Ein Bild, das Text enthält.

Automatisch generierte BeschreibungReaktionstest

Arbeitsauftrag

# Aufgabenstellung

Es soll ein Programm geschrieben werden, welches die Reaktionsfähigkeit überprüft. Dazu soll eine grafische Oberfläche mit einem Schalter erzeugt werden. Nach dem Starten des Programmes soll eine zufällige Zeit vergehen (zwischen 2s und 10s). Nach dieser Zeit soll dem Benutzer signalisiert werden, das der Reaktionstest starten, z.B. über ein rotes Label. Auf dieses Signal soll der Benutzer reagieren, indem er mit der Maus auf den Schalter drückt. Die Zeit zwischen dem Signal und dem Drücken auf den Schalter soll gemessen und ausgegeben werden.

## Vorgehensweise

1. Erstelle das objektorierentierte Grundgerüst für ein Fenster mit tkinter
2. Erstelle eine Methode create\_widgets. Erstelle dort den Schalter und alle anderen Widgets, die in der Oberfläche angezeigt werden sollen. Rufe diese Methode im Konstruktor auf. (Beispiel: Hello World im Skriptum von Herrn Signitzer)  
   Vergiss nicht den richtigen Layout-Manager zu verwenden.  
   Überprüfe, ob das Fenster richtig aussieht.
3. Suche dir im Internet oder im Skriptum von Herrn Signitzer, wie man das Drücken eines Schalters auswertet. Schreibe eine Methode show\_result, welche vorerst nur etwas auf die Konsole ausgibt. Überprüfe, ob der Code bis hierher funktioniert: Wenn du den Schalter drückst, soll die Ausgabe auf der Konsole stattfinden. Wenn du mehrmals drückst, sollte der Text mehrmals erscheinen.
4. Um etwas nach einer gewissen Zeit starten zu lassen gibt es beim Frame eine Methode after. Diese kann mit der Zeit und einer anderen Methode aufgerufen werden. Diese Methode wird dann nach dieser Zeit aufgerufen. Schreibe zwei Methoden:
   1. init\_timer: In dieser Methode ist die Zufallszahl zu berechnen, anschließend die Methode after aufzurufen
   2. signal\_user: In dieser soll vorerst auch nur ein Text auf die Konsole ausgegeben werden. Überprüfe, ob das passiert.
5. Jetzt kümmern wir uns darum, dass das Signal angezeigt wird. Hierzu müssen wir folgende Änderungen am Code vornehmen:
   1. In der Methode create\_widget muss das Label, welches für das Signal zuständig ist, unsichtbar gemacht werden. Das funktioniert mit label.pack\_forget().
   2. In der Methode signal\_user muss das Label dann wieder sichtbar gemacht werden.  
      Überprüfe, ob das so funktioniert.
6. Zum Schluss müssen wir noch die Zeit messen, welche zwischen dem Signal und dem Drücken vergeht. Hierfür verwenden wir die Bibliothek time, welche eine Standardbibliothek ist. Mit time.time() kann die aktuelle Zeit ermittelt werden. Folgendes ist konkret zu tun:
   1. Um die Zeit zu messen, müssen wir uns in einem Attribut in der Methode signal\_user die Zeit merken.
   2. In der Methode show\_result muss dann die Differenz der aktuellen Zeit und der gespeicherten Zeit berechnet werden. Wie das funktioniert, findet man im Internet. Gib die Differenz auf der Konsole aus.  
      Überprüfe, ob das Programm so funktioniert
   3. Stelle das Ergebnis in der Oberfläche dar. Überlege dir selbst, wie du das machen möchtest. Du kannst auch gerne in den Widgets von tkinter schmökern.

## Erweiterungen für Schnelle:

* Lade die Zeitspanne für das Anzeigen des Signals aus der Datei settings.txt – Standardwerte sind 2-10
* Lass den Benutzer mehrere Reaktionstests hintereinander durchführen, ohne dass er das Programm neu starten muss. Wie könnte das funktionieren?
* Erstelle eine Bestenliste in der Oberfläche, mit den besten drei Ergebnissen des aktuellen Programmstarts
* Speichere die besten Versuche in einer Datei best\_tries.txt und zeige somit die besten Werte aller Zeiten an.