



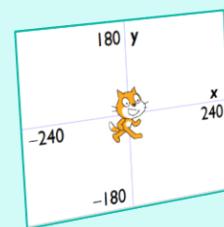
# STEUERUNG FÜR DEN AVATAR



## CHALLENGE

Eine Figur in Scratch so zu programmieren, dass sie sich mit den 4 Pfeiltasten nach oben, unten, links und rechts steuern lässt, sobald das Spiel gestartet wird!

## TIPPS



Die Lernkarte „**Koordinaten- system in Scratch**“ (15) könnte dir weiterhelfen.

## DIESE BEFEHLE HELFEN DIR DABEI



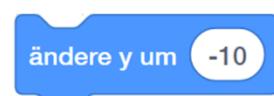
Man benötigt ein Startereignis!



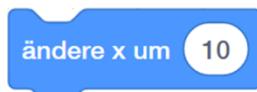
Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]



Tu etwas, falls ... (z.B. eine Taste gedrückt wird)  
[Bedingte Anweisung]



Die Position einer Figur auf der y-Achse ändern.  
[Anweisung]



Die Position einer Figur auf der x-Achse ändern.  
[Anweisung]



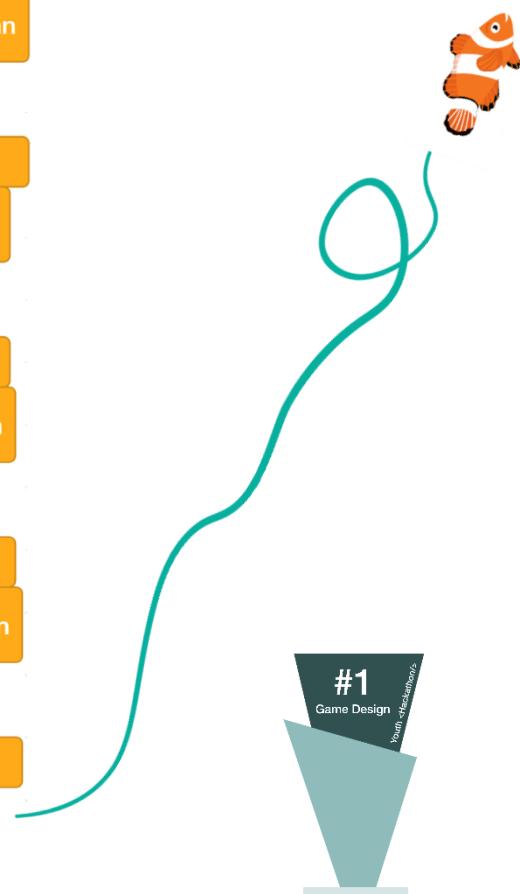
Es wird überprüft, ob eine Taste gedrückt wird.  
[Bedingung]



# STEUERUNG FÜR DEN AVATAR

## SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```
Wenn grüne Flagge angeklickt wird
wiederhole fortlaufend
  falls Taste Pfeil nach rechts gedrückt? , dann
    ändere x um 10
  falls Taste Pfeil nach links gedrückt? , dann
    ändere x um -10
  falls Taste Pfeil nach oben gedrückt? , dann
    ändere y um 10
  falls Taste Pfeil nach unten gedrückt? , dann
    ändere y um -10
ende
```

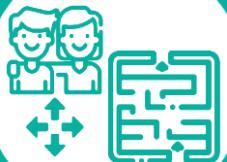


## CODING HERO-CHALLENGE

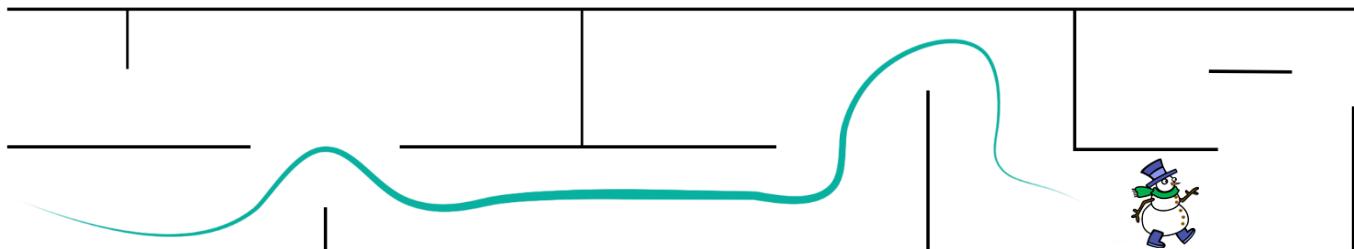
Dein Avatar ist zu langsam oder zu schnell? Wie könnte sich dein Avatar schneller oder langsamer bewegen?

Wie könnte dein Avatar immer von derselben Position starten?





# AVATAR IM LABYRINTH



## CHALLENGE

Eine Figur in Scratch so zu programmieren, dass sie sich mit den 4 Pfeiltasten steuern lässt. Durch die Labyrinthwände (oder andere Hindernisse) soll die Figur allerdings nicht durchkommen.

## TIPPS

Falls du die Steuerung – **Lernkarte „Steuerung für den Avatar“ (1)** – bereits programmiert hast, musst du sie nicht neu programmieren. Es genügt, die Steuerung um **bedingte Anweisungen** zu erweitern!

## DIESE BEFEHLE HELFEN DIR DABEI

ändere x um 10

Die Position einer Figur auf der x-Achse ändern.  
[Anweisung]



Tu etwas, falls ... (z.B. wenn eine Figur berührt wird)  
[Bedingte Anweisung]

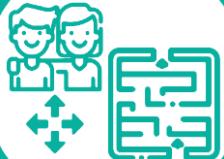
**TIPP** Gleiche Veränderung der Position auf der x bzw. y-Achse mit negativem Vorzeichen, die in der Steuerung mit positivem Vorzeichen angegeben sind und umgekehrt.

wird labyrinth berührt?

Es wird überprüft, ob eine Figur berührt wird.  
[Bedingung]

ändere y um -10

Die Position einer Figur auf der y-Achse ändern.  
[Anweisung]



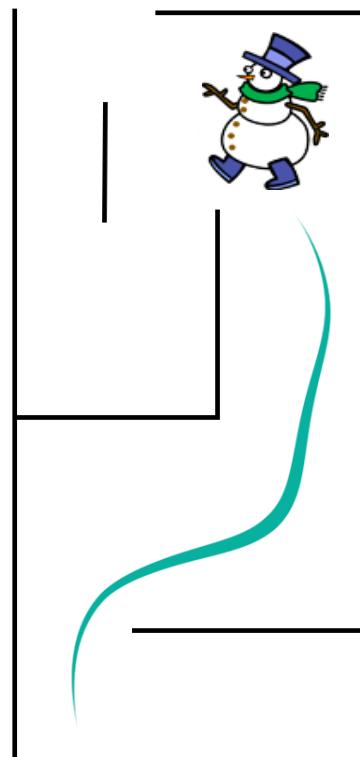
# AVATAR IM LABYRINTH

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn grüne Flagge angeklickt wird
  setze Drehtyp auf links-rechts
  wiederhole fortlaufend
    falls Taste Pfeil nach rechts gedrückt? , dann
      ändere x um 5
    falls wird Labyrinth berührt? , dann
      ändere x um -5
    falls Taste Pfeil nach links gedrückt? , dann
      ändere x um -5
    falls wird Labyrinth berührt? , dann
      ändere x um 5
    falls Taste Pfeil nach oben gedrückt? , dann
      ändere y um 5
    falls wird Labyrinth berührt? , dann
      ändere y um -5
    falls Taste Pfeil nach unten gedrückt? , dann
      ändere y um -5
    falls wird Labyrinth berührt? , dann
      ändere y um 5
  endewiederhole

```



## CODING HERO-CHALLENGE

In einem Labyrinth könnte es sinnvoll sein, dass es eine bestimmte Startposition gibt. Wie könnte dein Avatar immer von derselben Position starten?

gehe zu x: [ ] y: [ ]



# SPRINGENDER AVATAR

## CHALLENGE

Eine Figur in Scratch so zu programmieren, dass sie springt, wenn die Leertaste gedrückt wird.

