

Lektion 2

In Lektion 1 hast du Scratch kennengelernt und dein erstes eigenes Programm geschrieben und gespeichert.

In dieser Lektion lernst du wichtige Dinge, die wir für unsere Spiele brauchen.

Erkunde das Koordinatensystem

Jetzt lernst du das Koordinatensystem von Scratch kennen, das wir für unsere Programme brauchen.

Bewege die Katze

- **(A)** Bewege die Katze über die Bühne und beobachte, wie sich die Zahlen der Koordinaten unter der Bühne **(B)** verändern.
- **(B)** Gib Koordinaten in die beiden Koordinatenfelder ein und beobachte, wo sich die Katze hinbewegt.
 - Versuche die Katze so in jede Ecke der Bühne zu bewegen.
 - Probiere die Katze aus der Bühne herauszubewegen. Was passiert?

Was bedeuten die Symbole? Zur Erinnerung:



Klicken: Klicke etwas an, um es auszuwählen.



Klicken und ziehen: Wenn du etwas anklickst und die Maustaste gedrückt hältst, kannst du es bewegen.

The screenshot shows the Scratch interface. In the center, a cat sprite is positioned at the origin (0,0) of a coordinate system with yellow axes. A yellow hand icon with a click symbol is labeled 'A' and is positioned over the cat. In the bottom right corner, the 'Bühne' (Stage) area contains the 'Figur' (Sprite) panel. The 'Figur' panel shows the cat sprite selected. The 'Sichtbarkeit' (Visibility) section has a checked box. The 'Größe' (Size) section shows a value of 100. The 'x' and 'y' coordinate fields are highlighted with a yellow box and labeled 'B'. The 'x' field contains the value 0 and the 'y' field contains the value 0. A yellow hand icon with a click-and-drag symbol is positioned over the 'y' field.

Das Koordinatensystem erklärt

Koordinaten? Was ist das?

- Jeder Punkt auf der Bühne hat Koordinaten: **Zwei Zahlen (x und y)**, die die **eindeutige Position** der Figur angeben.

Im Bild rechts:

- **(A)** Auf der Bühne siehst du das eingezeichnete Koordinatensystem. An den Rändern stehen die maximalen x- und y-Werte.
- **(B)** Unter der Bühne siehst du die Koordinaten der ausgewählten Figur (der Katze).

