycursor.rowcount)+1

mycursor.execute("Insert into question values (%s,%s,%s,%s,%s,%s,%s)",

(question\_id,question,option\_1,option\_2,option\_3,option\_4,answer))

mydb.commit()

key=input("Question added successfully.. Do you want to add more (Y/N) ")

Home()

**# Quiz section**

def Quiz():

print("Welcome to Quiz portal")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

mycursor.execute("Select \* from question")

data=mycursor.fetchall()

name=input("Enter your name: ")

total\_question=mycursor.rowcount

to\_attempt=int(input(f"Enter the number of questions to attempt (max {total\_question}):"))

question\_ids=[i for i in range(1, total\_question+1)]

question\_ids=random.sample(question\_ids, to\_attempt) #only some ids are required

print("Quiz has started")

c=1

score=0

for i in range(0,len(question\_ids)):

mycursor.execute("Select \* from question where question\_id=%s",(question\_ids[i],))

ques=mycursor.fetchone()

print("--------------------------------------------------------------------------------------------")

print("Q.",c,": ",ques[1],"\nA.",ques[2],"\t\tB.",ques[3],"\nC.",ques[4],"\t\tD.",ques[5])

print("--------------------------------------------------------------------------------------------")

c+=1

ans=None

while ans==None:

choice=input("Answer (A,B,C,D): ")

if choice=='A' or choice=='a':

ans=ques[2]

elif choice=='B' or choice=='b':

ans=ques[3]

elif choice=='C' or choice=='c':

ans=ques[4]

elif choice=='D' or choice=='d':

ans=ques[5]

else:

print("Kindly select A,B,C,D as option only")

if ans==ques[6]:

print("Correct")

score=score+1

else:

print("Incorrect.. Correct answer is: ",ques[6])

print("Quiz has ended !! Your final score is: ",score)

mycursor.execute("Select \* from users")

data=mycursor.fetchall()

user\_id=(mycursor.rowcount)+1

mycursor.execute("Insert into users values (%s,%s,%s)",

(user\_id,name,score))

mydb.commit()

input("Press any key to continue: ")

Home()

**# Creating home page**

def Home():

opt=1

while opt!=3:

print("Welcome to Quiz")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("1. Enter Questions")

print("2. Take Quiz")

print("3. Exit")

opt=int(input("Enter your choice: "))

if opt==1:

Question()

elif opt==2:

Quiz()

elif opt==3:

print("Exiting the Quiz")

mycursor.close()

mydb.close()

sys.exit();

else:

Home()

**# Database connection**

mydb=mysql.connector.connect(

host= "localhost",

user= "root",

passwd="abhi",

database= "quiz"

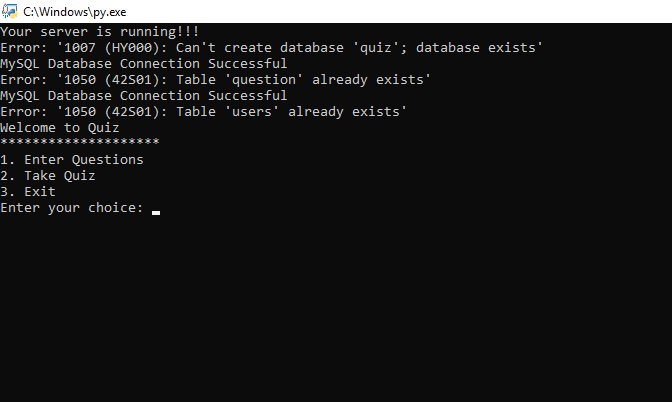
)

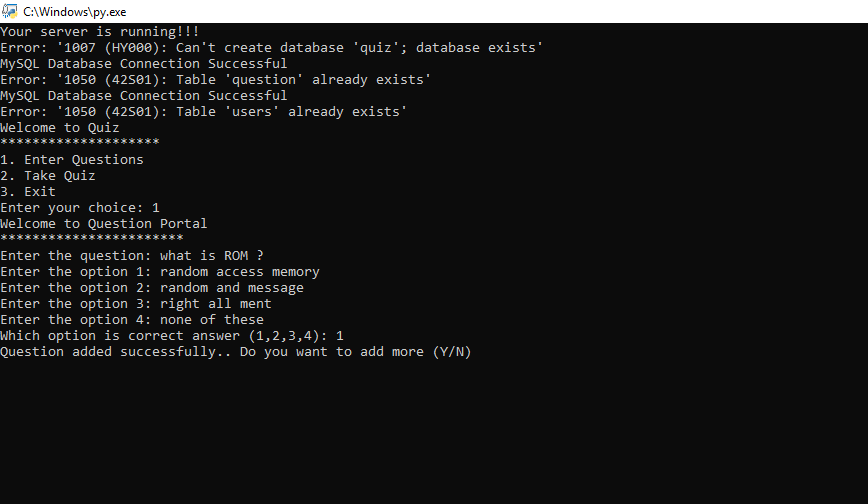
mycursor=mydb.cursor()

