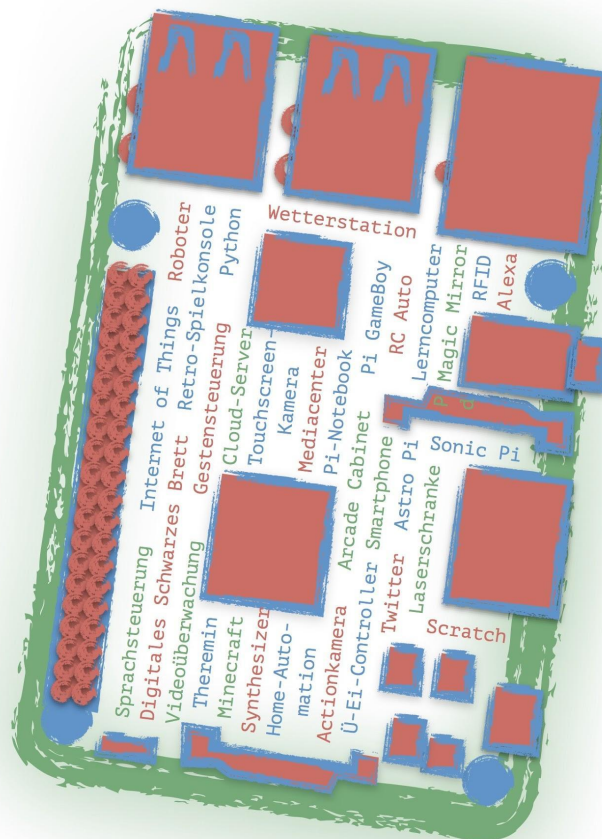


Ein Text-Adventure coden

Die ersten Computerspiele hatten noch keine Grafik, sie bestanden nur aus Text. Spaß gemacht haben sie trotzdem! Wir zeigen Dir, wie Du mit Python Dein eigenes Text-Adventure programmieren und ganz nach deinen Wünschen erweitern kannst.



DIE RÄUME

In Text-Adventures bewegt man sich wie auf einem Schachbrett. Von jedem Raum aus kann man entweder nach Norden, Osten, Süden oder Westen gehen. Wir beginnen daher unser Programm auch damit, dass wir alle erlaubten Richtungen festlegen. Dazu schreiben wir in einem neuen Python3-Programm eine Liste "erlaubte_richtungen", die wie folgt aussieht:

```
erlaubte_richtungen = ["norden", "sueden", "osten", "westen"]
```



Die Liste ist eine der wichtigsten Datentypen in Python.
Listen sind eine Sammlung von Werten mit Ordnung.
Die Sammlung beginnt bei 0!
Erkennungsmerkmal: eckige Klammern [].

Nun muss man sich überlegen, wie viele Räume man in seinem Spiel verwenden möchte und wie diese angeordnet sind. Hier ist ein Beispiel für einen Spielplan mit vier Räumen:

Raum 0	Raum 1
Raum 2	Raum 3

Nun muss man aufbauend auf diesem Plan notieren, welche Bewegungsmöglichkeiten die Spielfigur in jedem Raum hat. Dazu erstellen wir ein sogenanntes Dictionary mit geschweiften Klammern.

```
norden= {0: None, 1: None, 2:0, 3:1}
```



Dictionaries enthalten Schlüssel-Wert-Paare. Jedem Schlüssel ist ein Wert zugeordnet. Der Wert kann wiederum eine Liste sein. Das Erkennungsmerkmal für Dictionaries sind geschweifte Klammern!

Dieser Code besagt, dass man in Raum 0 und 1 nicht nach Norden gehen kann (deswegen "None"). Wenn man von Raum 2 aus nach Norden geht, gelangt man in Raum 0 und nördlich von Raum 3 ist Raum 1. Das Gleiche macht man nun noch für die anderen Himmelsrichtungen.

```
sueden= {0:2, 1:3, 2:None, 3:None}  
osten= {0:1, 1:None, 2:None, 3:None}  
westen= {0:None, 1:0, 2:None, 3:None}
```

Zum Navigieren benötigen wir jetzt noch ein weiteres Dictionary (dazu später mehr):

```
kompass = {"norden": norden, "sueden": sueden, "osten": osten, "westen":  
westen}
```

Nun benötigt jeder Raum eine kurze Beschreibung. Dazu erstellen wir ein weiteres Dictionary namens "beschreibung".

```
beschreibung= {  
    0: "Du bist in Raum 0. Du kannst nach Sueden oder Osten gehen.",  
    1: "Du bist in Raum 1. Du kannst nach Sueden oder Westen gehen.",  
    2: "Du bist in Raum 2. Du kannst nach Norden gehen.",  
    3: "Du bist in Raum 3. Du kannst nach Norden gehen."  
}
```



Man könnte diese Zeilen auch hintereinander schreiben, aber so ist es wesentlich übersichtlicher. Achte darauf, die Komma am Ende jeder Zeile und die Klammer am Ende nicht zu vergessen.