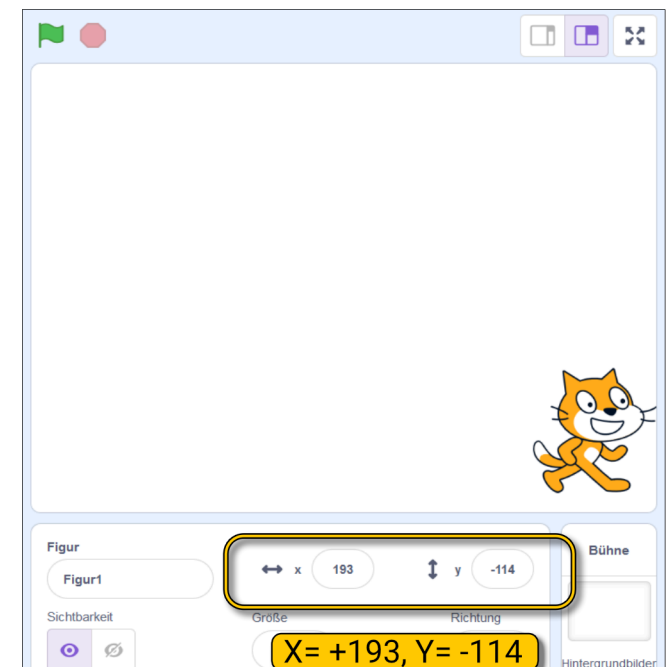
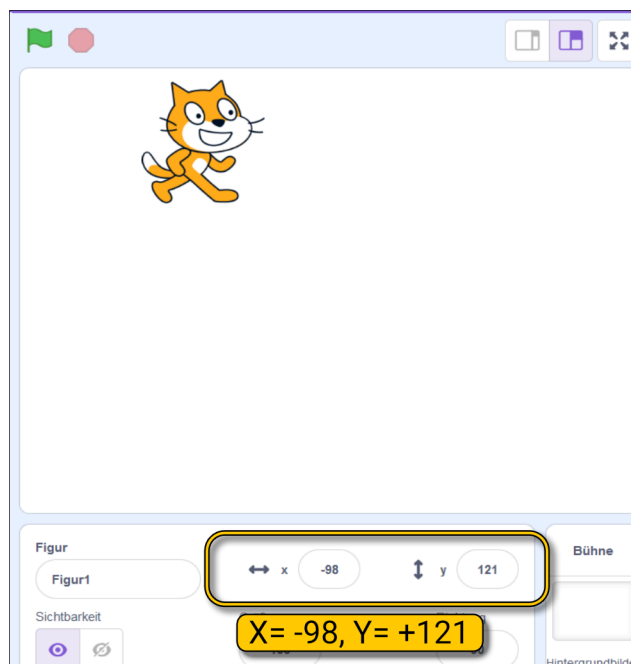
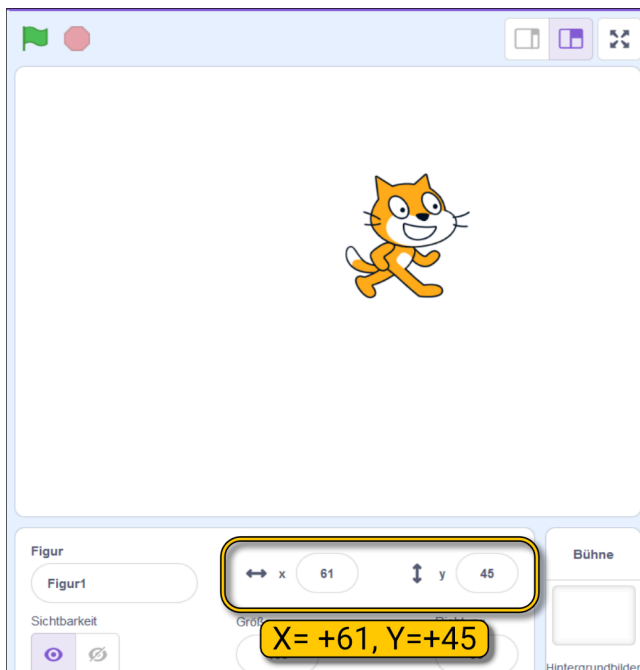
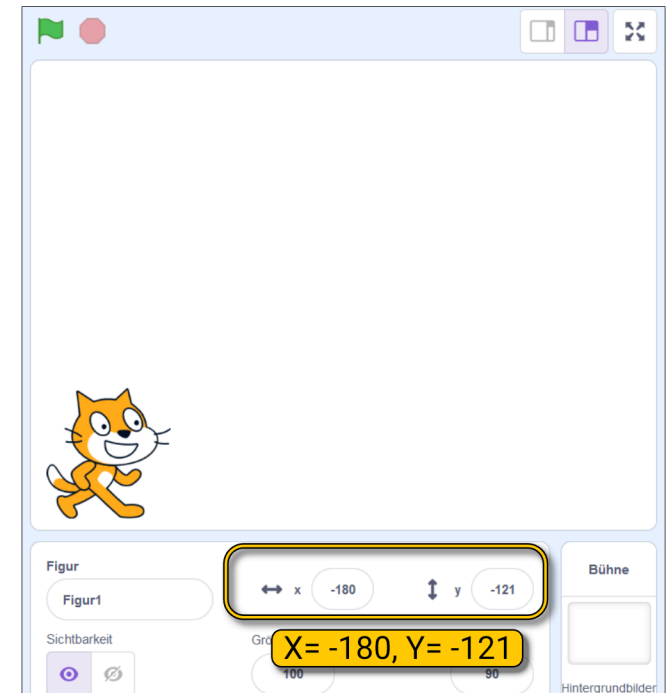
Merke: Der Nullpunkt ($x=0$ und $y=0$) liegt immer in der Mitte der Bühne, bei **(A)**.

The image shows a Scratch stage with a coordinate system overlaid. The origin (0,0) is marked with a yellow circle labeled 'A'. The x-axis ranges from -240 to +240, and the y-axis ranges from -180 to +180. A yellow cat figure is positioned in the lower-left quadrant. Below the stage, the 'Figur' (Figure) properties panel is visible, showing the selected figure 'Figur1' with its coordinates set to x: -180 and y: -121. A yellow circle labeled 'B' is placed over the y-coordinate field. The 'Größe' (Size) is set to 100 and 'Richtung' (Direction) to 90. The 'Bühne' (Stage) panel shows 'Hintergrundbilder' (Backgrounds) with '1' selected.

Beispiele

In den Bildern siehst du vier Beispielkoordinaten für die Katze.

- Achte darauf, wie sich die Werte und die Vorzeichen (+ oder -) verändern, je nachdem, wo die Figur auf der Bühne steht.



