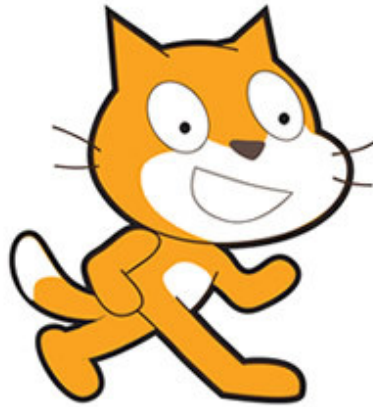


# Devovx4Kids - Scratch Workshop

...ganz viele Freiwillige ...

11. Februar 2016

# SCRATCH



# Inhaltsverzeichnis

<b>1</b>	<b>Wähle ein Sprite aus und bewege es in 4 Richtungen</b>	<b>2</b>
1.1	Aufgabe . . . . .	2
1.2	Die Scratch Oberfläche . . . . .	2
1.2.1	Wähle einen Sprite . . . . .	2
1.2.2	Bewege deinen Sprite in 4 Richtungen (Rechts, Links, Hoch, Runter) . . . . .	5
<b>2</b>	<b>Spüre die Welt – Bedingte Anweisung und Verzweigung</b>	<b>12</b>
2.1	Mal ein Labyrinth . . . . .	12
2.2	Abprallen vom Hinderniss . . . . .	13
<b>3</b>	<b>Etwas zu Essen. Bedingungen, Variablen, Ausblenden, and Sound.</b>	<b>17</b>
3.1	Erzeuge das Essen! . . . . .	17
3.2	Erzeuge das Skript für das Ergebnis und das Ausblenden . . .	18
3.3	Füge einen Klang hinzu, wenn der Ball deinen Sprite berührt .	20
3.4	Kopiere Objekte . . . . .	22
3.5	Füge Hintergrundmusik hinzu . . . . .	22
<b>4</b>	<b>Führe die Katze</b>	<b>23</b>
4.1	Male eine Rennbahn . . . . .	23
4.2	Katze einfügen . . . . .	25
4.3	Erzeuge das Script für die Steuerung der Katze . . . . .	26
4.4	Zusatzaufgabe: Erweitere das Spiel damit zwei Spieler spielen können . . . . .	29
<b>5</b>	<b>Werde zum Tennisspieler</b>	<b>31</b>
5.1	Programmiere einen Spieler . . . . .	31
5.1.1	Male den Spieler . . . . .	31
5.1.2	Programmiere das Verhalten des Spielers . . . . .	33
5.1.3	nun der zweite Spieler . . . . .	34
5.2	Der Ball kommt ins Spiel . . . . .	35
5.2.1	Male einen Ball . . . . .	35
5.2.2	Der Ball fliegt . . . . .	36
5.3	Der Hintergrund . . . . .	38
5.4	Wer hat gewonnen? . . . . .	40
5.5	Erweiterungen . . . . .	44

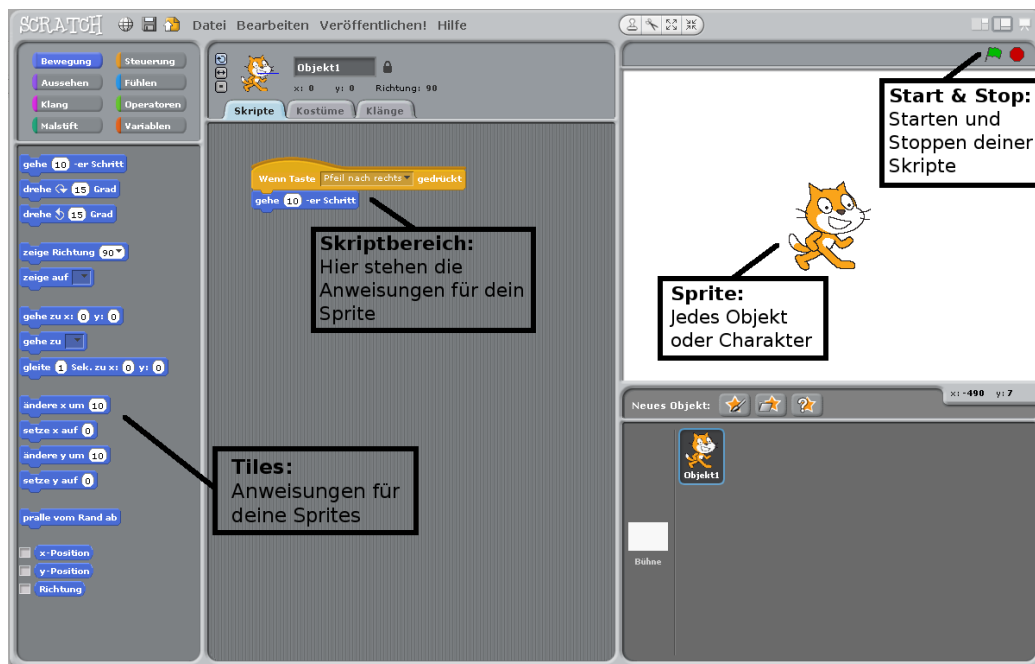
# 1 Wähle ein Sprite aus und bewege es in 4 Richtungen

## 1.1 Aufgabe

Das Scratch Programm wurde am MIT entwickelt, um jungen Schülern die Programmierung und die Multimedia Kommunikation beizubringen.