Nun muss man festlegen, in welchem Raum das Spiel beginnt und wo es endet. Wir beginnen in Raum 0 und enden in Raum 3.

```
aktueller_raum = 0  
zielraum = 3
```

Zudem brauchen wir noch eine neue Variable, die wir "wohin" nennen. Diese Variable speichert im Programm unsere Eingaben. Zunächst ist sie aber noch leer:

```
wohin = ""
```

Nun benötigen wir den eigentlichen Code, um uns auf der Karte bewegen zu können. Dazu leiten wir eine Schleife ein. Jede Codezeile, die nun folgt, wird nur ausgeführt, solange die Variable "aktueller_raum" nicht leer ist:

while(aktueller_raum is not None):



Schleifen sind programmierbare Wiederholungen in Python. Die while-Schleife ist eine davon. Sie läuft solange, bis eine Bedingung nicht mehr gilt.
Vorsicht vor Endlosschleifen!

Nun soll als Erstes die Beschreibung des Raums angezeigt werden, in dem der Spieler steht. Achte darauf, dass die folgende Zeile eingerückt sein muss.

print(beschreibung[aktueller_raum])

Dieser Code greift auf das "Dictionary" zurück und fügt die passende Beschreibung ein. Das ist praktisch, weil man das Programm später sehr einfach erweitern bzw. verändern kann. Dazu muss man nämlich nicht in den eigentlichen Code eingreifen, sondern lediglich die Dictionaries ändern.

Als nächstes soll der Spieler eingeben, in welche Richtung er gehen möchte. Dafür benötigen wir eine Eingabe, die mit dem Befehl "input" eingeleitet wird.

wohin = input("In welche Richtung möchtest Du gehen?").lower()



Der Zusatz ".lower()" sorgt dafür, dass die Antwort in Kleinbuchstaben umgewandelt wird, sodass es z.B. egal ist, ob jemand "Norden", "NORDEN" oder "norden" schreibt.

Nun soll überprüft werden, ob die Antwort wirklich eine der vier Himmelsrichtungen ist, ansonsten wird die Eingabe wiederholt:

```
while wohin not in erlaubte_richtungen:  
    wohin= input("In welche Richtung moechtest Du gehen?").lower()
```

Nun soll das Programm überprüfen, ob der Spieler überhaupt in die angegebene Richtung gehen kann. Hier kommt nun unser "Kompass-Dictionary" ins Spiel:

```
if kompass [wohin][aktueller_raum] is not None:
```

Der folgende Code wird also nur ausgeführt, wenn eine Himmelsrichtung eingegeben wurde, in die man auch wirklich gehen kann. Wenn dem so ist, ändert sich der aktuelle Raum, die Variable "aktueller_raum" wird also überschrieben:

```
    aktueller_raum = kompass[wohin][aktueller_raum]
```

Nun soll überprüft werden, ob der Zielraum erreicht wurde und wenn es der Fall ist soll die while-Schleife mit dem break abgebrochen werden.

```
    if aktueller_raum == zielraum:  
        aktueller_raum = None  
        print("Du hast den Zielraum gefunden!")  
        break
```

Nun muss nur noch eine Meldung erfolgen, falls die Eingangsbedingung nicht erfüllt wurde, also eine falsche Himmelsrichtung eingegeben wurde:

```
    else:  
        print("---Hier geht es nicht lang.---")
```

Hier ist noch einmal das gesamte Programm

```
erlaubte_richtungen = ["norden", "sueden", "osten", "westen"]
norden= {0: None, 1: None, 2:0, 3:1}
sueden= {0:2, 1:3, 2:None, 3:None}
osten= {0:1, 1:None, 2:None, 3:None}
westen= {0:None, 1:0, 2:None, 3:None}
kompass = {"norden": norden, "sueden": sueden, "osten": osten,
"westen": westen}
beschreibung= {
    0: "Du bist in Raum 0. Du kannst nach Süden oder Osten gehen.",
    1: "Du bist in Raum 1. Du kannst nach Süden oder Westen gehen.",
    2: "Du bist in Raum 2. Du kannst nach Norden gehen.",
    3: "Du bist in Raum 3. Du kannst nach Norden gehen."
}
aktueller_raum = 0
zielraum = 3
wohin = ""
while(aktueller_raum is not None):
    print(beschreibung[aktueller_raum])
    wohin = input("In welche Richtung mochtest Du gehen?").lower()
    while wohin not in erlaubte_richtungen:
        wohin=input("In welche Richtung mochtest Du gehen?").lower()
    if kompass [wohin][aktueller_raum] is not None:
        aktueller_raum = kompass[wohin][aktueller_raum]
        if aktueller_raum == zielraum:
            aktueller_raum = None
            print("Du hast den Zielraum gefunden!")
            break
    else:
        print("---Hier geht es nicht lang.---")
```