Aufgabe: Die Katze will in die Ecken

Erstelle ein Skript, welches folgendes macht:

Wenn die **grüne Fahne geklickt** wurde, **gleitet** die Katze in **alle vier Ecken** und danach wieder zurück zum **Nullpunkt** ($x=0$ und $y=0$).

Nutze dafür diese beiden Blöcke (wenn nötig mehrmals):




=> Die **Lösung** findest du auf der nächsten Seite.

Lösung: Die Katze will in die Ecken

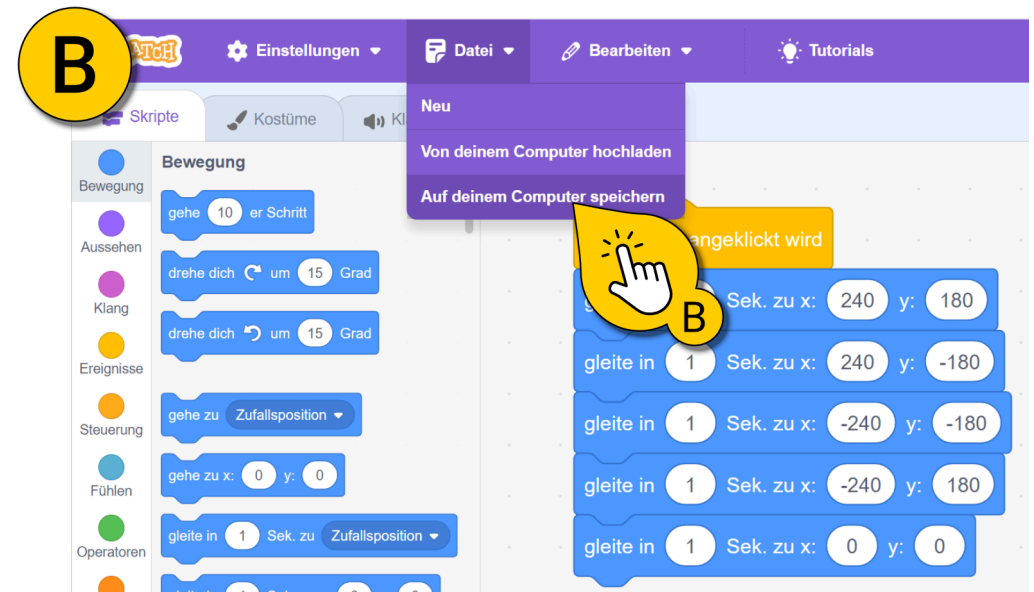
Hat dein Skript funktioniert?

(A) Im Bild rechts siehst du die Lösung.

Baue die Lösung nach, wenn dein Skript anders aussieht.

 (B) Wenn das Skript funktioniert, speichere es auf deinem Computer. Wir brauchen es später nochmal. Nenne die Datei "Katze in Ecken".

Wichtig: Merke dir, wo du deine Dateien speicherst, damit du sie später wiederfindest.



Die Katze springt in die Ecken

Lass uns jetzt ein Skript erstellen das die Katze in die Ecken springen lässt.

Wenn du dein Projekt gespeichert hast, öffne ein neues Projekt oder lösche alle vorhandenen Blöcke.

