KONGU ENGINEERING COLLEGE

(Autonomous)

PERUNDURAI ERODE – 638 060

Quiz App

A Project Report

submitted by

DHARUN VIDYAKAR K S

23ITR035

DHARUN PRAKASH G T

23ITR034

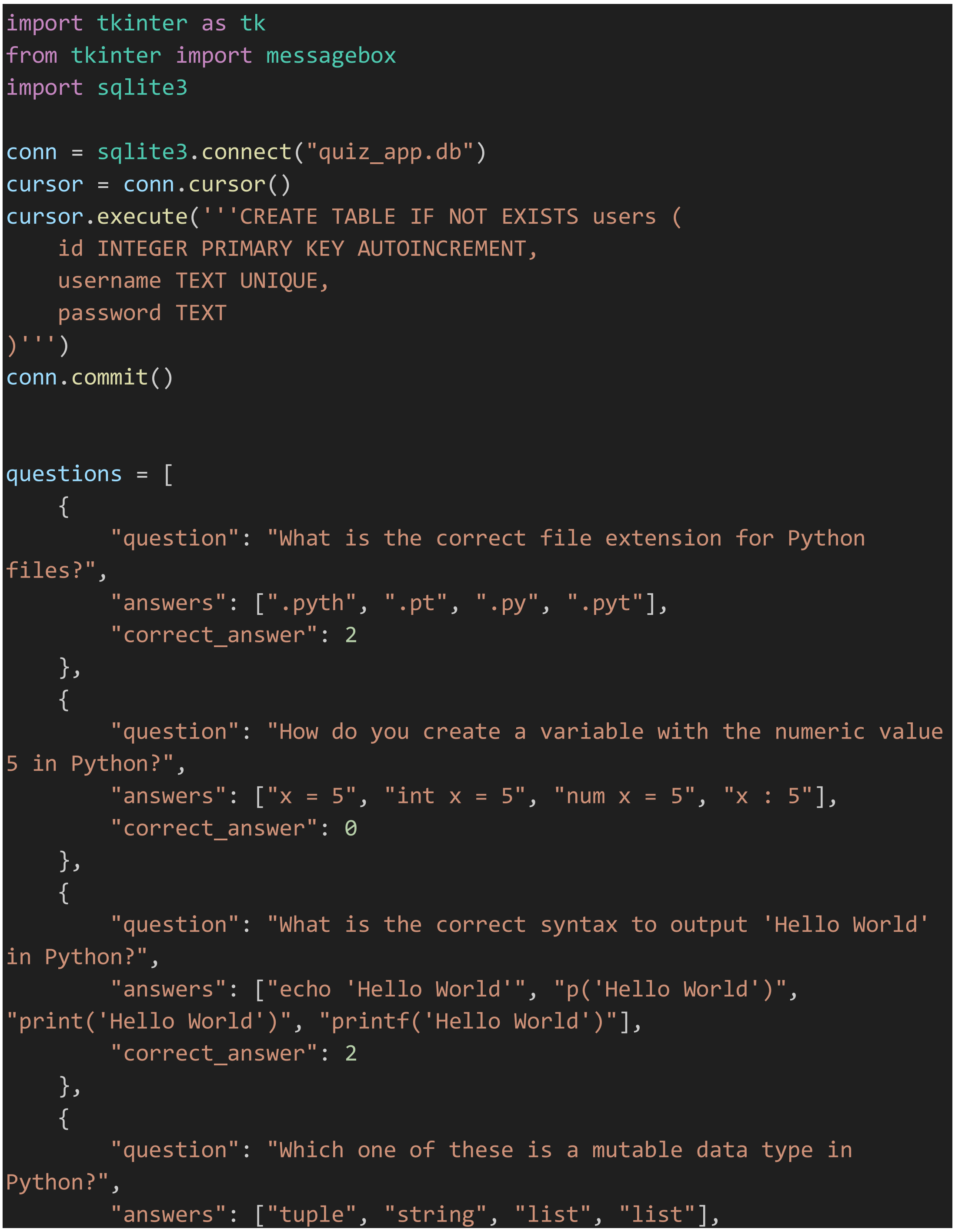
ASWATH SIVA K M

23ITR010

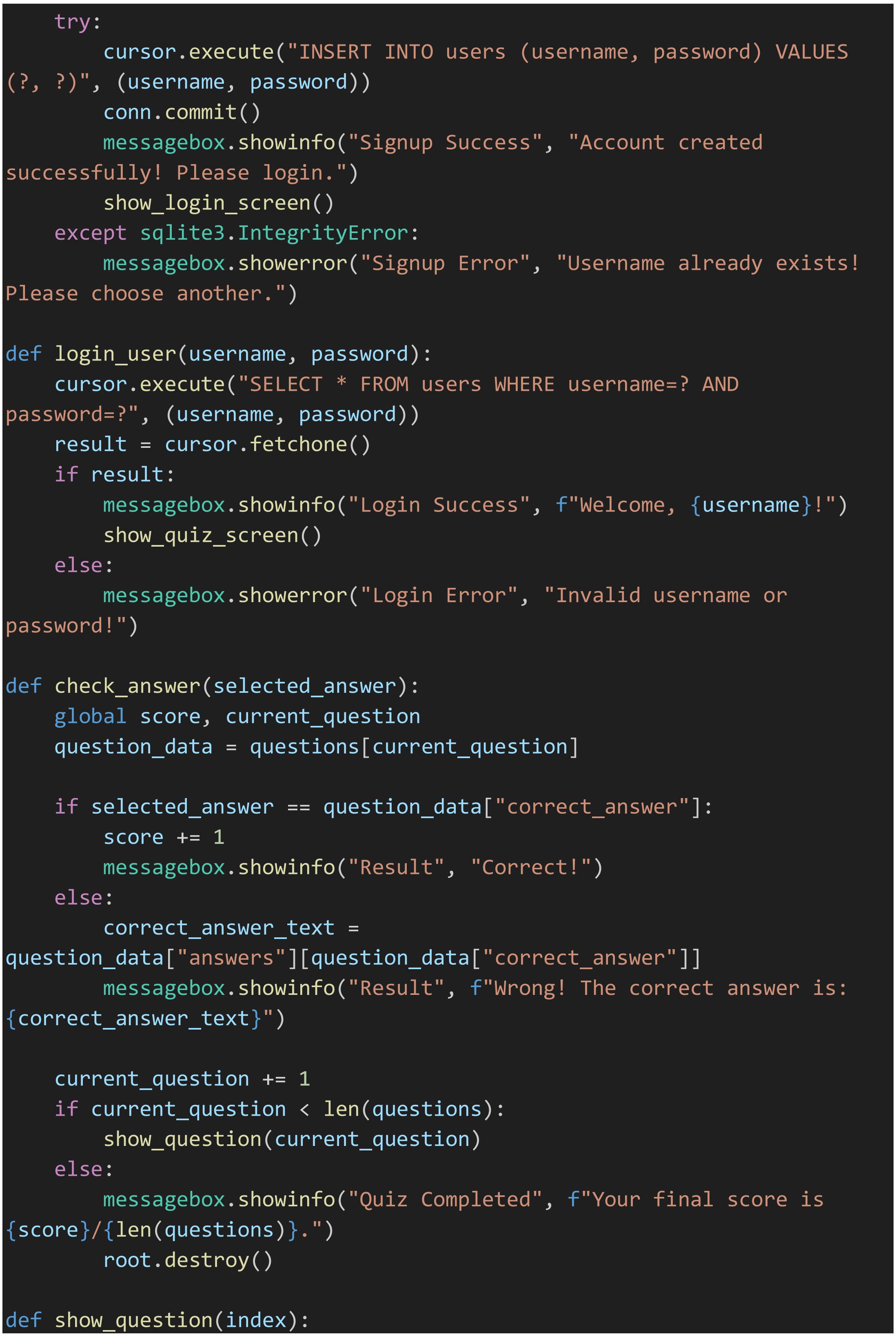
PYTHON PROGRAMMING AND FRAMEWORKS (22ITT32)