## TIPPS

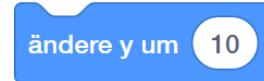
Überlege dir, welcher Wert (x oder y) zu verändern ist, dass dein Avatar nach oben springt. x? y? what? → Die Lernkarte „**Koordinatensystem in Scratch**“ (15) hilft dir gewiss weiter.



## DIESE BEFEHLE HELFEN DIR DABEI

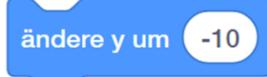


Man benötigt ein Startereignis. In diesem Fall startet das Skript, wenn eine Bedingung erfüllt wird (Leertaste gedrückt).



Die Position einer Figur auf der y-Achse um **10** ändern.  
[Anweisung]

**WUSSTEST DU SCHON?** Alle Werte, die in solchen Boxen stehen, können verändert werden!



Die Position einer Figur auf der y-Achse um **-10** ändern.  
[Anweisung]



Der Inhalt dieses Blocks wiederholt sich **10-mal**.  
[Schleife]



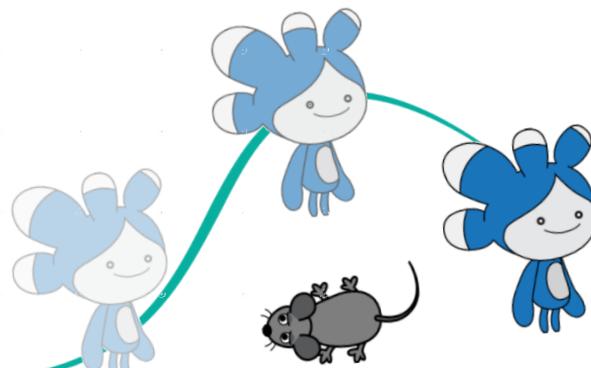
# SPRINGENDER AVATAR

## SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```
Wenn Taste Leertaste gedrückt wird
  wiederhole (10 mal)
    ändere y um (10)
  wiederhole (10 mal)
    ändere y um (-10)
```

```
wiederhole bis [y-Position < 50]
```

**ALTERNATIV** Du kannst dir auch einen solchen Wiederholungsblock basteln und es damit versuchen.



## CODING HERO-CHALLENGE

Dein Avatar sollte jetzt gerade nach oben springen können. Wie könnte sich dein Avatar dabei auch nach vorne bewegen?



Ohje! Deine Figur ist am Bildschirmrand angekommen. Wie könnte man nun das Bühnenbild wechseln und den Avatar wieder zurück an die Startposition bringen?



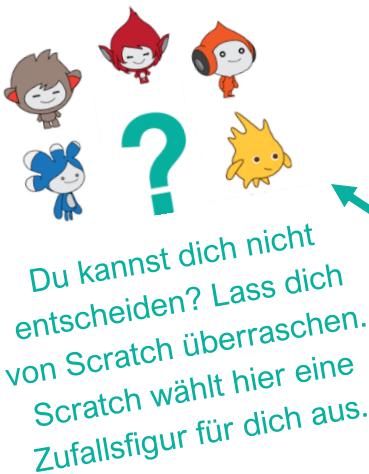


# FIGUR ERSTELLEN, VERÄNDERN, KOSTÜME WECHSELN

## CHALLENGE

Eine Figur in Scratch zu **(1) ERSTELLEN**, zu **(2) VERÄNDERN**, und/oder deren **(3) KONSTÜM ZU WECHSELN**.

### (1) FIGUR ERSTELLEN



Du kannst dich nicht entscheiden? Lass dich von Scratch überraschen. Scratch wählt hier eine Zufallsfigur für dich aus.

**VERSCHIEDENE MÖGLICHKEITEN** Scratch bietet dir verschiedene Optionen, eine Figur zu erstellen.

Überraschung

Figur wählen

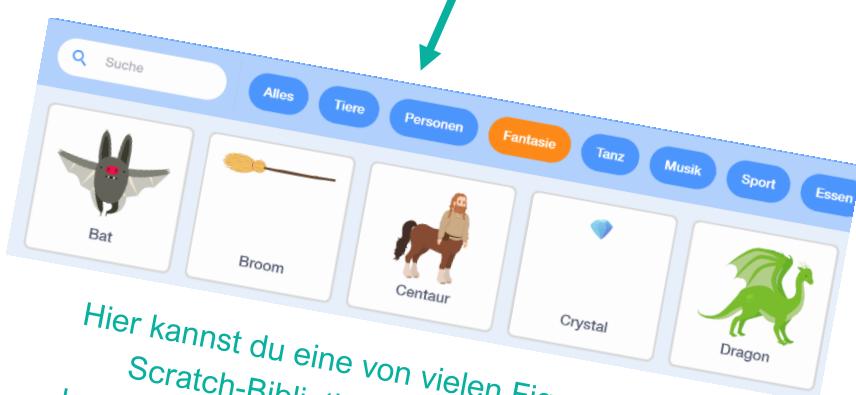


Figur hochladen

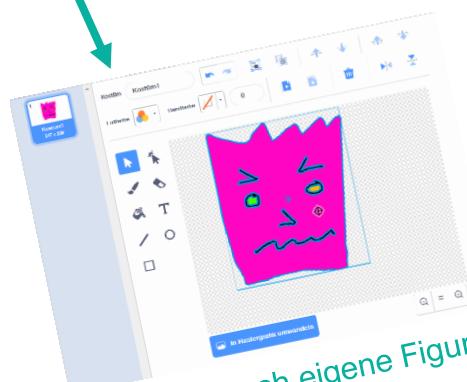
Malen



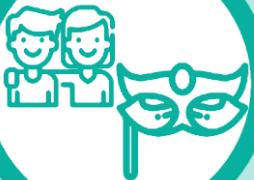
Hier kannst du eine eigene Bilddatei hochladen und als Figur verwenden. Die Bilder sollten einen transparenten Hintergrund haben (PNG-Bilddateien). Aber VORSICHT: Beachte die **Nutzungsrechte**. Bist du dir nicht sicher, ob du ein Bild verwenden darfst, benutze lieber eine eigene Figur oder eine aus Scratch!



Hier kannst du eine von vielen Figuren aus der Scratch-Bibliothek auswählen. Du wirst bemerken, dass einige Figuren bereits mehrere Kostüme haben.



Du kannst auch eigene Figuren erstellen, indem du sie malst und mit verschiedenen Farben gestaltest.

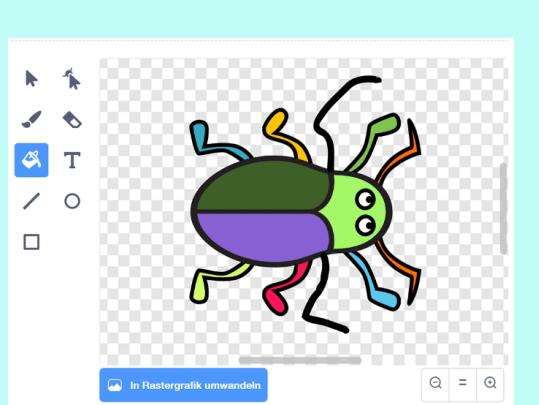


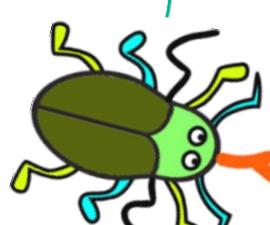
# FIGUR ERSTELLEN, VERÄNDERN, KOSTÜME WECHSELN

## (2) FIGUR VERÄNDERN

 Beetle  
Figur auswählen!

Kostüme anklicken!

 Figur bearbeiten!



## (3) KOSTÜM WECHSELN



DIESE BEFEHLE HELFEN DIR DABEI

 wechsle zum nächsten Kostüm  
Mit diesem Block wechselt man zum nächsten Kostüm einer Figur.  
[Anweisung]

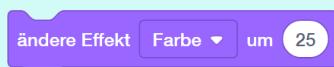
 wiederhole fortlaufend  
Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]

 warte 0.5 Sekunden  
Mit diesem Block lässt sich die Geschwindigkeit in einer Schleife regeln (z.B. wie schnell sich das Kostüm wechseln soll) oder eine Wartezeit festlegen.  
[Anweisung]

CODING HERO-CHALLENGE

Um eine Figur zu verändern, muss man nicht unbedingt ein neues Kostüm erstellen oder ein vorhandenes bearbeiten. Wie könnte man z.B. die Veränderung der Farbe einer Figur programmieren?

 setze Effekt Fischauge auf 18

 ändere Effekt Farbe um 25

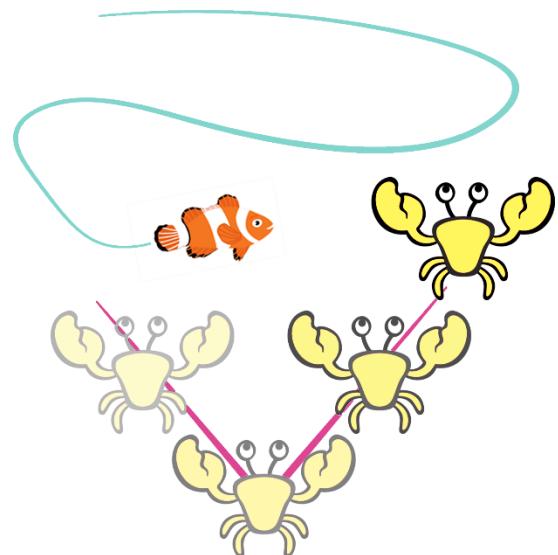


# GEGNER, DER ZUFÄLLIG STARTET UND SICH ZUFÄLLIG BEWEGT

## CHALLENGE

**SCHRITT 1** Eine Figur in Scratch so zu programmieren, dass sie zufällig erscheint und sich zufällig auf geraden Wegen bewegt, sobald das Spiel gestartet wird.

**SCHRITT 2** Sollte diese Figur (Gegner) deinen Avatar berühren, stoppt dein Programm automatisch.



## DIESE BEFEHLE HELFEN DIR DABEI

Wenn angeklickt wird

Man benötigt ein Startereignis!

stoppe alles ▾

Dein Programm soll gestoppt/beendet werden.  
[Anweisung]

pralle vom Rand ab

Eine Figur soll vom Bildschirmrand abprallen.  
[Anweisung]

falls , dann

Tu etwas, falls ... (z.B. eine Figur berührt wird)  
[Bedingte Anweisung]

gehe zu Zufallsposition ▾

Eine Figur soll zu einer Zufallsposition gehen.  
[Anweisung]

wird Freddy ▾ berührt?

Es wird überprüft, ob eine Figur berührt wird.

[Bedingung]

Zufallszahl von -179 bis 180

Generiert eine zufällige Zahl zwischen -179 und 180.  
[Algorithmus]

setze Richtung auf ▾ Grad

Die Richtung, in die eine Figur ausgerichtet ist, wird hier festgelegt. [Anweisung]

wiederhole fortlaufend

Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.

[Schleife]

**-179? 180?** Ein Kreis hat bekanntlich 360 Grad. Ein Halbkreis hat die Hälfte davon, also 180. In Scratch wird ein Kreis in zwei Halbkreise unterteilt: einen zwischen -179 und 0, einen zwischen 0 und 180.

gehe 4 er Schritt

Eine Figur soll sich für eine bestimmte Anzahl von Schritten fortbewegen. [Anweisung]



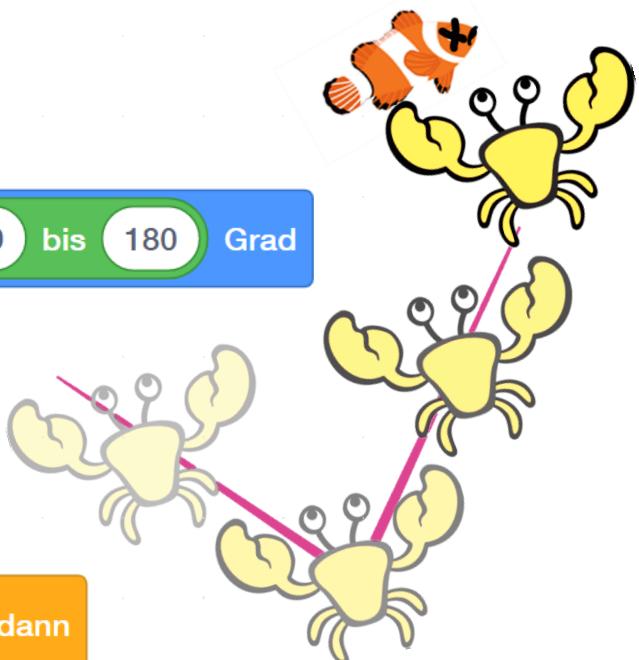
# GEGNER, DER ZUFÄLLIG STARTET UND SICH ZUFÄLLIG BEWEGT

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn grüne Flagge angeklickt wird
  gehe zu Zufallsposition
  setze Richtung auf Zufallszahl von -179 bis 180 Grad
wiederhole fortlaufend
  gehe 3 er Schritt
  pralle vom Rand ab
falls wird Freddy berührt? , dann
  stoppe alles

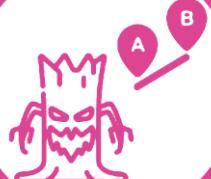
```