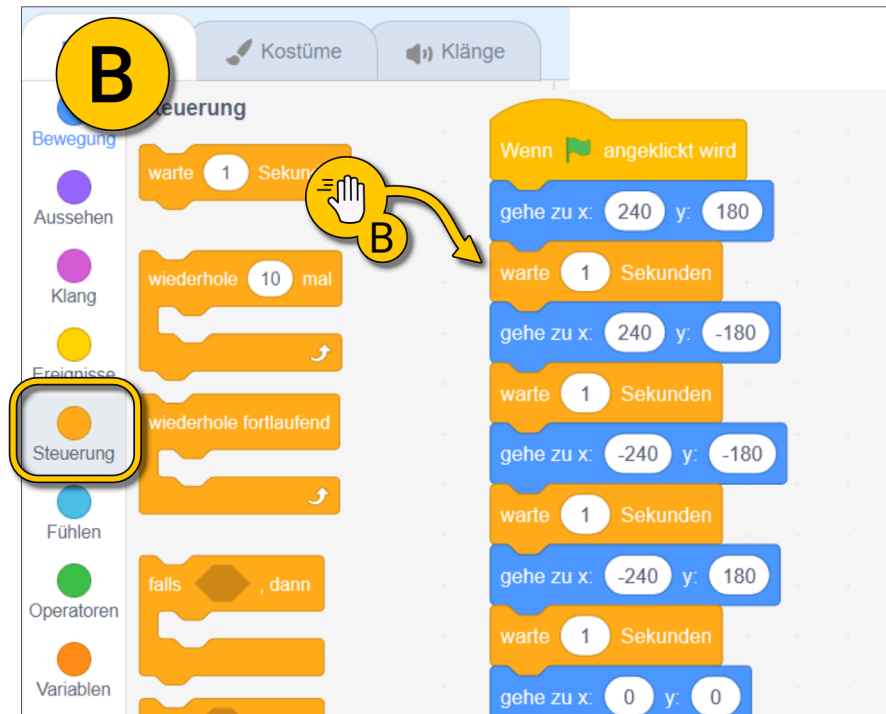
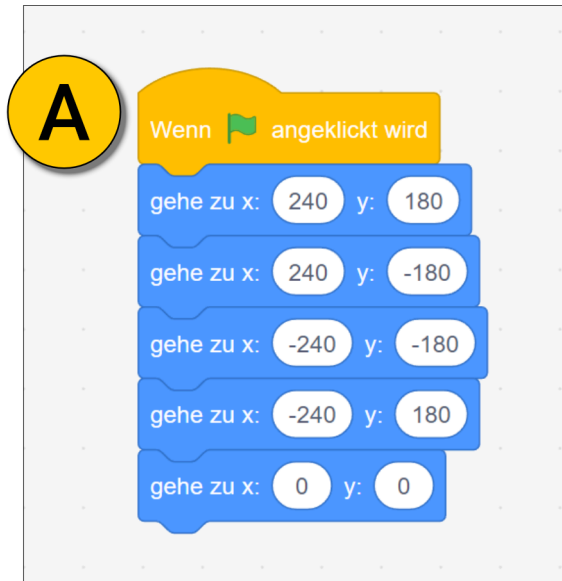
Wir benutzen dann den Block "gehe zu ..." (**Bewegung**). Baue das Script bei (A) nach und probier es aus.

Funktioniert es? NEIN? Überlege, woran das liegen könnte.

Das liegt daran, dass der Computer die fünf Befehle so schnell nacheinander ausführt, dass wir die einzelnen Schritte nicht sehen.

(B) Der "warte..." Block (bei **Steuerung**) ist hier die Lösung. Nun bleibt die Katze für 1 Sekunde in jeder Ecke. Probier es aus!

Merke: Skripte können so schnell ablaufen, dass wir gar keine Veränderung sehen. Abhilfe schafft hier der "warte ...Sekunden" Block".



Quiz:

1. Die Katze soll sich in die **rechte untere Ecke** bewegen. Welche Koordinaten musst du benutzen? (eine richtige Antwort)

- $x= 240$ und $y= -180$
- $x= 240$ und $y= 180$
- $x= 0$ und $y= -240$

2. Um die Katze nacheinander in alle Ecken zu bewegen, kannst du die Blöcke **“gleite zu ...”** oder **“gehe zu...”** nutzen. Welche Aussagen sind richtig? (mehrere korrekte Antworten)

- “gehe zu...” passiert so schnell, dass es so wirkt als würde die Katze sich gar nicht bewegen. “gleite zu...” bewegt die Katze langsam von Ecke zu Ecke.
- Da “gleite zu ...” die Katze langsam bewegt muss man höhere Werte für x und y eingeben als bei “gehe zu...”
- Nach “gehe zu ...” sollte man den Block “warte ...” nutzen.
- Beide Blöcke machen das Gleiche und es gibt keinen Unterschied.

3. Angenommen, die Katze ist verschwunden und du kannst sie nicht anklicken und nicht bewegen. Wie kannst du sie trotzdem sicher zum Nullpunkt der Bühne (die Mitte) bewegen? (mehrere richtige Antworten)

- Gar nicht. Aber ich kann eine neue Katze hinzufügen.
- Ich nutze dafür die beiden Koordinatenfelder unter der Bühne.
- Ich schreibe mir ein Skript, das bei “grüne Fahne geklickt” die Katze zurück zum Nullpunkt bringt.

Schreibe deine Antworten in das Forum des Kurses.

Wiederholen-Schleifen I

Jetzt lernst du einen sehr nützlichen Block für deine Programme kennen: "wiederhole..." Schleifen .



