

```
def b_click(b):
    global clicked, count
    if b["text"] == " " and clicked == True:
        b["text"] = "X"
        clicked = False
        count += 1
        checkifwon()
    elif b["text"] == "0" and clicked == False:
        b["text"] = "0"
        clicked = True
        count += 1
        checkifwon()
    else:
        messagebox.showerror("Tic Tac Toe", "Dieses Feld wurde bereits ausgewählt.
```

WPU Programmieren

10R1 / 10R2

Python- Programmierprojekt

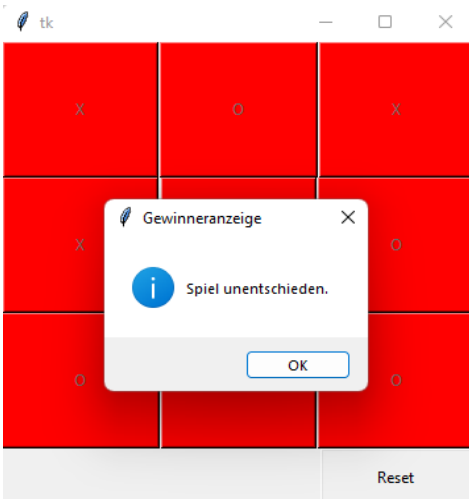
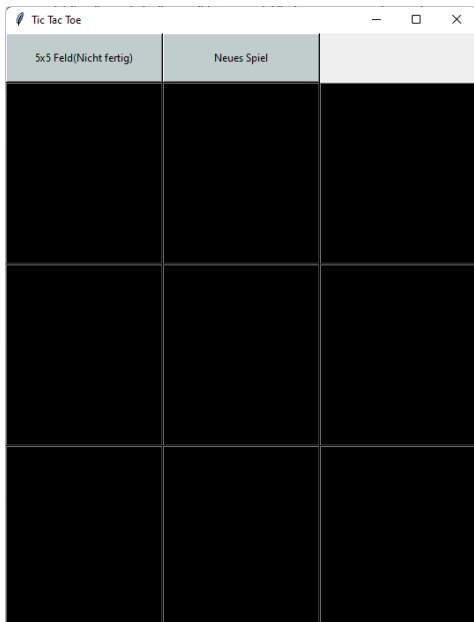
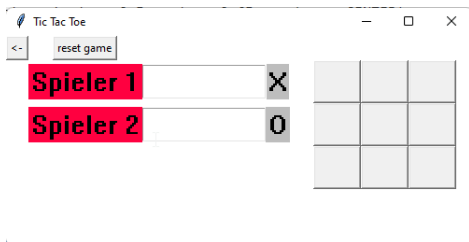
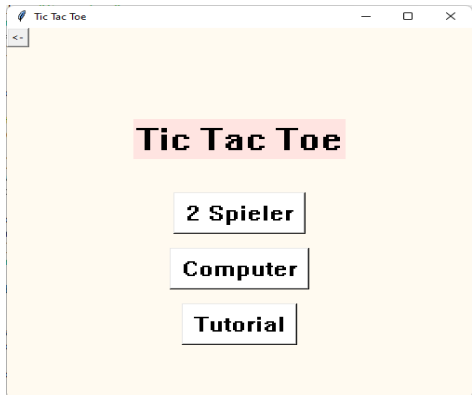


Aufgabe:



Erstellt ein
TicTacToe-Spiel mit
grafischer
Benutzeroberfläche

Die Ergebnisse



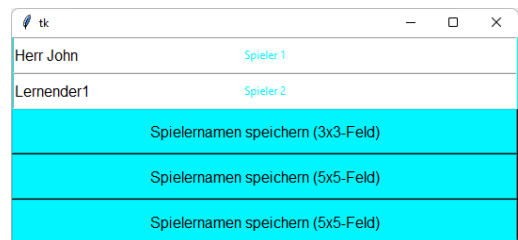
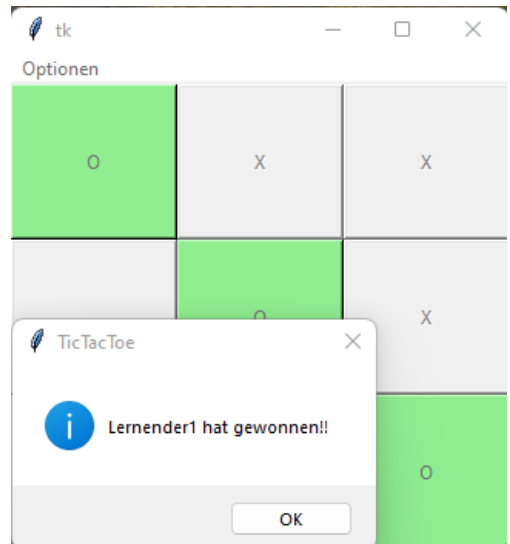
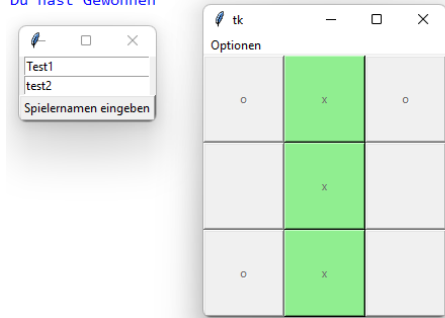
TIC TAC TOE