## CODING HERO-CHALLENGE

Deine Krabbe steht auf dem Kopf? Wie könnte man dieses Problem in den Griff bekommen?



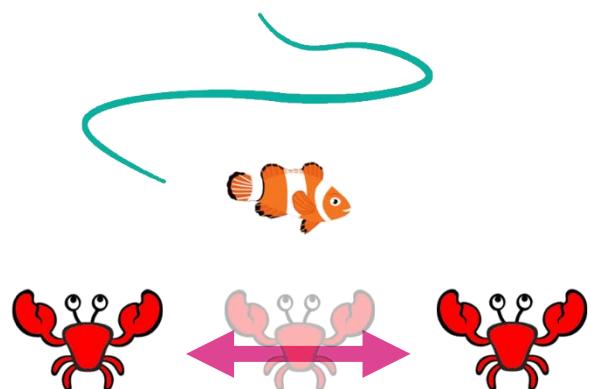


# GEGNER, DER AUF FIXEM WEG GEHT

## CHALLENGE

**SCHRITT 1** Eine Figur in Scratch so zu programmieren, dass sie sich automatisch immer am gleichen Weg bewegt, sobald das Spiel gestartet wird.

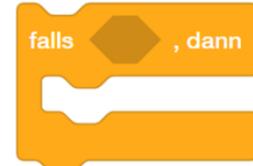
**SCHRITT 2** Sollte diese Figur (Gegner) deinen Avatar berühren, stoppt dein Programm automatisch.



## DIESE BEFEHLE HELFEN DIR DABEI



Dein Programm soll gestoppt/beendet werden.  
[Anweisung]



Tu etwas, falls ... (z.B. eine Figur berührt wird)  
[Bedingte Anweisung]



Man benötigt mehrere Startereignisse!

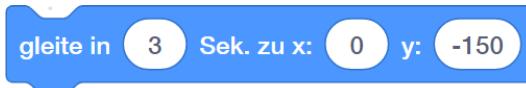


Die Position einer Figur mittels Koordinaten festlegen (z.B. Startposition).  
[Anweisung]

**ZWEI STARTEREIGNISSE – ZWEI SKRIPTE** Deine Falls-Dann-Bedingung würde mit einem Skript nicht funktionieren. Beim Programmieren gibt es nämlich eine strenge lineare Abfolge. Wenn also deine Krabbe gerade für 3 Sekunden zu einer bestimmten Position gleitet, kann gleichzeitig nicht überprüft werden, ob dein Avatar berührt wird. Du benötigst deshalb ein zweites Skript!