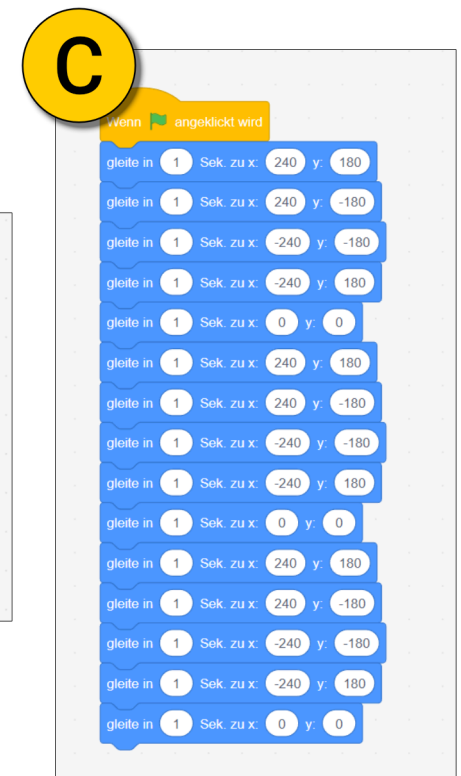
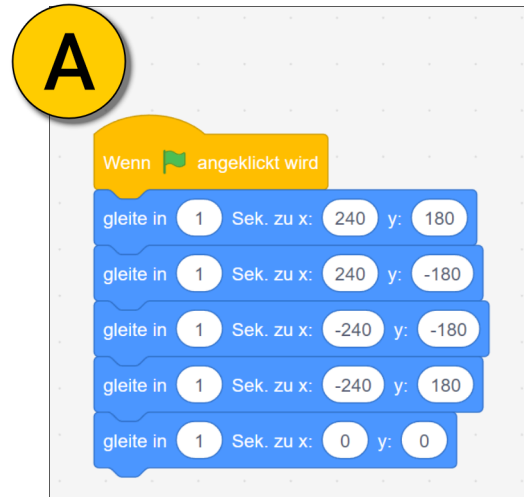
Als Beispiel nutzen wir unser Skript, das die Katze in die Ecken gleiten lässt. Du hast es gespeichert unter dem Namen "**Katze in Ecken**". Lade das Projekt über **Datei > hochladen** oder erstelle es neu.

(A) Rechts im Bild siehst du das Skript, das wir jetzt brauchen.

Wir wollen die Katze mehrmals in alle Ecken bewegen.

(B) Mit einem **Rechtsklick** auf den obersten blauen Block kannst du die gesamte Sequenz duplizieren und unten noch einmal anfügen.

(C) Wir wollen die Katze insgesamt **dreimal** in alle Ecken und zurück zum Nullpunkt bewegen.



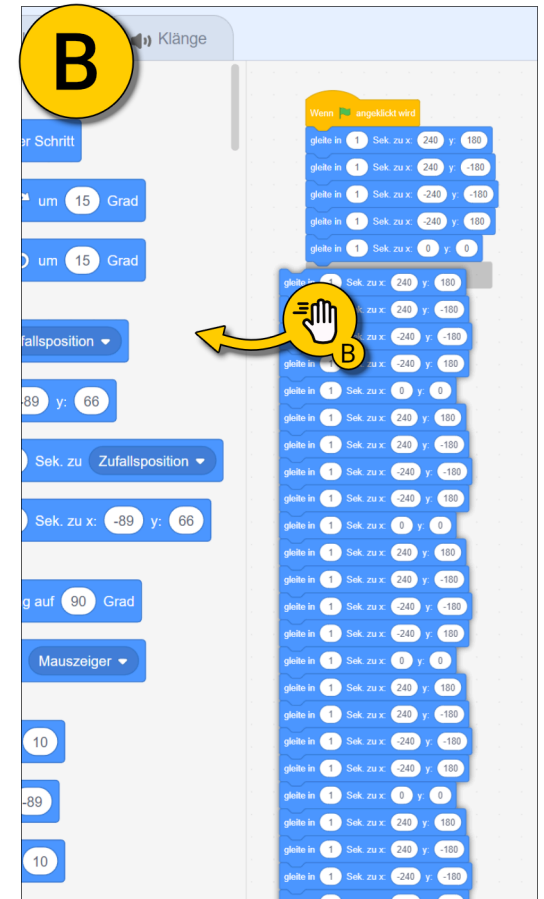
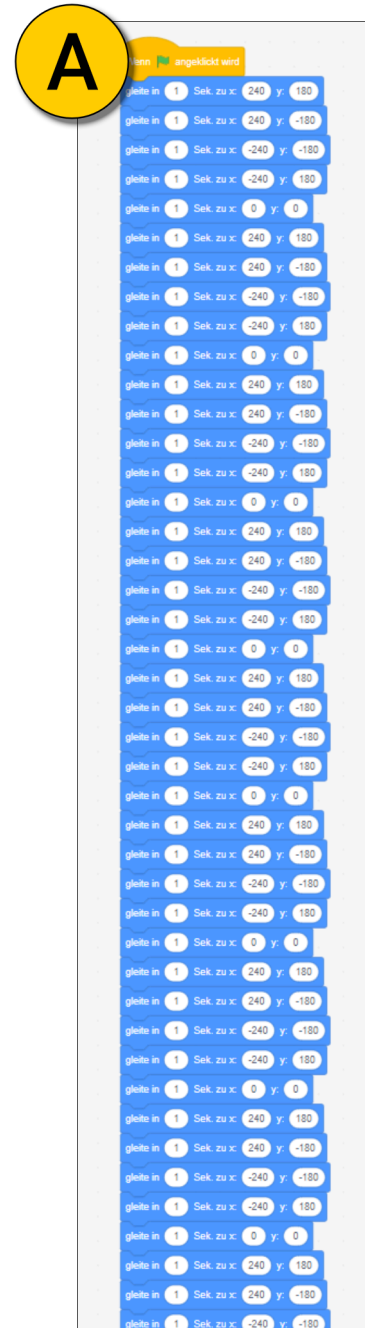
Teste dein Skript. Funktioniert es?