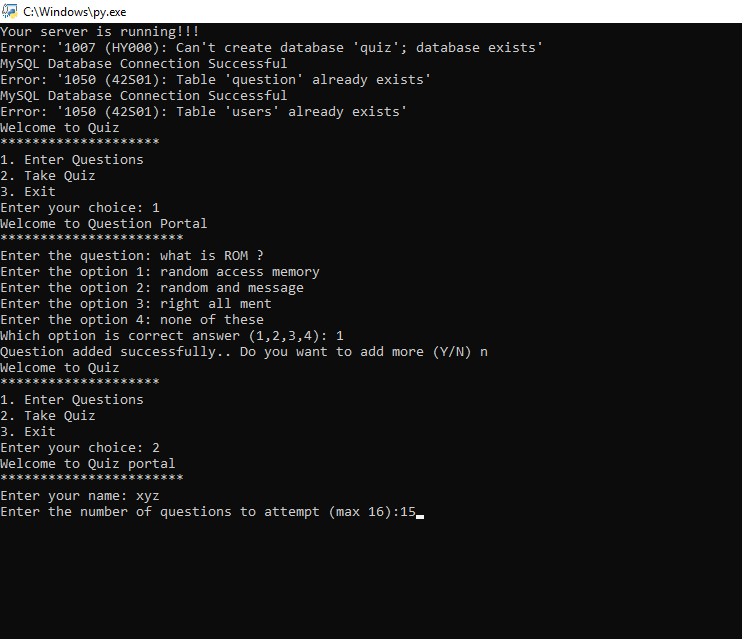
**# Running the program**

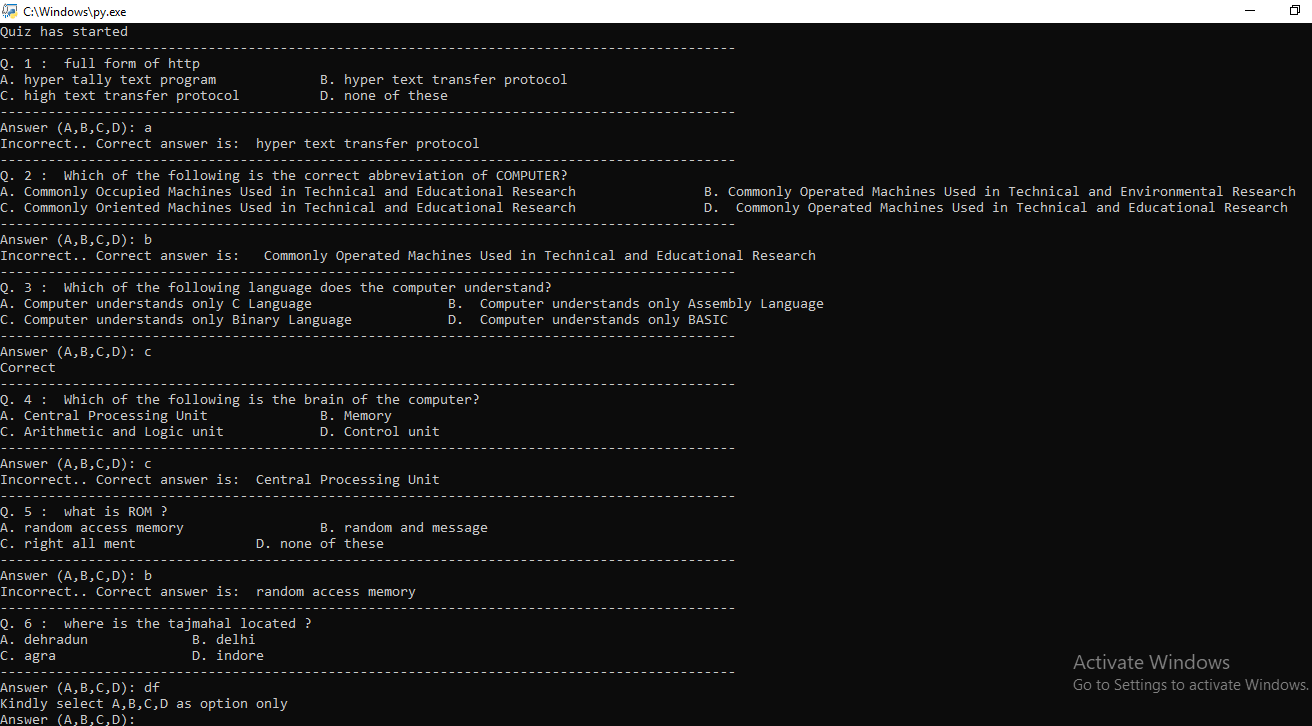
Home()