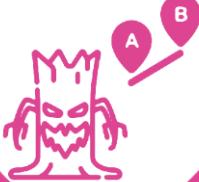
Es wird überprüft, ob eine Figur berührt wird.  
[Bedingung]



In einer bestimmten Zeit zu einer bestimmten Position am Bildschirm gleiten.  
[Anweisung]



Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]



# GEGNER, DER AUF FIXEM WEG GEHT

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

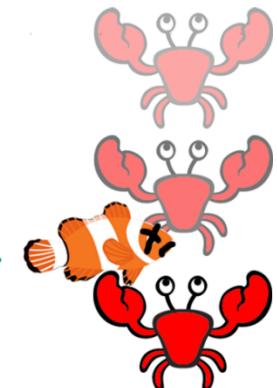
```

Wenn Flagge angeklickt wird
  gehe zu x: 0 y: -150
  wiederhole fortlaufend
    gleite in 3 Sek. zu x: 190 y: -23
    gleite in 3 Sek. zu x: 39 y: 133
    gleite in 3 Sek. zu x: -170 y: -17
    gleite in 3 Sek. zu x: 0 y: -150
  
```

```

Wenn Flagge angeklickt wird
  wiederhole fortlaufend
    falls wird Freddy berührt? , dann
      stoppe alles
  
```

→



## CODING HERO-CHALLENGE

Dein Gegner soll sich nur auf einer gewissen Achse bewegen oder sich mit zufälliger Geschwindigkeit bewegen? Wie ließe sich dies verwirklichen?

```

Zufallszahl von [ ] bis [ ]
gleite in [ ] Sek. zu x: [ ] y: [ ]
  
```

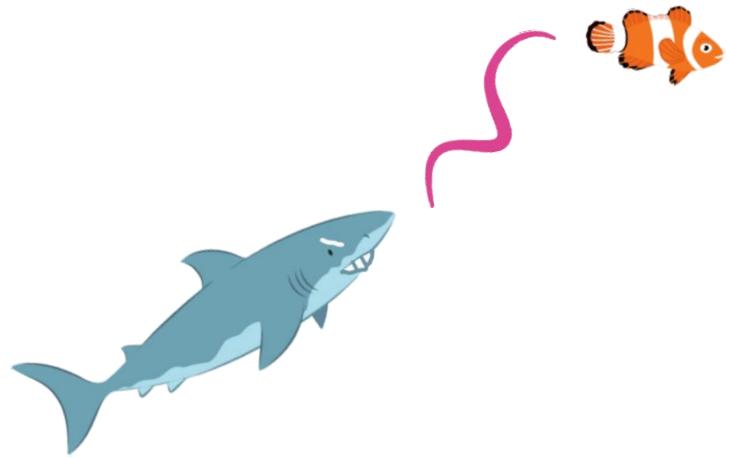


# GEGNER, DER ANDERE VERFOLGT

## CHALLENGE

**SCHRITT 1** Eine Figur in Scratch so zu programmieren, dass sie automatisch eine andere Figur verfolgt, sobald das Spiel gestartet wird.

**SCHRITT 2** Sollte diese Figur (Gegner) deinen Avatar berühren, stoppt dein Programm automatisch.



## DIESE BEFEHLE HELFEN DIR DABEI



Tu etwas, falls ... (z.B. eine Figur berührt wird)  
[Bedingte Anweisung]



Die Figur ändert seine Richtung, indem er sich zu einer anderen Figur dreht.  
[Anweisung]



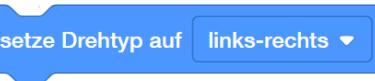
Man benötigt ein Startereignis!



Dein Programm soll gestoppt/beendet werden.  
[Anweisung]



Eine Figur soll sich für eine bestimmte Anzahl von Schritten fortbewegen.  
[Anweisung]



Die Figur soll sich nicht rundherum drehen, sondern nur seine Ausrichtung verändern(links oder rechts).  
[Anweisung]



Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]



Eine Figur soll zu einer Zufallsposition gehen.  
[Anweisung]





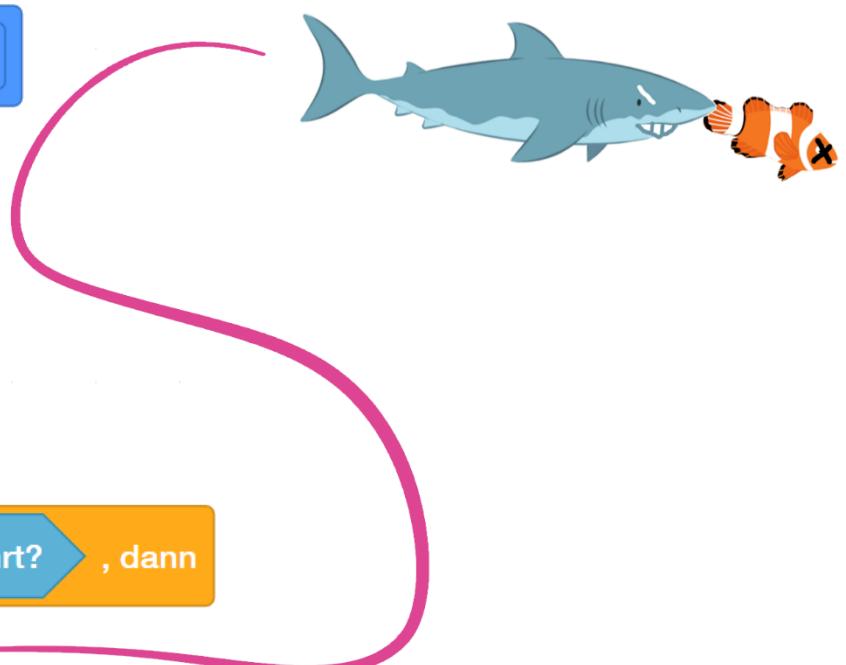
# GEGNER, DER ANDERE VERFOLGT

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn  angeklickt wird
  setze Drehtyp auf links-rechts ▾
  gehe zu Zufallsposition ▾
wiederhole fortlaufend
  drehe dich zu Freddie ▾
  gehe 2 er Schritt
  falls wird Freddie ▾ berührt? , dann
    stoppe alles ▾

```



## CODING HERO-CHALLENGE

Wie könnte dein Gegner erst ab einem gewissen Punktestand, einer gewissen Zeit oder bei einer gewissen Lautstärke erscheinen?