Wiederholen-Schleifen II

(A) Rechts siehst du ein Skript, das die Katze 10-mal in alle Ecken bewegt. Bau es nach - oder halt, warte! Geht das nicht einfacher? Ja, das geht einfacher. Und zwar mit Schleifen.

(B) Lass uns dafür unser Skript nochmal vereinfachen. **Lösche alle blauen Blöcke, bis auf die obersten 5.** Klicke dafür auf den 6. blauen Block und ziehe die ausgewählten Blöcke zurück in die Blockauswahl (B).

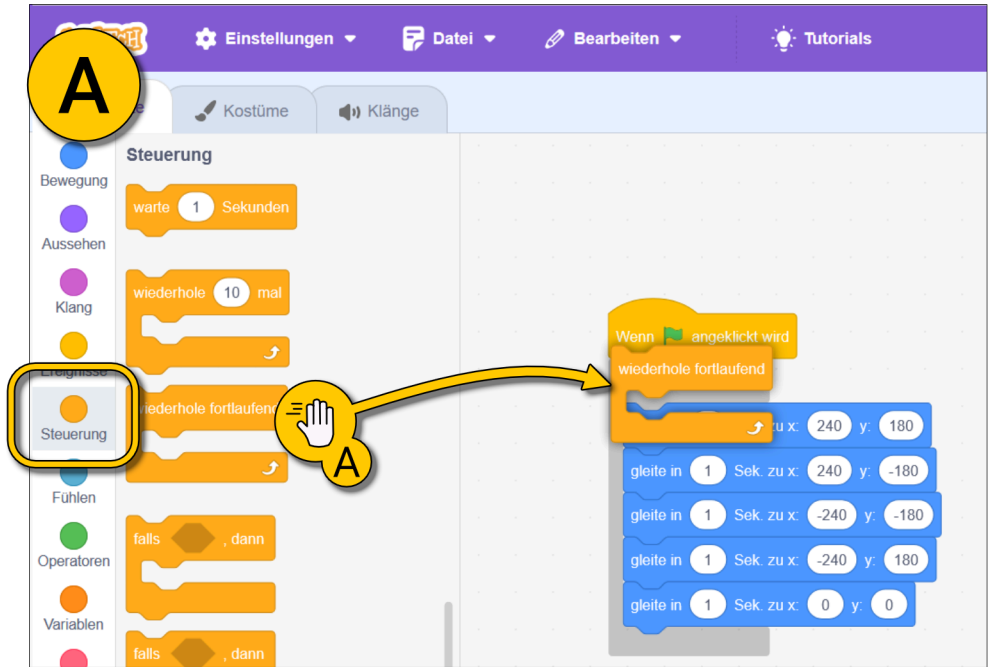
Gehe dann zur nächsten Aufgabe.