DEPARTMENT OF INFORMATION TECHNOLOGY

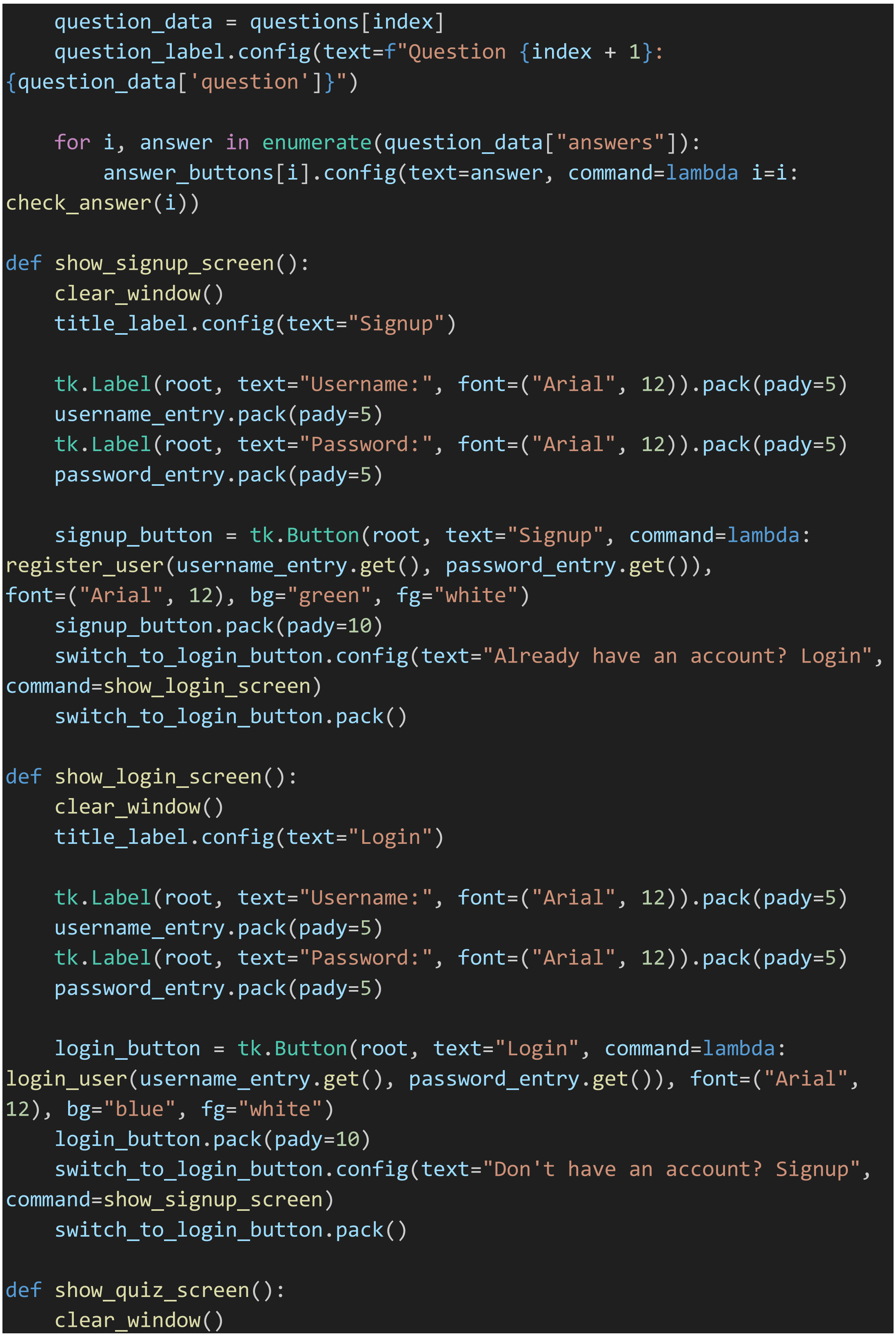
CODING : app.py



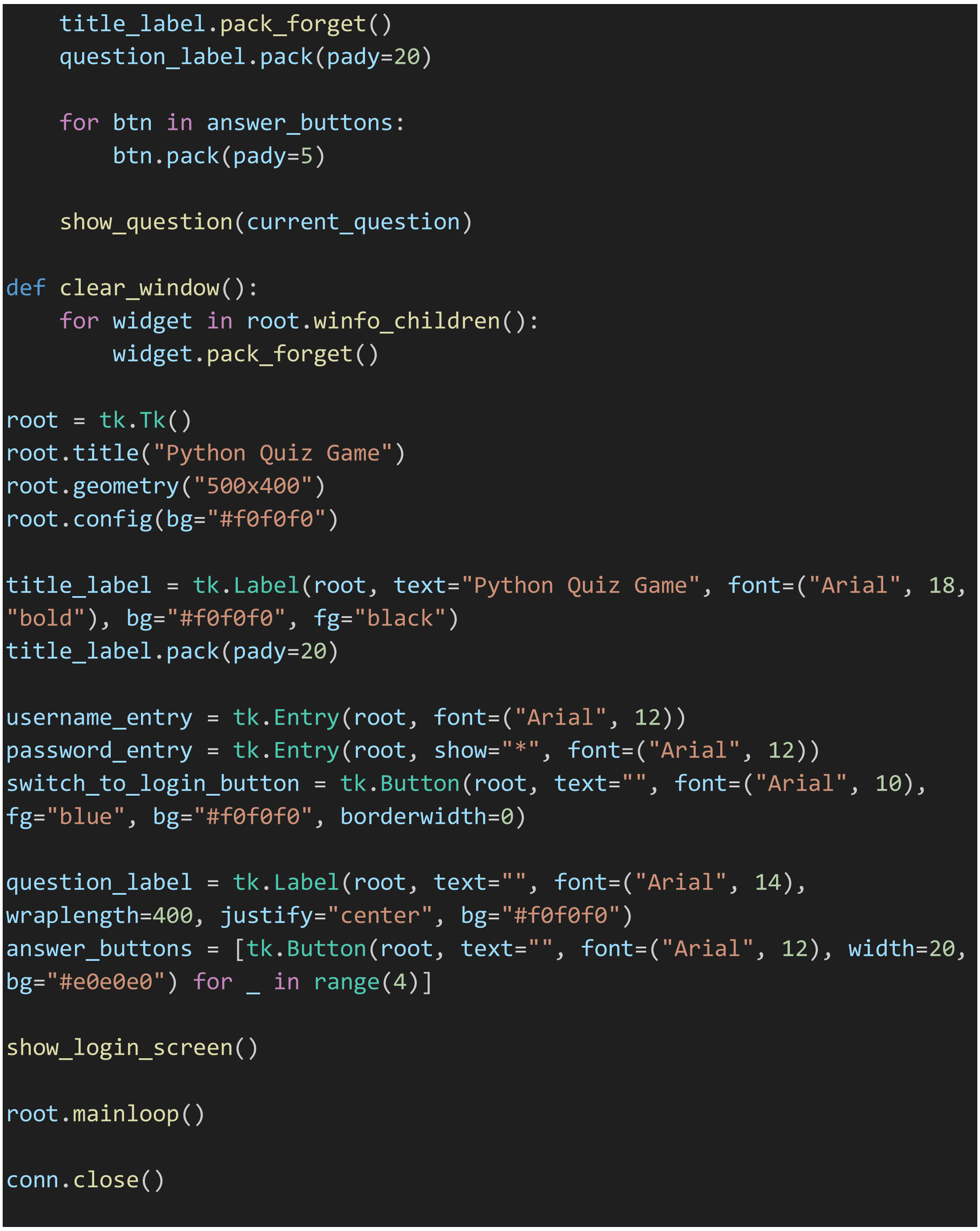
```
import tkinter as tk
from tkinter import messagebox
import sqlite3

conn = sqlite3.connect("quiz_app.db")
cursor = conn.cursor()
cursor.execute('''CREATE TABLE IF NOT EXISTS users (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    username TEXT UNIQUE,
    password TEXT
)''')
conn.commit()


questions = [
    {
        "question": "What is the correct file extension for Python
files?",
        "answers": [".pyth", ".pt", ".py", ".pyt"],
        "correct_answer": 2
    },
    {
        "question": "How do you create a variable with the numeric value
5 in Python?",
        "answers": ["x = 5", "int x = 5", "num x = 5", "x : 5"],
        "correct_answer": 0
    },
    {
        "question": "What is the correct syntax to output 'Hello World'
in Python?",
        "answers": ["echo 'Hello World'", "p('Hello World')",
"print('Hello World')", "printf('Hello World')"],
        "correct_answer": 2
    },
    {
        "question": "Which one of these is a mutable data type in
Python?",
        "answers": ["tuple", "string", "list", "list"],
        "correct_answer": 2
    },
    {
        "question": "Which of the following keywords is used for
function declaration in Python?",
        "answers": ["function", "def", "func", "declare"],
        "correct_answer": 1
    },
    {
        "question": "What is the output of 3 * 'Python'?",
        "answers": ["Python3", "Python Python Python", "Error",
"3Python"],
        "correct_answer": 1
    },
    {
        "question": "What is the output of len(['Python', 'Java',
'C++'])?",
        "answers": ["2", "3", "4", "Error"],
        "correct_answer": 1
    },
    {
        "question": "Which of the following is a Python framework for
web development?",
        "answers": ["React", "Django", "Spring", "Laravel"],
        "correct_answer": 1