```
Wenn Lautstärke > 10
```

```
warte 5 Sekunden
```

```
warte bis
```

Verstecke dich

= 50

Zeige dich

Punkte:

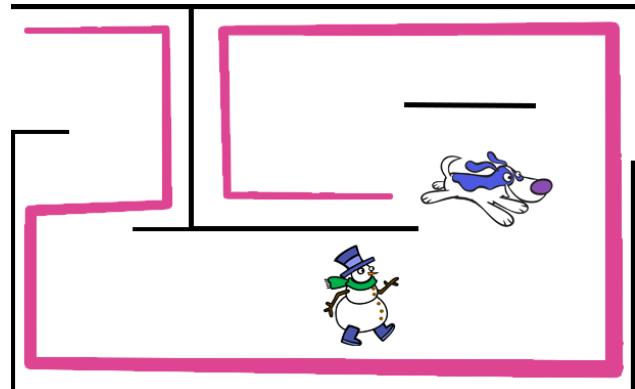


# GEGNER, DER ZUFÄLLIG IM LABYRINTH LÄUFT

## CHALLENGE

**SCHRITT 1** Eine Figur in Scratch so zu programmieren, dass sie sich auf zufälligen Wegen durch das Labyrinth bewegt und die Wände nicht überwinden kann.

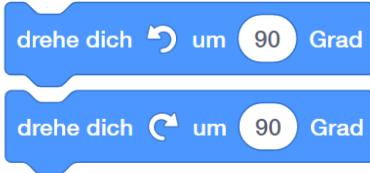
**SCHRITT 2** Sollte diese Figur (Gegner) deinen Avatar berühren, stoppt dein Programm automatisch.



## DIESE BEFEHLE HELFEN DIR DABEI



Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]



Eine Figur soll sich um eine gewisse Gradanzahl nach links oder rechts drehen. [Anweisung]



Dieser Block überprüft, ob eine z.B. Variable oder eine generierte Zufallszahl einem bestimmten Wert entspricht.  
[Mathematische Vergleichsoperation]



Eine Figur soll sich für eine bestimmte Anzahl von Schritten fortbewegen. [Anweisung]

**0 UND 1** Mit diesen Zahlen arbeitet ein moderner Computer. Das Ganze nennt sich Binärsystem, wobei die 0 und die 1 je einen Zustand (z.B. 0 = aus, 1 = ein) repräsentieren.



Die Position einer Figur mittels Koordinaten festlegen (z.B. Startposition). [Anweisung]

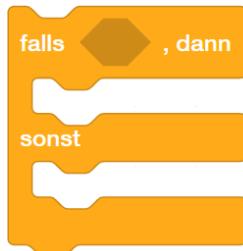


Man benötigt ein Startereignis!

Zufallszahl von 0 bis 1

Generiert eine zufällige Zahl – in diesem Beispiel 0 oder 1.

[Algorithmus]



Tu etwas, falls ... (z.B. der Zufallsgenerator 1 generiert). Wenn diese Bedingung nicht erfüllt wird, kannst du das alternative Verhalten im sonst-Teil definieren. [Bedingte Anweisung]



Es wird überprüft, ob eine Figur berührt wird. [Bedingung]

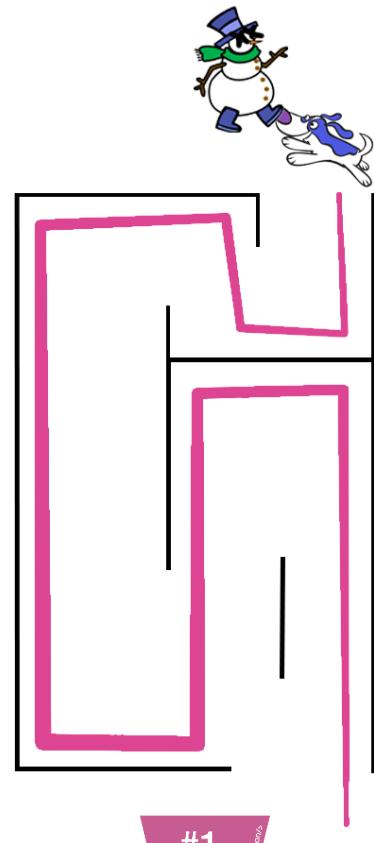
# GEGNER, DER ZUFÄLLIG IM LABYRINTH LÄUFT



SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn  angeklickt wird
  gehe zu x: -133 y: 124
  wiederhole fortlaufend
    gehe 3 er Schritt
    falls wird Labyrinth berührt? , dann
      gehe -3 er Schritt
    falls Zufallszahl von 0 bis 1 = 1 , dann
      drehe dich ⌂ um 90 Grad
    sonst
      drehe dich ⌂ um 90 Grad
  
```



#1  
Game Design  
Youth Hackathons

## CODING HERO-CHALLENGE

Du kannst die Schwierigkeit deines Spiels von Variablen abhängig machen. Solltest du auch **Sachen zum Einsammeln (9)** verwenden: Wie könnte nun dein Gegner – mit jedem von dir gesammelten Punkt – schneller und somit dein Spiel kontinuierlich schwieriger werden?



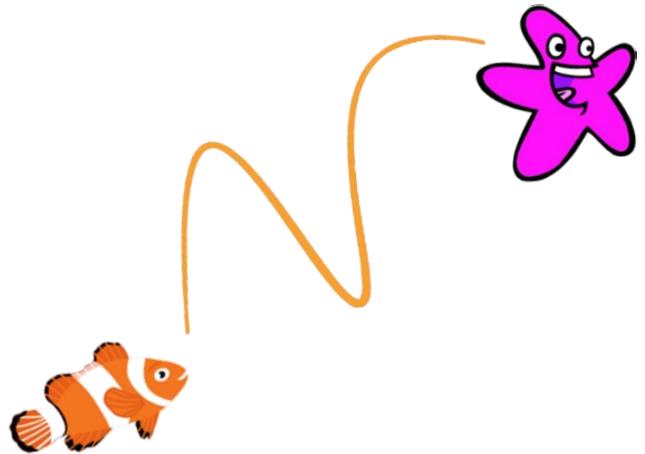


# SACHEN ZUM EINSAMMELN

## CHALLENGE

Eine Figur in Scratch so zu programmieren, dass sie zufällig an einer Position erscheint und bleibt, bis sie von einer anderen Figur berührt wird, sobald das Spiel gestartet wird.

Sobald sie berührt wurde, erscheint sie an einer anderen Stelle.



## DIESE BEFEHLE HELFEN DIR DABEI

 verstecke dich

Eine Figur soll sich verstecken bzw. vom Bildschirm verschwinden.  
[Anweisung]

 gehe zu x:  y:

Die Position einer Figur mittels Koordinaten festlegen (z.B. die Position, an der eine Figur erscheint).  
[Anweisung]

 wird  berührt?

Es wird überprüft, ob eine Figur berührt wird.  
[Bedingung]

 Wenn  angeklickt wird

Man benötigt ein Startereignis!

**-200, 150, ...?** Du erinnerst dich gewiss an das **Koordinatensystem in Scratch (15)**. Um zu gewährleisten, dass eine Figur an einer Zufallsposition innerhalb des gut sichtbaren Bildbereiches erscheint, müssen wir uns das Koordinatensystem in Erinnerung rufen. Die x-Achse umfasst einen Bereich von -240 bis 240, die y-Achse einen von -180 bis 180.

 zeige dich

Eine (versteckte) Figur soll sich zeigen.  
[Anweisung]

 Zufallszahl von  bis

 Zufallszahl von  bis

Generiert eine zufällige Zahl – hier zwischen -200 und 200 bzw. -150 und 150.  
[Algorithmus]

 warte bis

Es wird gewartet, bis eine Bedingung – z.B. die Berührung einer Figur – eintritt.  
[Bedingte Anweisung]

 wiederhole fortlaufend

Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]