Wiederholen-Schleifen III

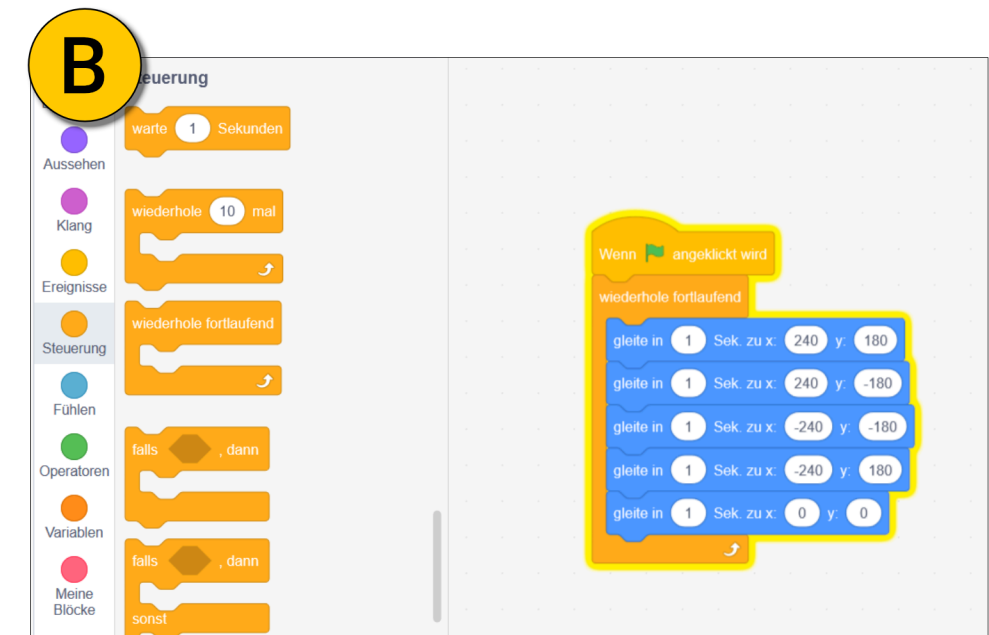
Jetzt zeige ich dir wie du dein Programm mit einer "wiederhole..." Schleife vereinfachen kannst.

(A) Ziehe eine "wiederhole fortlaufend"-Schleife (Steuerung) unter den "wenn grüne Fahne angeklickt wird" Block. Wenn alles richtig zusammengefügt ist, sollte das Skript wie bei (B) aussehen. Eventuell musst du die Blöcke noch etwas hin- und herbewegen bis alles passt.



 Probiere das Skript aus.

=> Die Katze sollte sich immer wieder in alle 4 Ecken und dann zum Nullpunkt bewegen, bzw. dahingleiten.



Wiederhole x-mal?

Dir ist vielleicht aufgefallen, dass wir gar nicht bestimmen können wie oft sich die Katze in die Ecken bewegt. Dafür ist der **“wiederhole ... mal”** Block.

Ersetze den “wiederhole fortlaufend” Block mit dem “wiederhole 10 mal” Block. Die 10 kannst du ändern wenn du möchtest.

Das coole an dem **“wiederhole ... mal”** Block ist, dass man danach noch weitere Blöcke anfügen kann.



Probiere das Skript aus!



```
Wenn [ ] angeklickt wird
  wiederhole 10 mal
    gleite in 1 Sek. zu x: 240 y: 180
    gleite in 1 Sek. zu x: 240 y: -180
    gleite in 1 Sek. zu x: -240 y: -180
    gleite in 1 Sek. zu x: -240 y: 180
    gleite in 1 Sek. zu x: 0 y: 0
```

Aufgabe: Schleifen und normale Blöcke kombinieren

Lass uns nun das Skript etwas verändern. Ändere es so, dass es Folgendes macht:

Wenn die grüne Fahne geklickt wurde, gleitet die Katze **dreimal** in alle vier Ecken ohne den Nullpunkt zu besuchen. **Erst danach gleitet die Katze zum Nullpunkt und bleibt dort.**

=> Die Lösung findest du auf der nächsten Seite.