    },
    {
        "question": "How can you create a comment in Python?",
        "answers": ["# This is a comment", "// This is a comment", "/*
This is a comment */", "-- This is a comment"],
        "correct_answer": 0
    },
    {
        "question": "Which Python keyword is used to handle
exceptions?",
        "answers": ["except", "try", "catch", "throw"],
        "correct_answer": 1
    }
]

score = 0
current_question = 0

def register_user(username, password):
```



```
    try:
        cursor.execute("INSERT INTO users (username, password) VALUES (?, ?)", (username, password))
        conn.commit()
        messagebox.showinfo("Signup Success", "Account created successfully! Please login.")
        show_login_screen()
    except sqlite3.IntegrityError:
        messagebox.showerror("Signup Error", "Username already exists! Please choose another.")

def login_user(username, password):
    cursor.execute("SELECT * FROM users WHERE username=? AND password=?", (username, password))
    result = cursor.fetchone()
    if result:
        messagebox.showinfo("Login Success", f"Welcome, {username}!")
        show_quiz_screen()
    else:
        messagebox.showerror("Login Error", "Invalid username or password!")

def check_answer(selected_answer):
    global score, current_question
    question_data = questions[current_question]

    if selected_answer == question_data["correct_answer"]:
        score += 1
        messagebox.showinfo("Result", "Correct!")
    else:
        correct_answer_text = question_data["answers"][question_data["correct_answer"]]
        messagebox.showinfo("Result", f"Wrong! The correct answer is: {correct_answer_text}")

    current_question += 1
    if current_question < len(questions):
        show_question(current_question)
    else:
        messagebox.showinfo("Quiz Completed", f"Your final score is {score}/{len(questions)}.")
        root.destroy()

def show_question(index):
```