# SACHEN ZUM EINSAMMELN

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

when green flag clicked
repeat (10)
    go to x: (random (-200) (200)) y: (random (-150) (150))
    show
    wait until [Freddie v. berührt?]
    hide
end
end

```

The scene shows a clownfish swimming towards a pink seastar. The seastar has a speech bubble saying "#1 Game Design Youth-Hackathon".

## CODING HERO-CHALLENGE

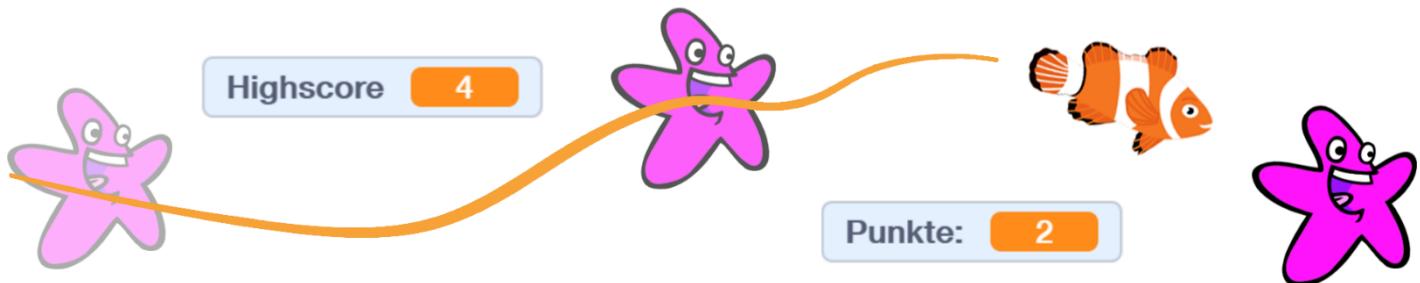
*Du kannst dein Spiel nun noch amüsanter gestalten. Wie könntest du z.B. Klänge einspielen, wenn du den Gegenstand erfolgreich eingesammelt hast?*

*Wie könntest du das Zeitintervall verändern, bis der Seestern wiederaufsteht?*





# PUNKTE SAMMELN UND HIGHSCORE



## CHALLENGE

Eine Figur in Scratch so zu programmieren, dass sie sich mit den 4 Pfeiltasten steuern lässt. Durch die Labyrinthwände (oder andere Hindernisse) soll die Figur allerdings nicht durchkommen.

## TIPPS

Falls du die Lernkarte – „**Sachen zum Einsammeln**“ (9) – bereits programmiert hast, musst du sie nicht neu programmieren. Es genügt, das bereits erstellte Skript zu erweitern!

## DIESE BEFEHLE HELFEN DIR DABEI



Der Wert einer Variable (hier: Punkte) wird um 1 erhöht.  
[Anweisung]



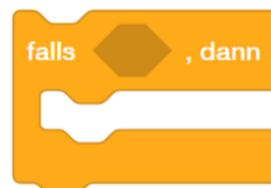
Dieser Block überprüft, ob ein Wert (z.B. jener einer Variable) größer als ein anderer Wert ist – z.B. falls Punkte > (=größer als) ein anderer Wert ist, soll sich etwas verändern.  
[Vergleichsoperation]



Der Wert einer Variable (hier: Punkte) wird auf 0 gesetzt.  
[Anweisung]



Zwei Variablen, die von dir erstellt und benannt werden müssen. Mit diesen Blöcken kann man z.B. den aktuellen Wert der Variablen abrufen.  
[individuelle Variablen]



Tu etwas, falls ... (z.B. der aktuelle Punktestand größer als dein bisheriger Rekord ist).  
[Bedingte Anweisung]



Der Wert einer Variable (hier: Highscore) wird auf einen bestimmten Wert gesetzt.  
[Anweisung]



# PUNKTE SAMMELN UND HIGHSCORE

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

when green flag clicked
set [Punkte v]
set [Highscore v]
repeat
  go to [x: (random (-200) 200) y: (random (-150) 150)]
  show [me v]
  wait until [Freddie v] touches me
  change [Punkte] by 1
  if [Punkte] > [Highscore] then
    set [Highscore] to [Punkte]
end
end
  
```

The Scratch script starts with a green flag click event. It initializes variables 'Punkte' and 'Highscore' to 0. It then enters a 'repeat' loop. Inside the loop, it moves to a random position (x: -200 to 200, y: -150 to 150), shows its sprite, and waits until the 'Freddie' sprite touches it. It then adds 1 to 'Punkte'. If 'Punkte' is greater than 'Highscore', it sets 'Highscore' to 'Punkte'. The script ends with an 'end' block.

## CODING HERO-CHALLENGE

Mittels der Punkte-Variable kannst du dein Spiel noch kreativer gestalten. Wie könntest du beispielsweise bei einer bestimmten Punkteanzahl die Farbe der einzusammelnden Figur verändern?

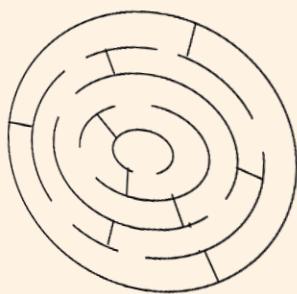
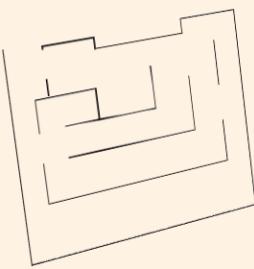




# LABYRINTH UND ÄHNLICHES ERSTELLEN

## CHALLENGE

Ein **Labyrinth oder ein ähnliches Hindernis** in Scratch so zu **erstellen**, dass es mit deinen anderen Figuren interagieren kann.



**(1)** Überlege dir, wie dein Labyrinth aussehen soll. Es gibt **unzählige Möglichkeiten** ...

**(2)** Um ein Labyrinth zu erstellen, muss man eine neue **Figur erstellen**. Verwende dazu das **Malwerkzeug**.

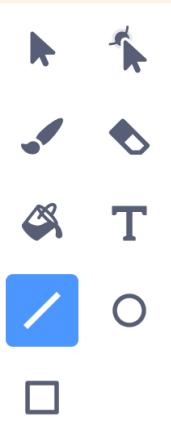
Malen



**(3)** Nun stehen dir viele Werkzeuge zur Verfügung, um ein Labyrinth zu erstellen!

### TRICK 17

Um gerade Linien zu ziehen, drücke während des Malens mit dem Linienwerkzeug **STRG + SHIFT (↑)**

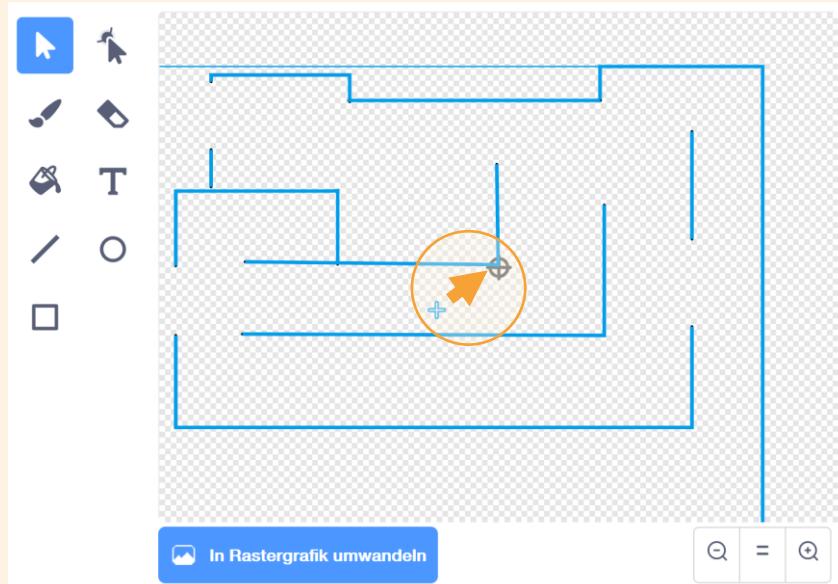




# LABYRINTH UND ÄHNLICHES ERSTELLEN

(4) Achte darauf, dass dein Labyrinth in den Figureinstellungen zentriert ist. Das **Fadenkreuz** hilft dir dabei!

Um alle Linien auszuwählen, drücke **STRG + A**.

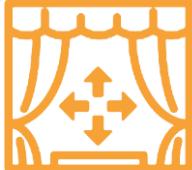


Achte ebenso darauf, dass dein Labyrinth im Bildbereich deines Programms zentriert ist:



**TIPP** Möglicherweise musst du noch die **Größe deines Avatars** anpassen, damit er auch durch die Gänge deines Labyrinths gehen kann.

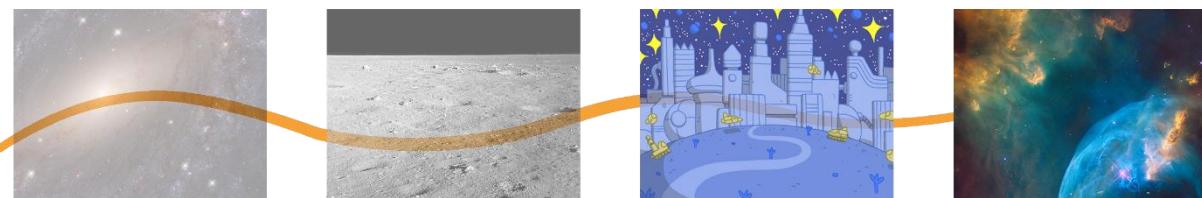




# BÜHNENBILD VERÄNDERN

## CHALLENGE

Ein Bühnenbild (Hintergrundbild in Scratch zu programmieren, dass es sich mit (1) NACH EINER FESTGELEGTEN ZEIT, (2) BEI EINER GEWISSEN PUNKTEZAHL, und/oder (3) BEI BERÜHRUNG DES RANDES VERÄNDERT.



