

13. COMPUTERANIMATIONEN

■ DU LERNST HIER...

wie du mit Python animierte Computergrafiken erstellst.

■ MUSTERBEISPIELE

Eine Animation besteht aus Einzelbildern, die sich nur wenig ändern und Schritt um Schritt zeitlich nacheinander dargestellt werden. Da das menschliche Auge nur rund 25 Bilder pro Sekunde erfassen kann, ergibt sich wie beim Film eine fließende, flackerfreie Bewegung, wenn die Bildfolge genügend schnell gezeigt wird.

Als Beispiel zeichnest du mit der Funktion *propeller()* einen 3-blättrigen Propeller und lässt ihn langsam drehen. Die gedrehten Propeller-Positionen zeichnest du am einfachsten mit Hilfe des Befehls *setHeading(a)*, wobei der Winkel *a* zu Beginn 0 und für jede neu Position um 10 Grad grösser ist.

Den Animationsablauf kannst du so beschreiben:

Propellerrichtung auf 0 Grad initialisieren

Wiederhole:

- Bild löschen
- Propeller zeichnen
- 40 Millisekunden warten
- Propellerrichtung ändern



Programm: [[▶ Online-Editor](#)]

```
from gturtle import *

def rightArc():
    repeat 30:
        fd(4)
        rt(3)

def leftArc():
    repeat 30:
        fd(4)
        lt(3)

def propeller():
    repeat 3:
        startPath()
        rightArc()
        right(90)
        rightArc()
        left(30)
        fillPath()

makeTurtle()
setFillColor("blue")
hideTurtle()

a = 0
while True:
```

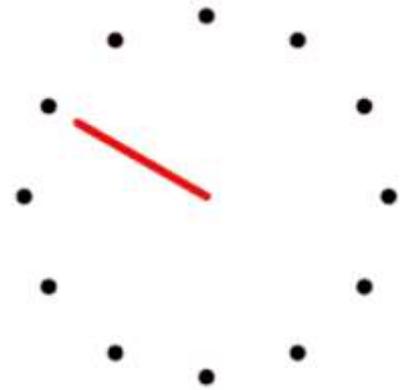
```
clear()
propeller()
delay(40)
a += 10
setHeading(a)
```

► **In Zwischenablage kopieren**

Beachte, dass du das Löschen unmittelbar vor dem neuen Zeichnen machst, damit der Bildschirm nicht längere Zeit leer ist, was zu Flackern führen kann.

Oft willst du nur in einem Teil des Turtlefensters animierte Figuren zeichnen, wobei andere Teile der Zeichnung erhalten bleiben. Falls der Hintergrund weiss ist, kannst du die Figur zum Löschen weiss übermalen.

In deinem Beispiel dreht sich der Uhrzeiger jeweils um 30 Grad nach rechts. Die schwarzen Zifferblattpunkte werden nicht gelöscht.



Programm: [► WebTigerJython](#) [► Online-Editor](#)

```
from turtle import *

def face():
    setPenColor("black")
    penUp()
    repeat 12:
        fd(110)
        dot(10)
        bk(110)
        rt(30)
    penDown()

def hand():
    setPenColor("red")
    setPenWidth(5)
    forward(90)
    back(90)

makeTurtle()
hideTurtle()
face()

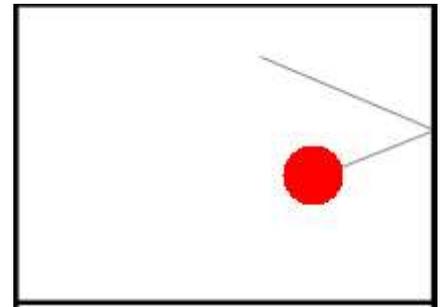
a = 0
while True:
    setHeading(a)
    setPenColor("red")
    hand()
    delay(250)
    setPenColor("white")
    dot(200)
    a += 30
```

► **In Zwischenablage kopieren**

Eine Billardkugel bewegt sich im Turtlefensters so, dass sie an den Rändern gemäss der Regel *Einfallswinkel = Ausfallswinkel* reflektiert wird. Mit *heading()* kannst du

die aktuelle Turtlerichtung zurückzuholen und sie mit `setHeading(winkel)` neu setzen.

Die Kugel wird zum Löschen weiss übermalt. Allerdings musst du dabei einen etwas grösseren Durchmesser wählen, damit auch die Peripheriepunkte verschwinden. Überlege dir, bei welchen x- und y-Koordinaten die Kugel umkehren muss, damit sie möglichst exakt am Rand reflektiert wird.



Programm: [\[► WebTigerJython\]](#) [\[► Online-Editor\]](#)

```
from gturtle import *

def billardTable():
    setPos(-250, -200)
    repeat 2:
        fd(400); rt(90)
        fd(500); rt(90)

makeTurtle()
hideTurtle()
billardTable()
setPos(-100, -100)
setHeading(60)

while True:
    setPenColor("red")
    dot(60)
    delay(20)
    setPenColor("white")
    dot(62)
    forward(5)
    if getX() > 214 or getX() < -214:
        setHeading(360 - heading())
    if getY() > 164 or getY() < -164:
        setHeading(180 - heading())
```

► **In Zwischenablage kopieren**

■ **MERKE DIR...**

Eine Animation entsteht, indem man eine Szene wiederholt in einer leicht veränderten Situation zeichnet. Dazu löscht man die alte Szene und zeichnet sie neu in der veränderten Situation... In einfachen Fällen kann man eine Figur durch Übermalen mit der Hintergrundfarbe löschen.

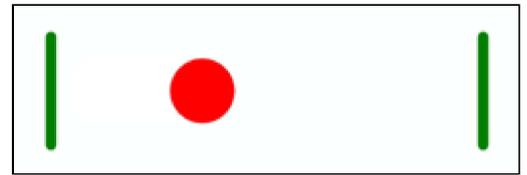
■ **ZUM SELBST LÖSEN**

1. Zeichne ein Rad und lasse es um den Punkt (0, 0) rotieren.



2. Zeichne zuerst zwei grüne Balken auf der linken und rechten Seite des Turtlefensters. Der rote

Ping-Pong-Ball soll sich zwischen den beiden Balken hin- und herbewegen.



3. Zeichne eine Fahne und schwenke sie um den untersten Punkt der Fahnenstange hinauf und hinunter.

