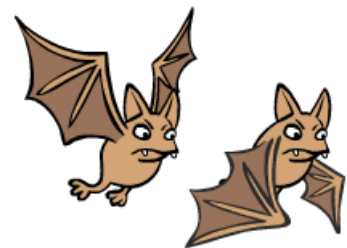


Animation

Wenn eine Figur „animiert“ ist, sieht es zum Beispiel so aus, als würde sie während des Laufens die Beine und Arme bewegen. Im Spiel sieht das flüssig aus. Flüssige Bewegung in Computerspielen ist aber eine Illusion: Sie setzt sich aus mehreren **Einzelbildern** zusammen, bei denen sich Arme und Beine jeweils ein Stück bewegen. Diese Einzelbilder werden dann schnell hintereinander abgespielt. Wenn sie schnell genug wechseln, nimmt das menschliche Gehirn sie nicht mehr als Einzelbilder, sondern als flüssigen Ablauf wahr.

Animation in Scratch

Eine einfache Animation lässt sich bereits mit zwei Kostümen realisieren, zum Beispiel mit der Fledermaus: Diese Figur hat zwei Kostüme, mit den Flügeln oben bzw. unten. Wenn man diese Kostüme schnell (aber nicht zu schnell) abwechseln lässt, sieht die Flugbewegung schon ganz passabel aus.



Es empfiehlt sich in diesem Fall **nicht**, wenn man den Block **nächstes Kostüm** einfach in die Wiederholschleife des Bewegungsskripts einbaut. Dann passiert er nämlich zu schnell (30 Mal pro Sekunde), und die Fledermaus sieht zu hektisch aus.

Nach dem Kostümwechsel sollte das Skript eine kurze Zeit warten. Das würde aber wieder die Bewegung stören, weil es sie unterbrechen würde.

Wie bei den Kollisionen können wir das Problem lösen, indem wir die Animation in einem zweiten Skript programmieren, das parallel zum Bewegungsskript läuft.

Skript für die Animation

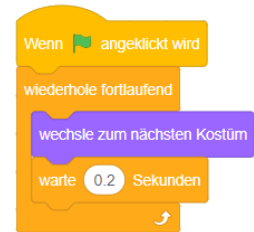
Das Animationskript enthält ebenfalls eine Wiederholschleife, die ständig das Kostüm der Fledermaus wechselt, aber nach jedem Wechsel ein bisschen wartet.

Die Dauer der Wartezeit probierst du am Besten aus – so, dass die Bewegung „natürlich“ aussieht.

Skript 1: Bewegung



Skript 2: Animation



Aufgabe 1

Bearbeite die bereitgestellte Vorlage „Aufgabe1.sb3“.

- a) Skript für die Bewegung
Die Fledermaus soll nach nach oben fliegen, wenn der Spieler die Pfeiltaste „oben“ drückt. Wenn er die Taste nicht drückt, sinkt die Fledermaus stattdessen nach unten. Bei den Tasten „links“ und „rechts“ bewegt sie sich nach links bzw. rechts.

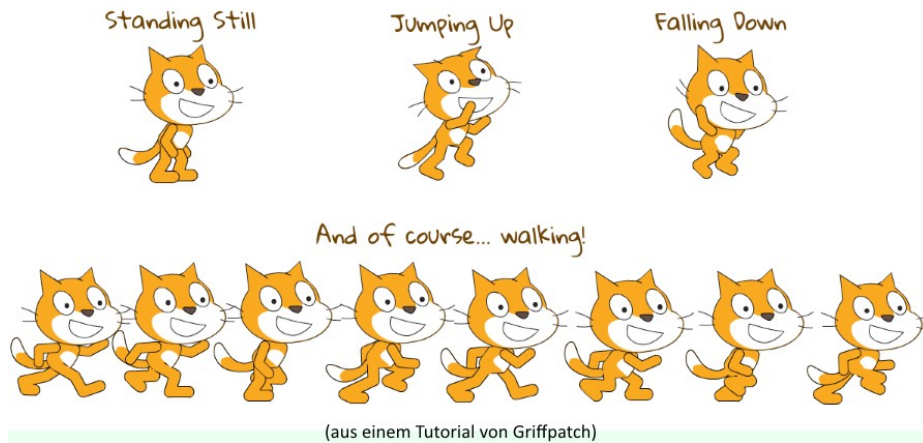
Tipp: Nutze die Blöcke **ändere x um ()** bzw. **ändere y um ()**.

- b) Skript für die Animation
Wie oben in der Abbildung: es wechselt permanent das Kostüm und wartet.

Figuren mit mehreren Animationen

Oft haben Figuren mehrere, unterschiedliche Animationen. Die Katze kann z.B. **laufen**, **stehen**, oder **springen** und hat dafür jeweils mehrere Kostüme:

- Laufen: Kostüme Nr. 1 bis 16 – viele Einzelbilder für eine flüssige Animation
- Stehen: Kostüme Nr. 17 und 18 – beim Stehen blinzelt die Katze ab und zu
- Springen: Kostüme Nr. 19 und 20 – eins für hochspringen, eins für runterfallen



Während die Katze läuft, dürfen also nur die Kostüme 1 bis 16 verwendet werden. Man kann daher nicht einfach immer weiter den Block **wechsle zum nächsten Kostüm** verwenden, denn dann würden beim Laufen auch die Kostüme für Stehen, Springen usw. gezeigt.

Mit einer Verzweigung kannst du sicherstellen, dass die Kostümnummer immer im richtigen Bereich zwischen 1 und 16 liegt: falls die Kostümnummer kleiner als 1 oder größer als 16 ist, wird einfach zum Kostüm Nr. 1 gewechselt.

```
Wenn [ ] angeklickt wird
wiederhole fortlaufend
  wechsle zum nächsten Kostüm
  falls (Kostüm Nummer < 1 oder Kostüm Nummer > 16), dann
    wechsle zu Kostüm Walk01
```

Aufgabe 2

Bearbeite die Vorlage „Aufgabe2.sb3“. Die Figur der Katze wurde hier mit vielen Kostümen erweitert (die Standard-Katze hat nur 2 Kostüme).

- a) Skript für die Bewegung
Die Katze läuft immer zwischen dem linken und rechten Bildrand hin und her (ohne Steuerung durch den Spieler – es soll hier nur die Animation geübt werden).
- b) Skript für die Animation
Die Katze wechselt zwischen den Kostümen 1 bis 16 hin und her.
Tipp: bei 16 Kostümen für das Laufen braucht es keine Wartezeit mehr im Animationskript.