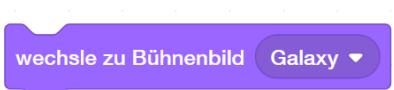
## DIESE BEFEHLE HELFEN DIR DABEI



Man benötigt ein Startereignis!



Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird. [Schleife]



Zu einem speziellen Bühnenbild wechseln.  
[Anweisung]

### (1) BESTIMMTE ZEIT



Mit diesem Block lässt sich u.a. eine Wartezeit festlegen (z.B. nach wieviel Sekunden das Bühnenbild wechselt soll). [Anweisung]

### (2) GEWISSE PUNKTEANZAHL



Tu etwas, falls ... (z.B. die Punkteanzahl einen bestimmten Wert übersteigt).  
[Bedingte Anweisung]



Dieser Block überprüft, ob ein Wert (z.B. jener einer Variable) größer als ein anderer Wert ist – z.B. falls Punkte > (=größer als) ein anderer Wert ist, soll sich etwas verändern.  
[Vergleichsoperation/Bedingung]

### (3) RAND BERÜHRT



Punkte:  
Punktevariable, welche die gesammelten Punkte speichert.  
[individuelle Variable]



Es wird überprüft, ob eine Figur den Rand berührt.  
[Bedingung]



# BÜHNENBILD VERÄNDERN

## SO KÖNNTE DEIN PROGRAMM AUSSEHEN

### (1) BESTIMMTE ZEIT

```

Wenn 🏁 angeklickt wird
warte 6 Sekunden
wechsle zu Bühnenbild Galaxy ▾
    
```

### (2) GEWISSE PUNKTEANZAHL

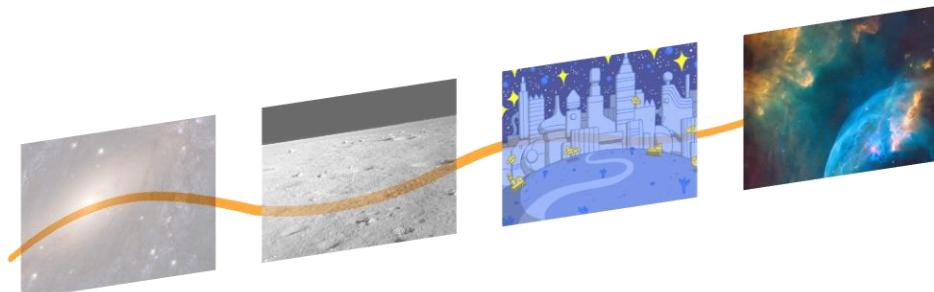
```

Wenn 🏁 angeklickt wird
wiederhole fortlaufend
falls Punkte > 5 , dann
wechsle zu Bühnenbild Galaxy ▾
    
```

### (3) RAND BERÜHRT

```

Wenn 🏁 angeklickt wird
wiederhole fortlaufend
falls wird Rand ▾ berührt? , dann
wechsle zu Bühnenbild Galaxy ▾
    
```



## CODING HERO-CHALLENGE

*Du möchtest, dass beim Spielstart immer dasselbe Bühnenbild erscheint? Wie könnte man dies programmieren?*

*Du planst an einem Jump 'n' Run-Spiel? Wie könnte man automatisch zum nächsten Bühnenbild wechseln, wenn man den Rand berührt und wie kommt man direkt darauf zurück zur Startposition?*

```

wechsle zu Bühnenbild Galaxy ▾
    
```

```

wechsle zum nächsten Bühnenbild
    
```

```

gehe zu x: ⬤ y: ⬤
    
```



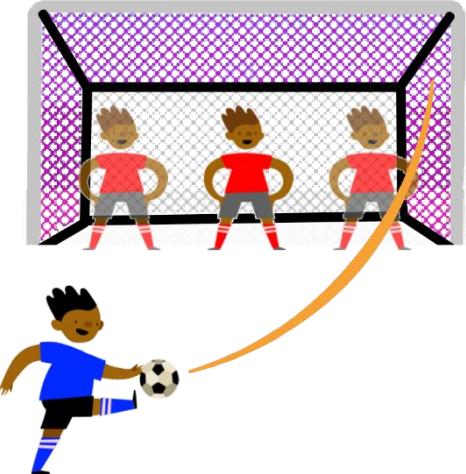
# BÄLLE WEGSCHIESSEN

## CHALLENGE

Eine Figur (z.B. einen Ball) in Scratch so zu programmieren, dass sie, wenn man eine Taste drückt, weggeschossen wird.

## TIPPS

Es gibt unzählige Möglichkeiten, wie man in Scratch einen Schuss programmieren kann. Einige hilfreiche Blöcke werden unten dargestellt, allerdings kannst du deiner Kreativität auch freien Lauf lassen.



## DIESE BEFEHLE HELFEN DIR DABEI



Tu etwas, falls ... (z.B. wenn eine Figur berührt wird) [Bedingte Anweisung]



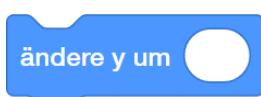
Der Wurfgegenstand geht zu einer anderen Figur (z.B. der Ball zum\*zur Fußballer\*in) [Anweisung]



Man benötigt ein Startereignis!



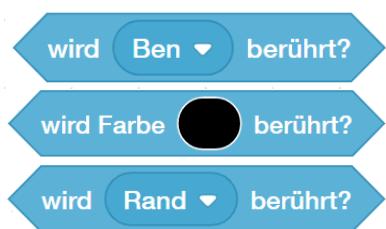
Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird. [Schleife]



Die Position einer Figur um einen bestimmten Wert auf der y-Achse ändern. [Anweisung]



Es wird überprüft, ob eine Taste gedrückt wird. [Bedingung]



Es kann z.B. überprüft werden, ob der Wurfgegenstand eine andere Figur berührt/abschießt, eine Farbe oder den Rand berührt. [Bedingungen]



Der Inhalt dieses Blocks wird so lange wiederholt, bis eine Bedingung eintritt. [Schleife mit Abbruchbedingung]



Mit diesem Block kann überprüft werden, ob zumindest eine von mehreren Bedingungen eintritt. [Vergleichsoperation/Logische Verknüpfung]



# BÄLLE WEGSCHIESSEN

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn grüne Flagge angeklickt wird
wiederhole fortlaufend
  gehe zu Karl
  falls Taste Leertaste gedrückt? , dann
    wiederhole bis wird Rand berührt? oder wird Ben berührt?
      ändere y um -10
  end
ende

```



## CODING HERO-CHALLENGE

Womöglich möchtest du deine Treffer zählen.  
Wie könnte man dies verwirklichen? Zusätzlich  
kannst du noch deinen Punktestand reduzieren,  
falls dein Wurfgegenstand das Ziel verfehlt.





# TIMER RUNTER/RAUF ZÄHLEN LASSEN

## CHALLENGE

In Scratch einen Timer/Zähler zu integrieren und programmieren, der die Zeit rauf/runter zählt.

## TIPPS

Es gibt viele Möglichkeiten, wie man in Scratch einen Timer/Zähler integrieren kann – z.B. kann man die Zeit, um ein Level zu bewältigen, vorgeben oder das Bühnenbild steuern. Die grundlegenden Blöcke für einen Timer werden unten dargestellt. Lass deiner Kreativität freien Lauf!

Zeit 0123456

Zeit 321

## DIESE BEFEHLE HELFEN DIR DABEI

Wenn angeklickt wird

Man benötigt ein Startereignis!

setze Zeit auf

Der Wert einer Variable (hier: Zeit) wird auf einen bestimmten Wert gesetzt.  
[Anweisung]

ändere Zeit um

Der Wert einer Variable (hier: Zeit) wird um einen bestimmte Wert verändert (erhöht/vermindert).  
[Anweisung]

wiederhole fortlaufend

Der Inhalt dieses Blocks wiederholt sich so lange, bis das Programm beendet wird.  
[Schleife]

warte Sekunden

Mit diesem Block lässt sich eine Wartezeit festlegen, sowie die Geschwindigkeit in einer Schleife regeln (z.B. wie schnell sich etwas ändern soll). [Anweisung]



# TIMER RUNTER/RAUF ZÄHLEN LASSEN

SO KÖNNTE DEIN PROGRAMM AUSSEHEN