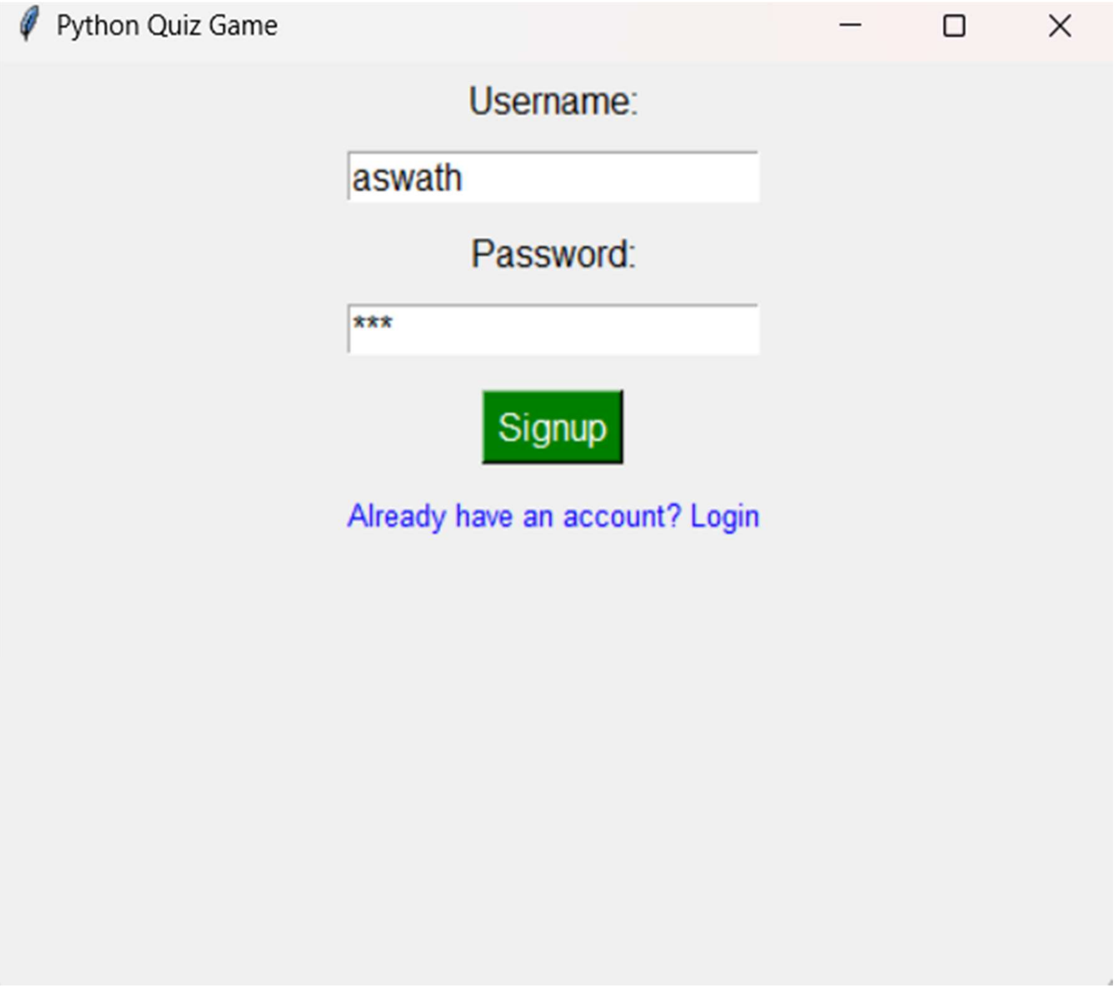
```
    question_data = questions[index]
    question_label.config(text=f"Question {index + 1}:
{question_data['question']}")

    for i, answer in enumerate(question_data["answers"]):
        answer_buttons[i].config(text=answer, command=lambda i=i:
check_answer(i))

def show_signup_screen():
    clear_window()
    title_label.config(text="Signup")

    tk.Label(root, text="Username:", font=("Arial", 12)).pack(pady=5)
    username_entry.pack(pady=5)
    tk.Label(root, text="Password:", font=("Arial", 12)).pack(pady=5)
    password_entry.pack(pady=5)

    signup_button = tk.Button(root, text="Signup", command=lambda:
register_user(username_entry.get(), password_entry.get()),
font=("Arial", 12), bg="green", fg="white")
    signup_button.pack(pady=10)
    switch_to_login_button.config(text="Already have an account? Login",
command=show_login_screen)
    switch_to_login_button.pack()

def show_login_screen():
    clear_window()
    title_label.config(text="Login")

    tk.Label(root, text="Username:", font=("Arial", 12)).pack(pady=5)
    username_entry.pack(pady=5)
    tk.Label(root, text="Password:", font=("Arial", 12)).pack(pady=5)
    password_entry.pack(pady=5)

    login_button = tk.Button(root, text="Login", command=lambda:
login_user(username_entry.get(), password_entry.get()), font=("Arial",
12), bg="blue", fg="white")
    login_button.pack(pady=10)
    switch_to_login_button.config(text="Don't have an account? Signup",
command=show_signup_screen)
    switch_to_login_button.pack()

def show_quiz_screen():
    clear_window()
```



```
    title_label.pack_forget()
    question_label.pack(pady=20)

    for btn in answer_buttons:
        btn.pack(pady=5)

    show_question(current_question)

def clear_window():
    for widget in root.winfo_children():
        widget.pack_forget()

root = tk.Tk()
root.title("Python Quiz Game")
root.geometry("500x400")
root.config(bg="#f0f0f0")

title_label = tk.Label(root, text="Python Quiz Game", font=("Arial", 18,
"bold"), bg="#f0f0f0", fg="black")
title_label.pack(pady=20)

username_entry = tk.Entry(root, font=("Arial", 12))
password_entry = tk.Entry(root, show="*", font=("Arial", 12))
switch_to_login_button = tk.Button(root, text="", font=("Arial", 10),
fg="blue", bg="#f0f0f0", borderwidth=0)

question_label = tk.Label(root, text="", font=("Arial", 14),
wraplength=400, justify="center", bg="#f0f0f0")
answer_buttons = [tk.Button(root, text="", font=("Arial", 12), width=20,
bg="#e0e0e0") for _ in range(4)]

show_login_screen()

root.mainloop()

conn.close()
```

