

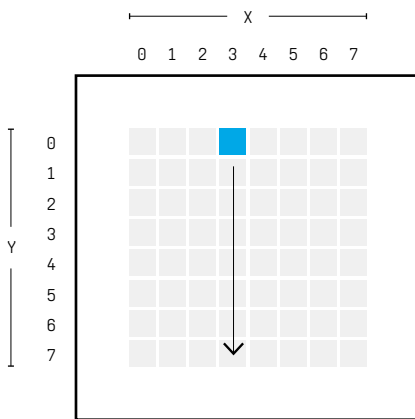
ARBEITSBLATT A008

BEDINGUNGEN

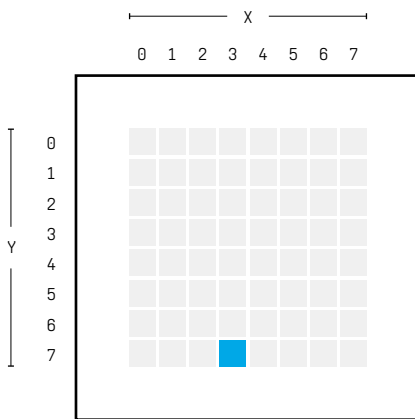
Du kannst Animationen auch mit Algorithmen programmieren und dabei viele Blöcke einsparen und tolle Effekte erzielen.

AUFGABE

Ziel dieser Aufgabe ist es, ein fallendes Pixel zu programmieren.



Zeichne das Pixel mit der Hilfe einer «y»-Variable.
Ist die Variable kleiner als sieben?
Wenn diese Bedingung erfüllt ist,
kann sie um eins erhöht werden.

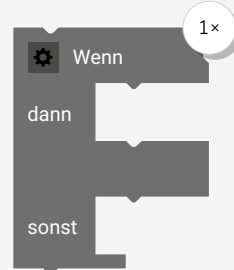
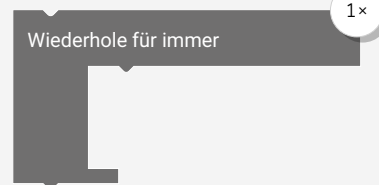


Das Pixel ist nun unten angekommen und nicht mehr kleiner als sieben. Nun muss die Variable wieder auf den Anfangswert zurückgesetzt werden, damit das Pixel wieder oben gezeichnet wird.

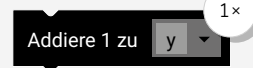
TIPP

Zeichne das Pixel nur einmal pro Schleifendurchgang. Im «Wenn/dann/sonst»-Block wird die «y»-Variable angepasst. Beachte dazu die oben stehende Illustration.

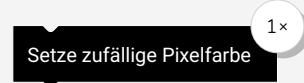
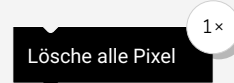
LOGIK



VARIABLEN



MATRIX



ZEIT



STÜCKLISTE LISTE DER ZU VERWENDENDEN BLÖCKE



LEVEL MITTEL

ZUSATZAUFGABE:

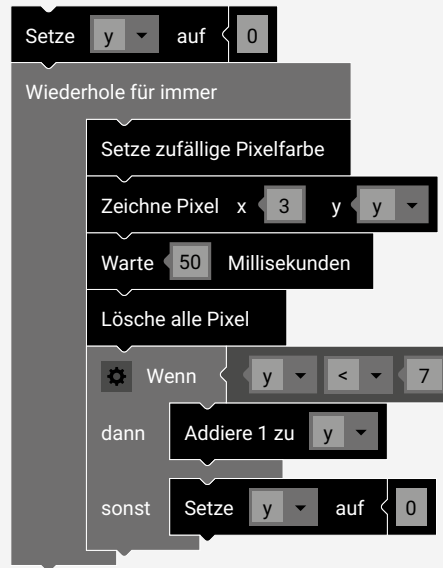
Lasse das Pixel nicht nur von oben nach unten wandern, sondern auch von links nach rechts. Erhöhe dazu nach jedem Schleifendurchlauf die x-Position des Pixels.

ARBEITSBLATT A008

BEDINGUNGEN

LÖSUNG

LÖSUNGSVARIANTE



ZUSATZAUFGABE

LÖSUNGSVARIANTE



ARBEITSBLATT A008

BEDINGUNGEN

Lernziele:

Die SuS lernen wie man eine erweiterte Bedingung stellt.

VORGEHEN

1. Zuerst wird die «y»-Variable definiert und mit null initialisiert. Dann kommt die «Wiederhole für immer»-Schleife. Alle weiteren Blöcke müssen in dieser Schleife positioniert werden.
2. Als nächstes wird eine zufällige Pixelfarbe gesetzt und das Pixel mit einem konstanten «x» und einem variablen «y» gezeichnet. Zum zeichnen gehört auch der «Warte»-Block und der «Lösche alle Pixel»-Block
3. Jetzt muss eine Bedingung gestellt werden damit überprüft werden kann ob das Pixel noch wandern soll oder zurückgesetzt werden muss.
4. Solange das Pixel noch nicht unten angekommen ist, muss die Variable um eins erhöht werden. Sobald die Zielposition erreicht oder überschritten wurde, wird die Variable auf den Anfangswert zurückgesetzt. Da bei jedem Schleifendurchgang das Pixel neu gezeichnet wird, fällt es nun Zeile für Zeile nach unten.

NEUE BEFEHLE

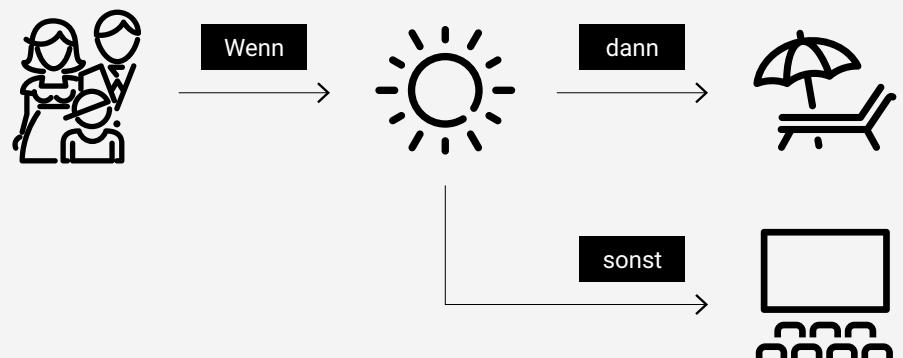


Hier kann man Variablen, Zahlen und Rechnungen mittels Vergleichszeichen einander gegenüberstellen, um eine Bedingung zu erzeugen.



In diesem Block können Bedingungen gestellt werden. Der «dann»-Block wird ausgeführt wenn die Bedingung wahr ist und sonst der «sonst»-Block.

ZUSATZINFO: BEDINGUNG/ELSE-ANWEISUNG



Hier ist wieder das Familienausflugs-Beispiel: Wenn die Sonne scheint geht sie baden, sonst geht sie ins Kino..