SCREENSHOTS OF O/P

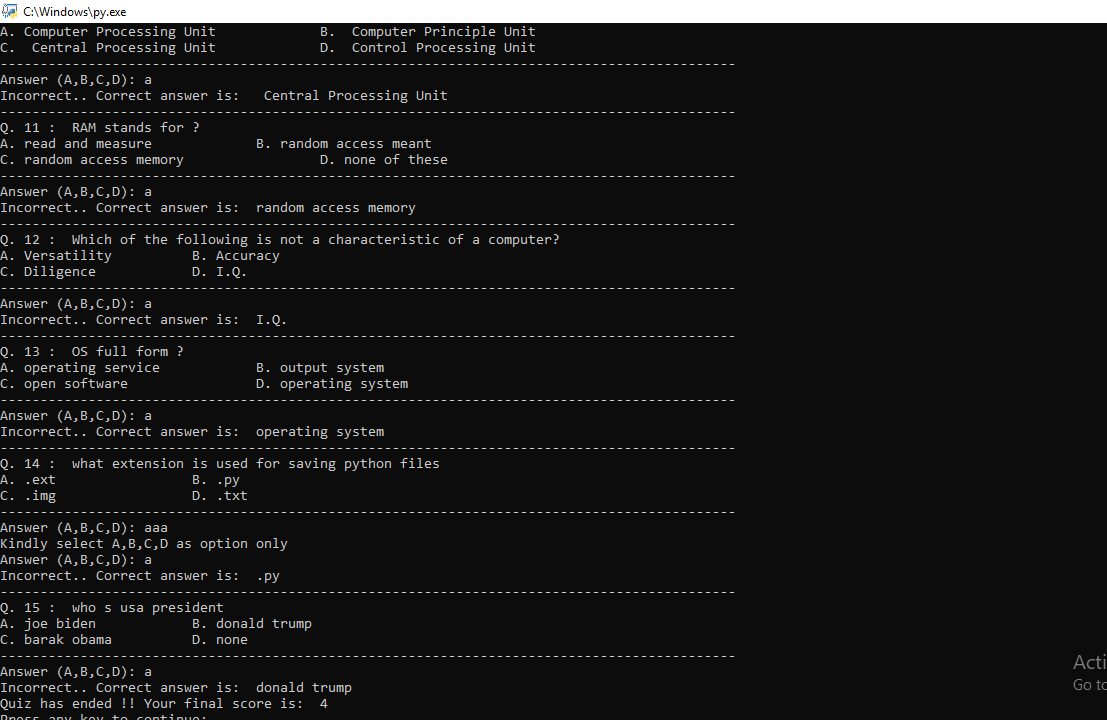
THIS IS THE OPENING INTERFACE OR HOME PAGE…

THIS IS THE SCREENSHOT OF WHEN WE CHOOSE 1st OPTION , THAT IS TO ADD QUESTIONS…

THIS IS WHEN WE CHOOSE OPTION 2ND , THAT IS TO TAKE THE QUIZ QUESTIONS…



THESE ARE THE SCREENSHOTS WHEN QUIZ QUESTIONS START…

AFTER THE SCORE , THE QUIZ WILL END…OR ELSE CHOOSE OPTION 3RD TO EXIT FROM QUIZ…

REQUIREMENTS

* Python(3.13 )
* Python IDLE
* MySQL 8.0
* Python MySQL –connector

BIBLIOGRAPHY

* Sumita Arora’s book of CS CLASS XII
* GitHub

THANK YOU

MADE BY :-

ABHIMANYU SINGH FAUZDAR

erticaRE

CBHDDNHD

1. CERTIFICATE
2. C