OUTPUT :

Sign up :-



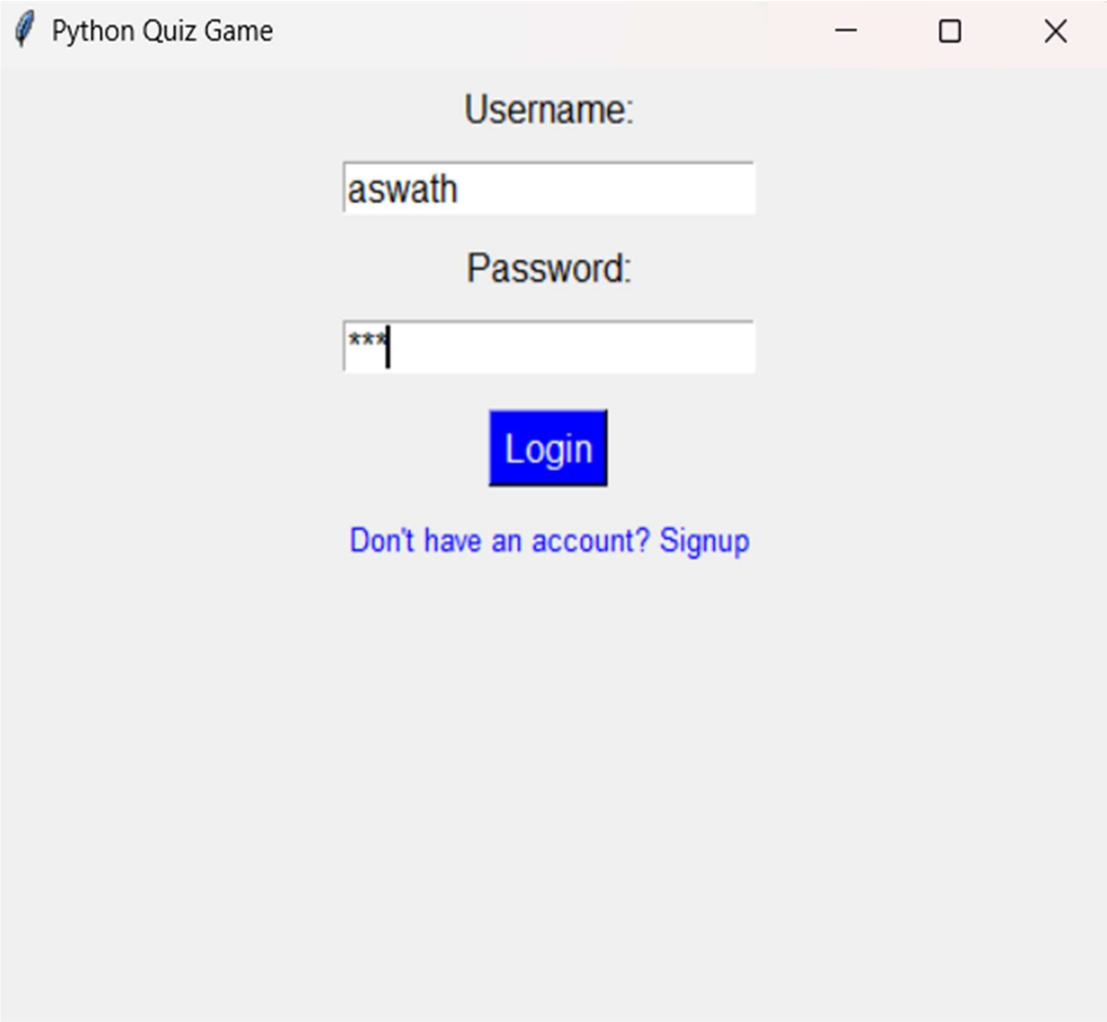
Username:

aswath

Password:

***

Login :-



Python Quiz Game

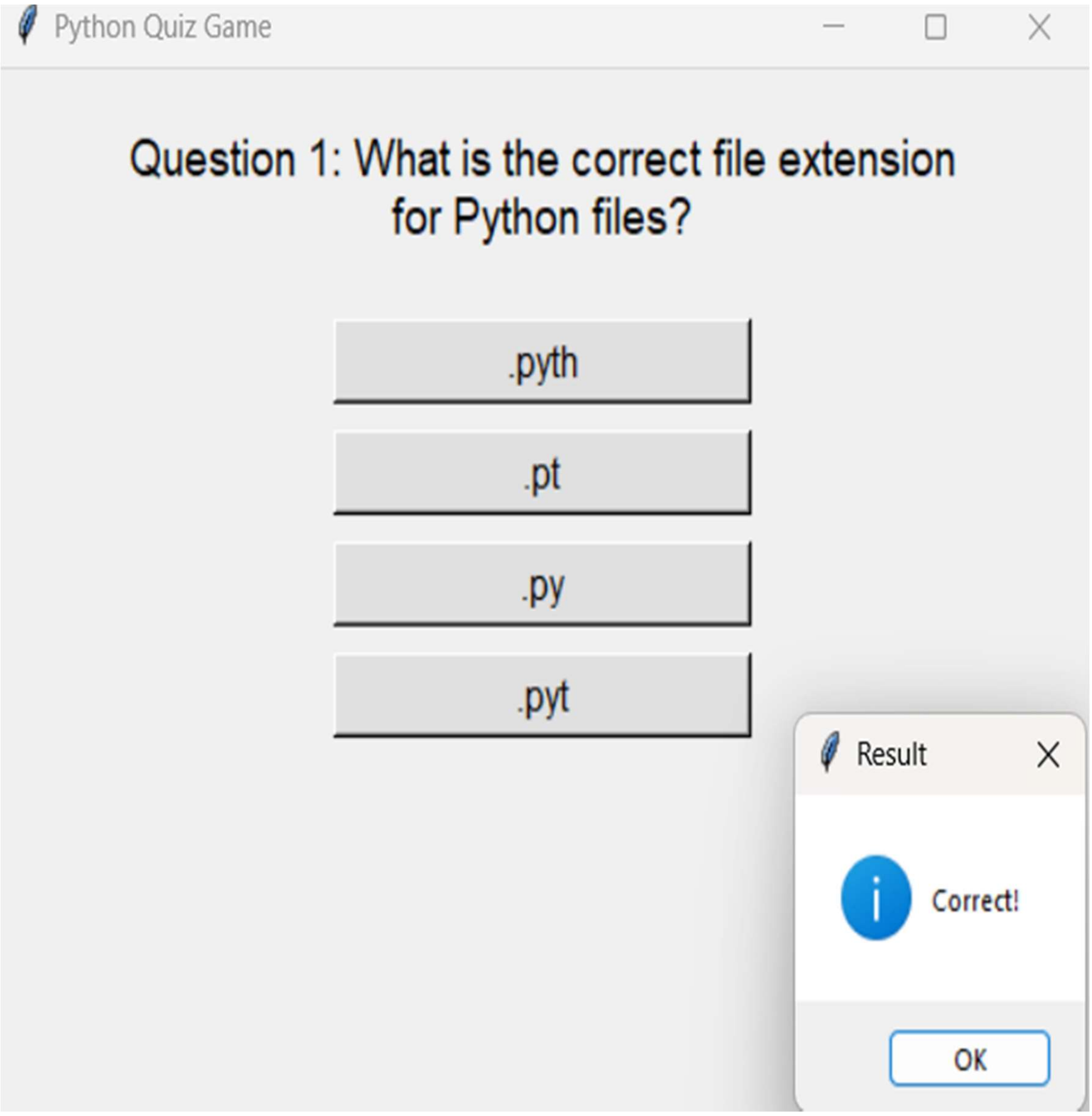
Username:

aswath

Password:

***

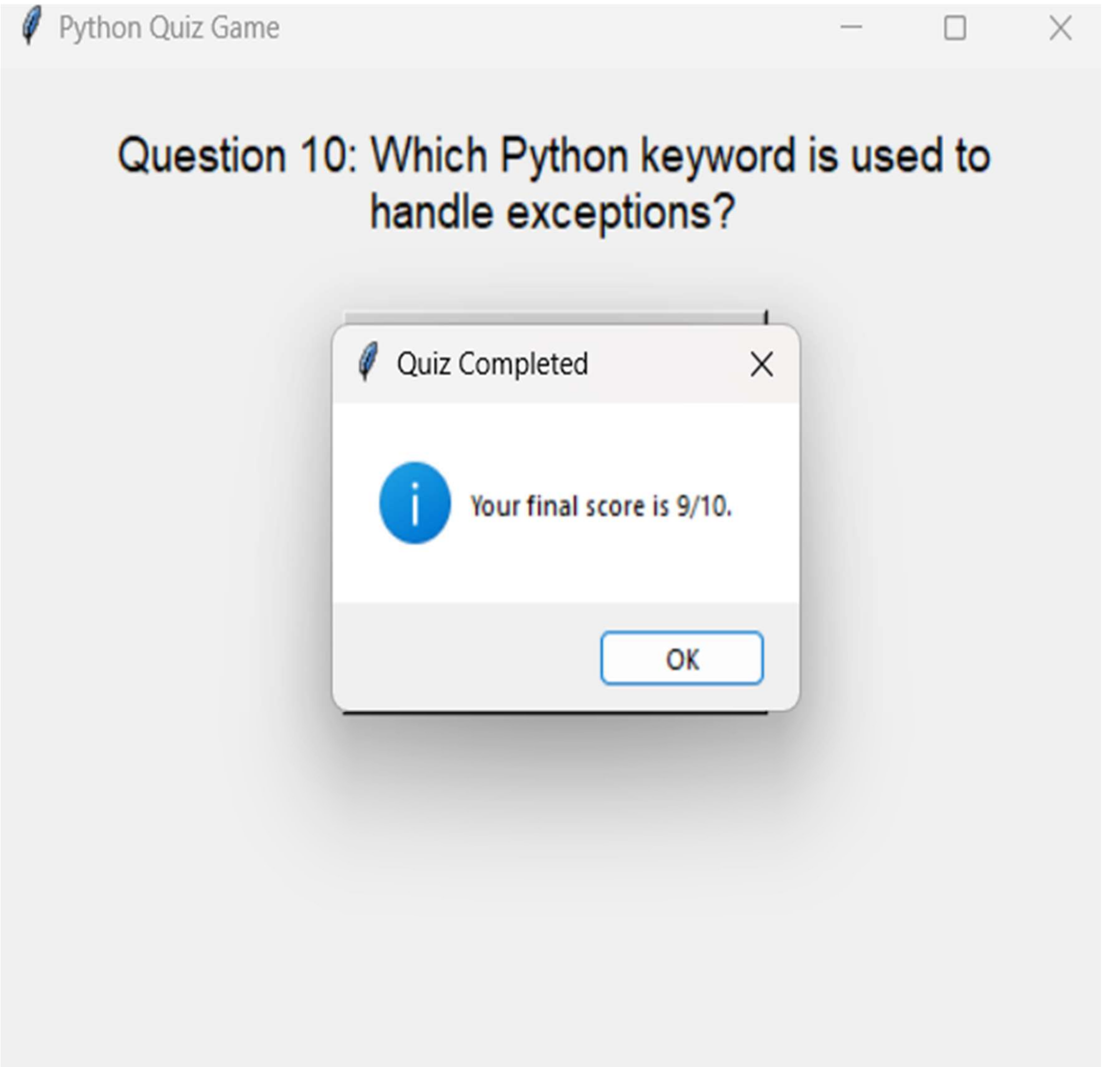
Ouiz :-



Question 1: What is the correct file extension for Python files?

Correct!

Score :-



Question 10: Which Python keyword is used to handle exceptions?

Your final score is 9/10.