Name-Aviral Jain

Project Name - Speed Typing Test in Python

Speed Typing Test in Python

Introduction-

The Speed Typing Test in Python is a simple program designed to measure an individual's typing speed and accuracy. This program randomly selects words from a predefined list and prompts the user to type them within a certain time limit. After completing the typing test, the program calculates various metrics such as the user's score, accuracy, average time per word, and typing speed in words per minute (WPM).

Functionality

The program uses the following steps to conduct the typing test:

Importing the necessary modules: The time module is imported to measure the time taken for typing, and the random module is used to randomly select words from the word list.

Defining the word list: The program includes a predefined list of words to be used in the typing test. You can customize this list to include different words or expand it as desired.

Calculating typing speed: The calculate_typing_speed function calculates the typing speed in WPM based on the total time taken and the number of words typed.

Typing test function: The typing_test function is the main function that conducts the typing test. It initializes variables for score, total time, and word count. It then prompts the user to type random words from the word list within a specific time limit.

Scoring and feedback: After each word is typed, the program checks if the user's input matches the selected word. If it matches, the score is incremented, and "Correct!" is displayed; otherwise, "Incorrect!" is displayed. The program provides immediate feedback for each word.

Metrics calculation and display: Once the typing test is completed, the program calculates and displays various metrics, including the score, accuracy, average time per word, and typing speed.

Usage

To use the Speed Typing Test program:

Ensure that Python is installed on your system. Copy the provided code into a Python editor or IDE. Customize the word list if desired by modifying the words variable. Run the program. The program will display random words from the list, and you need to type them accurately within the given time limit. After completing the test, the program will provide your score, accuracy, average time per word, and typing speed. The Speed Typing Test in Python is a useful tool for assessing and improving your typing skills. It can be

used for personal practice or incorporated into educational or training environments to evaluate typing proficiency.

Step 1: Import the necessary modules

In [3]:
```
import time
import random
```

Step 2: Define the list of words for the typing test

In [11]:
```
words = ['apple', 'banana', 'cat', 'dog', 'elephant']
```

Step 3: Define the function to calculate typing speed

In [5]:
```
def calculate_typing_speed(total_time, word_count):
    minutes = total_time / 60  # Convert total time to minutes
    wpm = word_count / minutes
    return wpm
```

Step 4: Define the typing test function

In [8]:
```
def typing_test():
    score = 0
    total_time = 0
    word_count = 0

    for i in range(5):  # Adjust the range for the number of words or sentences you want to include in the test
        word = random.choice(words)
        print('Word', i+1, ':', word)

        start_time = time.time()
        user_input = input('Type the word: ')
        end_time = time.time()

        total_time += end_time - start_time
        word_count += 1

        if user_input.strip() == word:
            score += 1
            print('Correct!')
        else:
            print('Incorrect!')

        print()  # Print a blank line for readability

    accuracy = (score / word_count) * 100
    average_time = total_time / word_count
    typing_speed = calculate_typing_speed(total_time, word_count)

    print('Test completed!')
    print('Score:', score, '/', word_count)
    print('Accuracy:', accuracy, '%')
    print('Average time per word:', round(average_time, 2), 'seconds')
    print('Typing Speed:', round(typing_speed, 2), 'WPM')
```

Step 5: Execute the typing_test function

In [12]:
```
typing_test()
```

```
Word 1 : cat
Type the word: cat
Correct!

Word 2 : cat
Type the word: cat
Correct!

Word 3 : apple
Type the word: apple
Correct!

Word 4 : elephant
Type the word: elephant
Correct!

Word 5 : elephant
Type the word: elephant
Correct!

Test completed!
Score: 5 / 5
Accuracy: 100.0 %
Average time per word: 1.99 seconds
Typing Speed: 30.1 WPM
```

In [18]:

```
import time
import random
import tkinter as tk
from tkinter import messagebox

# List of words for the typing test
words = ['apple', 'banana', 'cat', 'dog', 'elephant']

# Global variables
score = 0
total_time = 0
word_count = 0
difficulty_levels = {
    'Easy': 60,
    'Medium': 45,
    'Hard': 30
}
current_difficulty = 'Easy'
high_scores = []

# Create the main window
window = tk.Tk()
window.title('Typing Test')
window.geometry('400x300')

# GUI components
word_label = tk.Label(window, text='', font=('Arial', 24))
word_label.pack(pady=20)

entry_var = tk.StringVar()
entry_var.trace('w', lambda *args: check_input(entry_var.get()))

entry = tk.Entry(window, textvariable=entry_var, font=('Arial', 18))
entry.pack(pady=10)

score_label = tk.Label(window, text='Score: 0', font=('Arial', 12))
score_label.pack()

time_label = tk.Label(window, text='Time: 0s', font=('Arial', 12))
time_label.pack()

# Functions
def calculate_typing_speed(total_time, word_count):
    # Calculate typing speed in words per minute (WPM)
    minutes = total_time / 60  # Convert total time to minutes
    wpm = word_count / minutes
    return wpm

def update_word():
    # Update the word to be typed
    global word_count
    word = random.choice(words)
    word_label.config(text=word)
    word_count += 1
```

```
def check_input(user_input):
    # Check user input against the displayed word
    global score
    global total_time

    if user_input.strip().lower() == word_label.cget('text'):
        score += 1
        score_label.config(text='Score: ' + str(score))

        entry_var.set('')
        update_word()

def start_typing_test():
    # Start the typing test
    global score
    global total_time
    global word_count

    score = 0
    total_time = 0
    word_count = 0

    word_label.config(text='')
    score_label.config(text='Score: 0')

    start_button.config(state=tk.DISABLED)
    entry.config(state=tk.NORMAL)
    entry.focus()

    # Start the timer
    start_time = time.time()

    def update_timer():
        # Update the timer and check for test completion
        nonlocal start_time

        elapsed_time = round(time.time() - start_time)
        time_label.config(text='Time: ' + str(elapsed_time) + 's')

        if elapsed_time >= difficulty_levels[current_difficulty]:
            entry.config(state=tk.DISABLED)
            start_button.config(state=tk.NORMAL)
            save_high_score(score)

            messagebox.showinfo('Typing Test', 'Time\'s up!\nYour final score: ' + str(score))

        else:
            window.after(1000, update_timer)

    update_word()
    update_timer()

def save_high_score(score):
    # Save the high score in the leaderboard
    global high_scores
```

```
    high_scores.append(score)
    high_scores.sort(reverse=True)
    high_scores = high_scores[:5]  # Keep only the top 5 scores

def show_high_scores():
    # Display the high scores in a message box
    if not high_scores:
        messagebox.showinfo('High Scores', 'No high scores yet.')
    else:
        scores_text = '\n'.join(str(score) for score in high_scores)
        messagebox.showinfo('High Scores', 'Top Scores:\n' + scores_text)

# Difficulty level selection
def set_difficulty(difficulty):
    # Set the current difficulty level
    global current_difficulty
    current_difficulty = difficulty

# Create the difficulty buttons
difficulty_frame = tk.Frame(window)
difficulty_frame.pack(pady=10)

for difficulty in difficulty_levels.keys():
    button = tk.Button(difficulty_frame, text=difficulty, command=lambda d=difficulty: set_difficulty(d))
    button.pack(side=tk.LEFT, padx=5)

# Create the start button
start_button = tk.Button(window, text='Start', command=start_typing_test)
start_button.pack(pady=20)

# Create the high scores button
high_scores_button = tk.Button(window, text='High Scores', command=show_high_scores)
high_scores_button.pack(pady=10)

# Run the GUI
window.mainloop()
```

In [ ]:

In [ ]: