

Aliens einfangen - Spiel mit Tastatursteuerung

Ziele:

Wir entwickeln ein einfaches **Spiel**, bei dem man ein Raumschiff mit der Tastatur steuert und auf Alienfang geht.

Die Aliens erscheinen *zufällig* irgendwo auf dem Feld. Pro Alien gibt es einen Punkt. Das Ziel ist es, möglichst viele Aliens zu sammeln.



1. Steuerung mit der Tastatur

Um das Raumschiff zu steuern, muss das Programm entscheiden, was zu tun ist, wenn eine Taste gedrückt wird. Dazu hat jede Taste eine eigene Nummer. (siehe Bild)

Esc 27	F1 112				F2 113				F3 114				F4 115				F5 116				F6 117				F7 118				F8 119				F9 120				F10 121				F11 122				F12 123				PrScr 44	Scrlk 145	Break 19
~ 192	! 49	@ 50	# 51	\$ 52	% 53	^ 54	& 55	* 56	{ 57	} 58	[189	= 187	Backspace 8	Home 36	End 35	PgUp 33																																			
Tab 9	Q 81	W 87	E 69	R 82	T 84	Y 89	U 85	I 73	O 79	P 80	[219] 221	\ 220	Insert 45	Delete 46	PgDn 34																																			
Caps Lock 20	A 65	S 83	D 68	F 70	G 71	H 72	J 74	K 75	L 76	; 186	' 222	Enter 13																																							
Shift 16	Z 90	X 88	C 67	V 86	B 66	N 78	M 77	< 188	> 190	? 191	Shift 16																																								
Ctrl 17	Win 91	Alt 18	32				Alt 18	Win 92	Menu 93	Ctrl 17																																									
											↑ 38																																								
											← 37	↓ 40	→ 39																																						

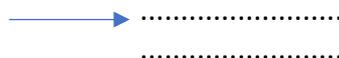
a) Zuerst fragst Du in der repeat-Schleife die Taste mit `key = getKeyValue()` ab:

```
repeat:
    key = getKeyValue()
    if key == 37:
        heading(-90)
```

Anschliessend entscheidet das Programm mit `if` ob die Taste (hier die linke) gedrückt wurde. Wenn ja, wird der Befehle `heading(-90)` ausgeführt, der die Turtle nach links ausrichtet.

b) Ergänze den Code mit den weiteren Richtungstasten (nur `if key == ??`) und teste Dein Programm. Die Turtle muss sich nun je nach Tasten nach links, rechts, oben oder unten ausrichten.

```
repeat:
    key = getKeyValue()
    if key == 37:
        heading(-90)
```



- Folgende Ausrichtungen bedeuten :
- `heading(0)` nach oben
 - `heading(90)` nach rechts
 - `heading(-90)` nach links
 - `heading(180)` nach unten

c) Nun ist es an der Zeit, die Turtle, fahren zu lassen. Ergänze zwei Zeilen direkt nach repeat:

```
repeat:
    wrap()
    forward(10)
    key = getKeyCode()
    if key == 37:
        heading(-90)
```

Mit wrap() wird die Turtle nachdem Verlassen des Feldes unten oder auf den Seiten wieder ins Feld fahren.

d) Nun kannst Du der ein Raumschiffbild geben und den Stift heben mit pu()

```
makeTurtle("sprites/spaceship.gif")
pu()
```

2. Die Aliens

a) Der Befehl alien() soll das Bild eines Alien zufällig irgendwo ins Fenster setzen. es gibt einen Befehl, der uns eine zufällige Zahl erzeugen kann. Dazu musst Du ganz oben zuerst eine Bibliothek einfügen:

```
from random import*
```

Mit x=randint(-300,300) wird dann eine zufällige Zahl zwischen -300 und 300 erzeugt und in x abgespeichert. Genauso funktioniert es mit y=randint(-200,200).

Definiere nun den gesamten Befehl alien() vor der repeat-Schleife.

```
def alien():
    x=randint(-300,300)
    y=randint(-200,200)
    setPos(x,y)
    heading(0)
    drawImage("sprites/alien.png")
```

b) Setze nun einem ein Alien (vor der repeat-Schleife) mit dem Befehl alien()

3. Die Aliens einsammeln

a) Nun kommen wir zum Teil , wo die Aliens eingesammelt werden. Wie kann das Programm feststellen, wann das Raumschiff bei einem Alien ist?

Wir machen das über die Farbe. Der Befehl getPixelColorStr() erkennt die Farbe unterhalb des Raumschiffs. Füge die folgende Zeil im repeat-Teile direkt unter den Tastaturbock ein:

```
color=getPixelColorStr()
```

- b) Nun überprüfen wir, ob die Farbe NICHT schwarz ist. Das ist dann der Fall, wenn das Raumschiff über dem Alien ist.

```

if color!="black":
    pushState()           ← Position des Raumschiffs speichern
    clear("black")       ← Bildschirm löschen (schwarz)
    alien()              ← Alien zufällig irgendwo setzen
    popState()           ← Position des Raumschiffs wieder abrufen
    showTurtle()         ← Raumschiff wieder einblenden
    
```

Wenn dies der Fall ist, müssen wir die Position des Raumschiffs kurz speichern mit `pushState()` (weil wir gleich die Alien mit der Turtle irgendwo hin setzen und dann wieder zur momentan Position zurückkehren müssen). Dann wird der Bildschirm gelöscht (schwarz), die Alien zufällig irgendwo hin gesetzt. Die Position des Raumschiffs wird wieder abgerufen aus dem Speicher und das Raumschiff wieder eingeblendet.

4. Punkte (eingesammelte Aliens) zählen und anzeigen

- a) Jedesmal, wenn ein Alien eingesammelt ist, soll es einen zusätzlichen Punkt geben. Erstelle dazu ganz oben einen eigenen Befehl `punkteAnzeigen()`:

```

def punkteAnzeigen():
    setPos(280,-280)
    label("Punkte",punkte)
    savePlayground()
    
```

`savePlayground()` braucht es, damit die Anzeige nicht immer gelöscht wird.

- b) Diesen Befehl rufst Du jetzt im `if`-Teil (3b) zwischen ein.

```

clear("black")
punkte=punkte+1
punkteAnzeigen()
alien()
    
```

5. Spiellevel eingeben

Du kannst nun einen Spiellevel eingeben lassen. Der soll einfach das Raumschiff schneller fliegen lassen. Dazu verlangst Du VOR der `repeat`-Schleife eine Eingabe:

```

level=input("Level? (1-4)")
    
```

Nun kannst Du beim `forward(6)` die 6 durch `level` austauschen, damit die Geschwindigkeit gemäss der Eingabe funktioniert.

```

forward(level)
    
```