Challenge: Typecast Impersonator

Objective

The goal of this challenge is to impersonate a user that has a unique typing profile for some specified text. To be able to successfully impersonate a user, you will need to type the specified text just as the user would, including replicating KHTs (key hold times) and KITs (key interval times). Obviously, this is really only realistically possible by implementing a conflation of what you have learned from class lectures and hands-on sessions related to keystroke dynamics.

You will be provided with a precompiled Python program (both Python 2 and Python 3 versions for your convenience) that will randomly select text (a short sentence) and a typing profile. The typing profile includes the features (keys and key combinations), and the KHTs and KITs that correspond to the features. Note that the text and typing profile changes every time the precompiled Python program is executed.

It is suggested that you execute the precompiled Python program in one terminal window and your solution (a Python program) in a separate terminal window. Once the precompiled Python program is executed, you will be prompted to press **Enter** when you are ready. This will give you time to modify your solution with the details of the text and typing profile. When ready, you will have three seconds before a popup window appears. This popup window must have focus as you are typing the text.

Hint

Set a five second delay before your solution begins typing the text sample. This will allow you to change focus to the terminal window of the precompiled Python program and subsequently press **Enter**, allowing the popup window to appear and have focus before your program begins typing the text. Remember that the text typed must be terminated with **Enter** (i.e., type the text and press **Enter** to signal that the text is finished).

When you think you have successfully completed the challenge, **upload the following files** to the appropriate location on the provided team's google drive folder.

1. A text document (.txt) that provides the decoded message, as well as the steps that EVERY group member took to get that message. It is the responsibility of every group member to edit their own portion of this text file.
2. The python code that you used to complete the challenge
3. Any other scripts that you created to assist you (if applicable)

You will be graded on how well:

(1) your code is written. Does it work? Is it well commented and easy to follow? (50%);
(2) you document the process that you went through to decrypt the covert message (for every single member of your team) (40%); and
(3) you submit the required files (10%).

GOOD LUCK!