Spielregeln:

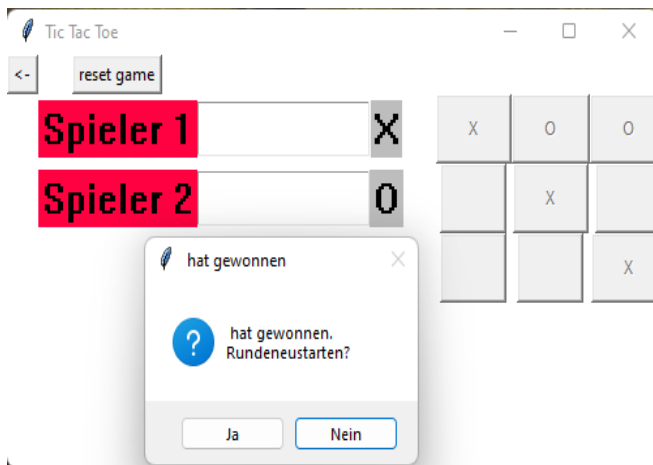
Das Spiel TicTacToe. Es ist ein Feld vorhanden von 3 x3. Es gibt immer 2 Spieler welche abwechselnd x oder o in die jeweiligen Felder einsetzen. Das Ziel dabei ist alle drei Zeichen hintereinander horizontal, senkrecht oder diagonal hat zu setzen. Die Person, die es zuerst geschafft hat, hat gewonnen. 0 BEGINNT:

Test1 ist (X). test2 ist (O).

Du hast Gewonnen



Beispielbetrachtung



Bei diesem Spiel zwischen zwei menschlichen Spielern hat „X“ gewonnen. Eine Möglichkeit zum Neustarten des Spiels wird direkt mitgeliefert.

Die Spielfelder bestehen aus programmierten Knöpfen (Buttons). Jeder Knopf hat dabei seinen eigenen Programmcode:

Mit `if...else` wird zunächst ermittelt, welcher Spieler gerade am Zug ist.

Anschließend wird dem Knopf (hier Knopf 0) der entsprechende Buchstabe als „Text“ zugewiesen und anschließend der Knopf deaktiviert. Dadurch kann keiner mehr den Knopf drücken.

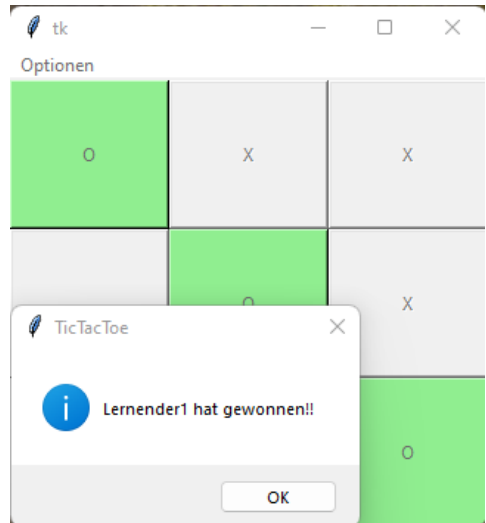
```
def co0():
    global Counter
    if Counter%2==1:
        but0["text"] = "O"
        but0["state"]=DISABLED
        field[0] = "O"
        Counter=Counter+1

    else:
        but0["text"]="X"
        but0["state"]=DISABLED
        field[0]="X"
        Counter=Counter+1
    if Counter>=5:
        checkWin()
```

Beispielbetrachtung

Jedes Mal wenn ein Knopf gedrückt wurde, muss das Programm überprüfen, ob es einen Gewinner gibt.

Für dieses Beispiel könnte der Programmcode so aussehen:



```
elif b1["text"] == "O" and b5["text"] == "O" and b9["text"] == "O":
    b1.config(bg="light green")
    b5.config(bg="light green")
    b9.config(bg="light green")
    winner = True
    disable_all_buttons()
    messagebox.showinfo("TicTacToe", po_win)
```

Hier wird verglichen, ob der Text auf den Knöpfen jeweils gleich „O“ ist. Ist dies der Fall, dann bekommen nach diesem Programmcode alle Gewinnerknöpfe die Farbe „Hellgrün“, alle Knöpfe werden abgeschaltet und ein Gewinner wird über eine separate Meldung ausgerufen.

Diese Gewinnüberprüfung muss für alle Möglichkeiten für „X“ und „O“ geschrieben werden!

In diesem Beispiel wurden dafür 134 Zeilen Code aufgeschrieben (inklusive dem Fall: Unentschieden)

Rückschau

Insgesamt haben **15 Schülerinnen und Schüler** in **Partner- oder Teamarbeit** in einem Zeitraum von **28 Schulstunden** über **10 Wochen** verteilt an **6 unterschiedlichen Versionen** von TicTacToe gearbeitet.

Die Lernenden haben den kompletten Ablauf **selbst organisiert** und ihre Zeit **eigenständig** verplant. Jedes Endprodukt hat ungefähr **1000 Zeilen Programmcode** und wurde im Unterricht mindestens **3 Mal** der gesamten Gruppe **vorgezeigt**.

Probleme wurden **selbstständig recherchiert** und behoben. Die Planung erfolgte über klassische Methoden zur **Projektplanung**, wie **Kanban-Boards** und **Stand-Up-Meetings**