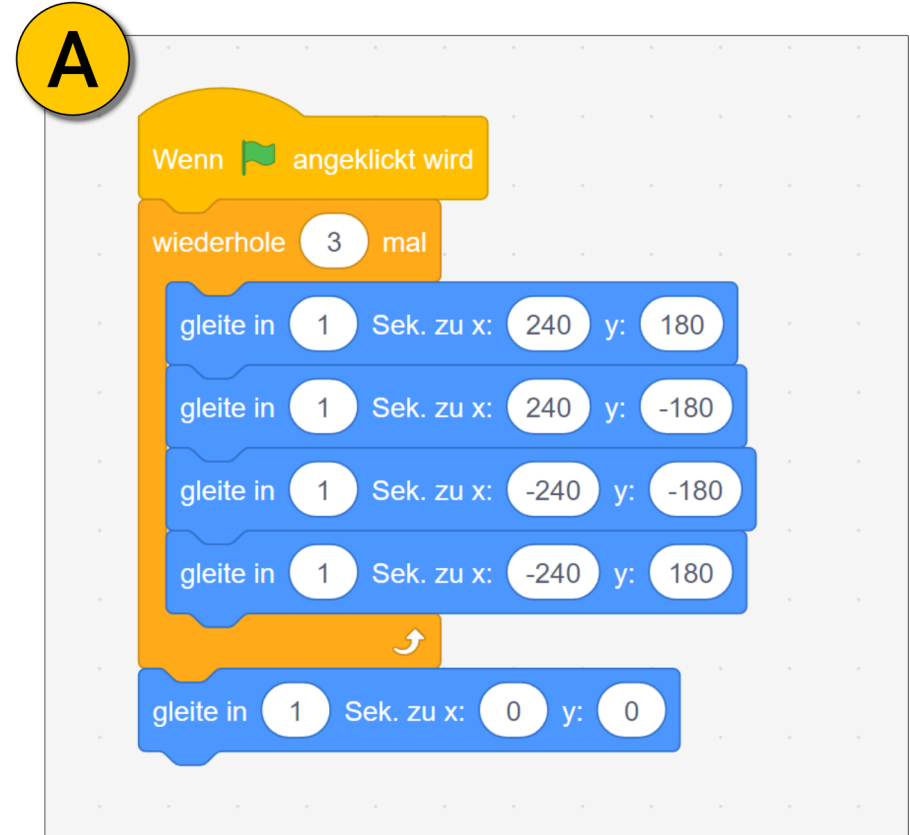
Lösung: Schleifen und Blöcke kombinieren

Hat dein Skript funktioniert?

(A) Im Bild rechts siehst du die richtige Lösung.

Teste diese Lösung, falls deine anders aussieht.

Merke: Schleifen sind nützlich, um Teile eines Skripts zu wiederholen. Sobald der letzte Block innerhalb der **“wiederhole ...”**-Klammer ausgeführt wurde springt die Schleife wieder an den Anfang.

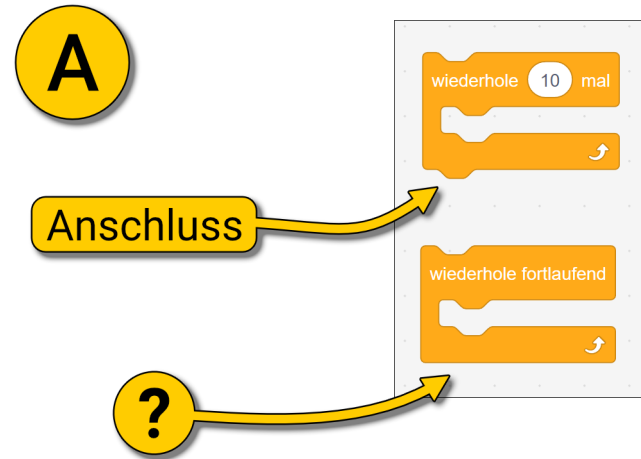


Denksport - Schleifen:

(A) Rechts siehst du die beiden Schleifen „wiederhole ... mal“ und „wiederhole fortlaufend“. Schau dir die markierten Stellen der Blöcke an. Der „wiederhole fortlaufend“ Block hat **keine Anschlussmöglichkeit** für weitere Blöcke.

Welche Aussage ist richtig?

- Der „wiederhole fortlaufend“ Block wird unendlich oft den Inhalt seiner Klammer ausführen. Ein Block nach der Klammer würde also niemals ausgeführt werden.
- Der „wiederhole fortlaufend“ Block benötigt kein Event um ihn auszulösen. Deshalb fehlt der Anschluss für weitere Blöcke.



(B) Rechts siehst du ein Skript für die Katze. **Was bewirkt das Skript?**

- Die Katze springt dreimal zwischen $x=0$ und $x=100$ hin und her. Auf jeder Position wartet sie 1 Sekunde. Danach dreht sie sich endlos im Kreis.
- Die Katze gleitet innerhalb von 1 Sekunde zu $x=100$ und anschließend zurück zum Nullpunkt. Nachdem sie das dreimal gemacht hat dreht sie sich einmal um 15 Grad.

Schreibe deine Lösung ins Forum.