```

Wenn FLAG angeklickt wird
  setze Zeit auf 10
  wiederhole fortlaufend
    warte 1 Sekunden
    ändere Zeit um -1
  
```

```

Wenn FLAG angeklickt wird
  setze Zeit auf 0
  wiederhole fortlaufend
    warte 1 Sekunden
    ändere Zeit um 1
  
```

**COUNTDOWN – RUNTER ZÄHLEN** Da unsere Variable auch einen Wert unter 0, also eine Minuszahl sein kann, müssen wir den Counter abfangen. Diese Blöcke können dafür hilfreich sein ...



## CODING HERO-CHALLENGE

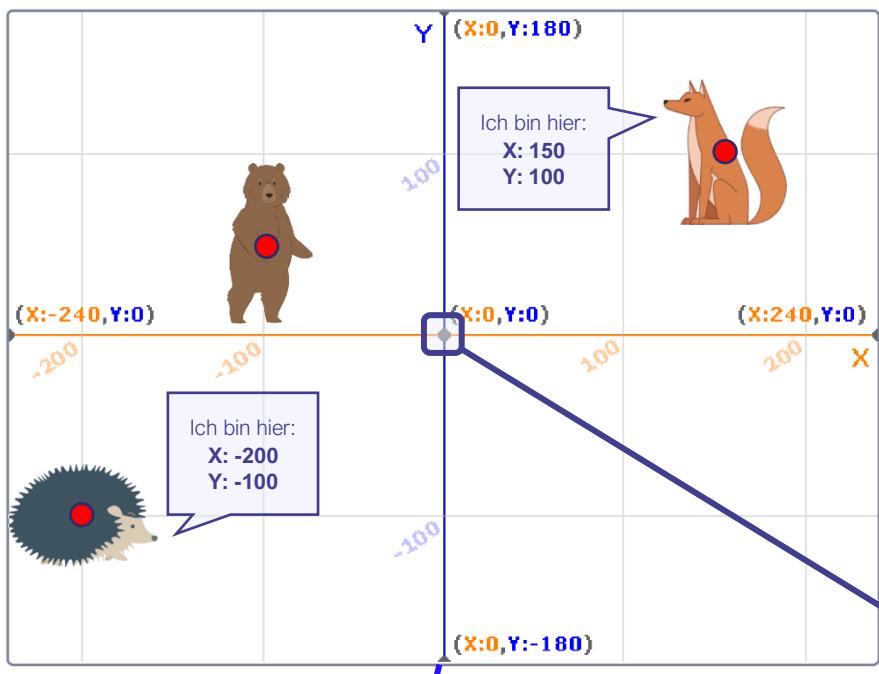
Wenn du es schaffst, ein Spiel für z.B. 10 Sekunden durchzuhalten, soll das Bühnenbild wechseln. Wie könnte man dies programmieren?



# DAS KOORDINATENSYSTEM IN SCRATCH

## ANLEITUNG

Hier erfährst du, **was** es mit dem **Koordinatensystem in Scratch** auf sich hat **und wofür** wir es bei der Programmierung benötigen!



### Y-ACHSE

Die **blaue**, senkrechte Linie bezeichnet man **y-Achse**. Sie umfasst in Scratch einen Bereich von **-180 bis 180 Pixel**.

**INFO AM RANDE** In der Mathematik wird das Koordinatensystem verwendet, um angeben zu können, wo sich ein Objekt im Raum befindet.

### X-ACHSE

Die **orangene**, waagrechte Linie bezeichnet man **x-Achse**. Sie umfasst in Scratch einen Bereich von **-240 bis 240 Pixel**.

### NULLPUNKT

Die beiden Achsen (x & y) treffen sich im sogenannten Nullpunkt. An ihm ist sowohl der **x- als auch der y-Wert 0**.

## TIPPS

Du kannst die derzeitige Position deiner Figur immer direkt unter dem Bühnenbild ablesen – auch während sie sich bewegt!



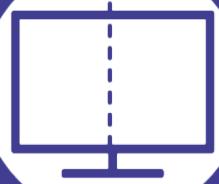
## CODING HERO-QUIZ

Wo befindet sich der Bär?

(Der **rote Punkt** markiert die Position)



- X: 200 | Y: -150
- X: -100 | Y: 50
- X: 100 | Y: -100

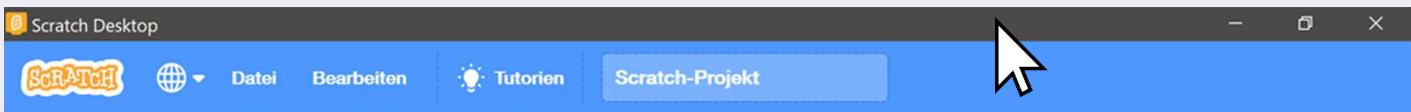


# BILDSCHIRM TEILEN

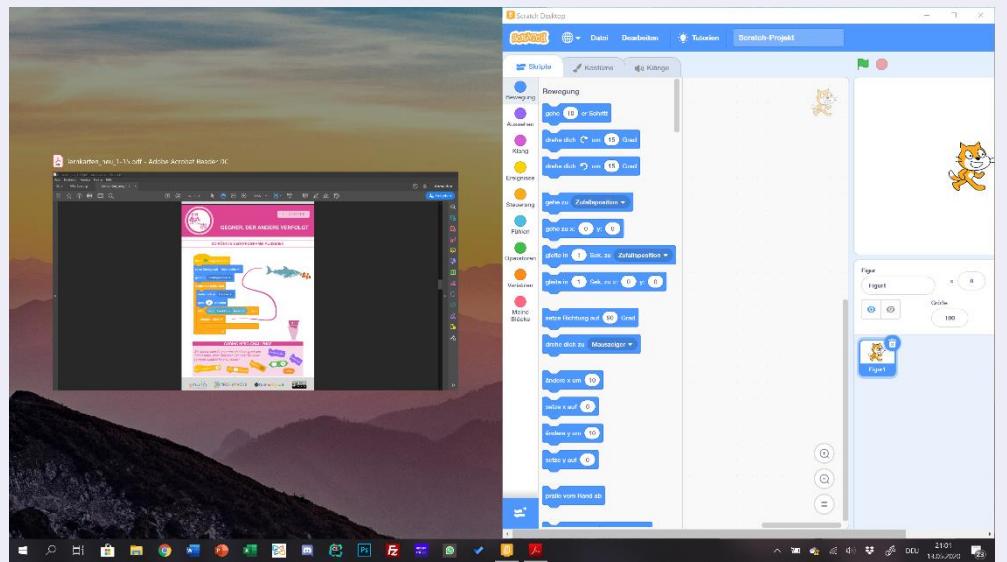
## ANLEITUNG

Hier erfährst du, wie man in Windows zwei geöffnete Fenster auf dem Bildschirm in zwei Hälften teilt (engl.: Splitscreen). Das ist zum Beispiel hilfreich, wenn du **auf einer Hälfte Scratch** platzierst, um zu programmieren **und auf der anderen Bildschirmseite eine hilfreiche Lernkarte** öffnest.

**(1)** Zuerst klickt man mit der **linken Maustaste** auf die **obere Leiste des Fensters** (Programmleiste).



**(2)** Man hält die **linke Maustaste** gedrückt, zieht den Mauszeiger **(und das Fenster)** bis **zum rechten oder linken Bildschirmrand** und lässt nun die **linke Maustaste los**. Nun sollte das Ganze so aussehen!



**(GESCHAFFT)** Womöglich wird dir bereits das Fenster der hilfreichen Lernkarte angezeigt. Falls nicht, wiederhole das Ganze noch einmal mit dem zweiten Fenster auf der gegenüberliegenden Bildschirmseite.