Die Programmierung erfolgt durch ein visuelles System, mittels Kacheln (auch genannt Tiles). Den Sprites können Befehle zugeordnet werden und durch das zusammenlegen der Tiles entstehen die Programme. Die Programme steuern Figuren und Objekte innerhalb des Spiels.

## 1.2 Die Scratch Oberfläche



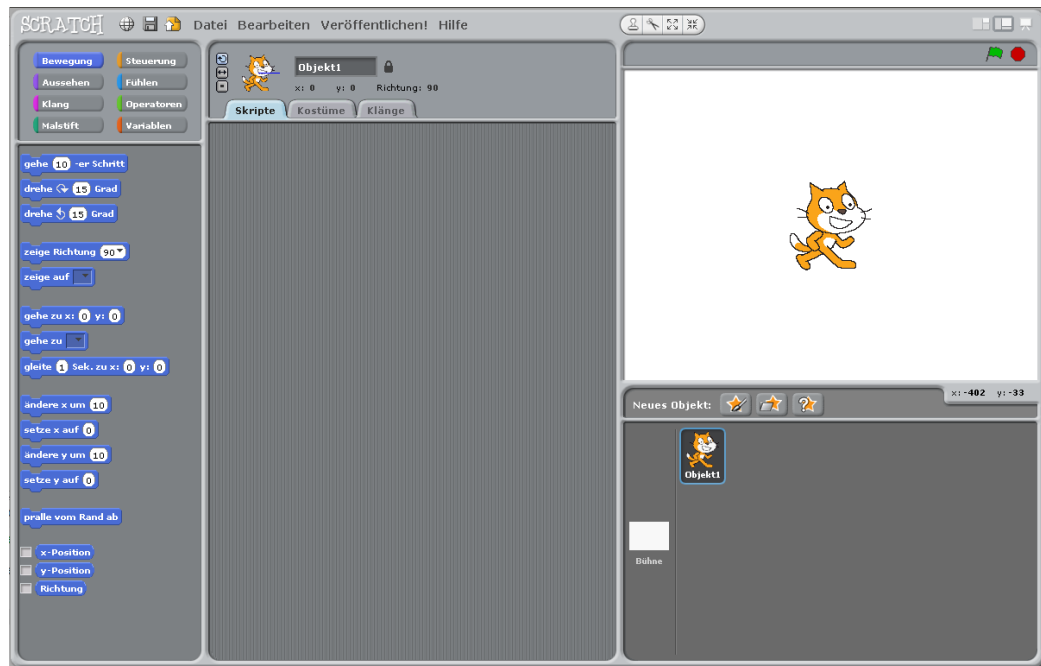
### 1.2.1 Wähle einen Sprite

Ein Sprite ist eine Figur oder ein Objekt in deinem Spiel. Die Sprites können sich bewegen oder still stehen. Wir wählen eine Sprite-Figur und lassen sie über deinen Bildschirm laufen.

1. Öffne Scratch
2. Öffne den Ordner in den Scratch installiert wurde

3. Klicke doppelt auf das Scratch-Icon.

4. Nun siehst du den Startbildschirm

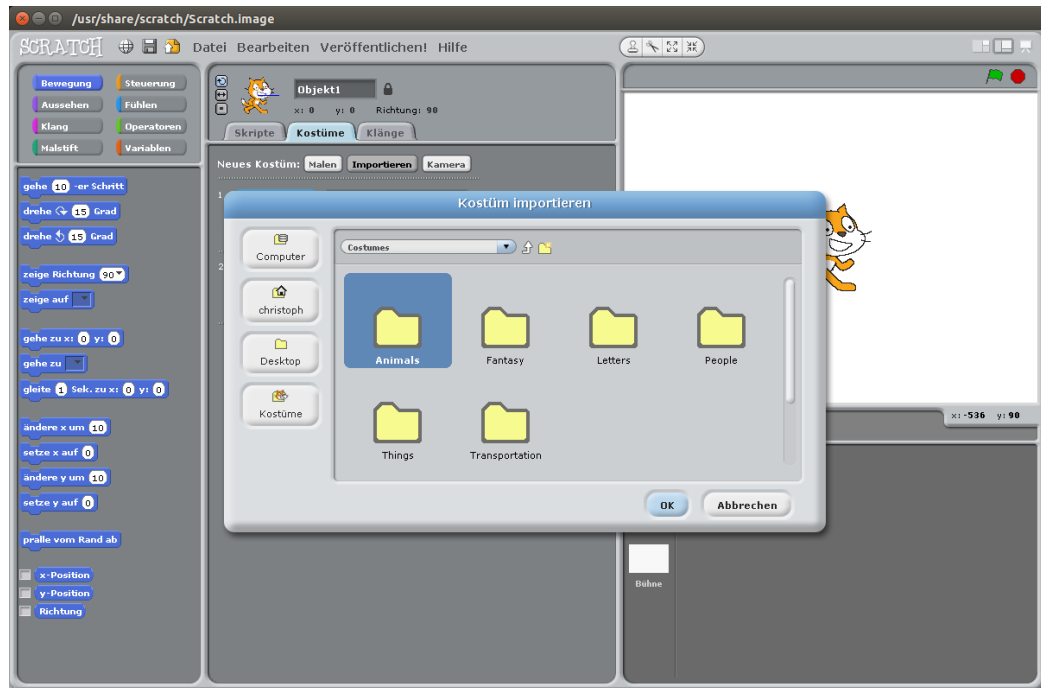


5. Klicke auf das Panel *Kostüme*

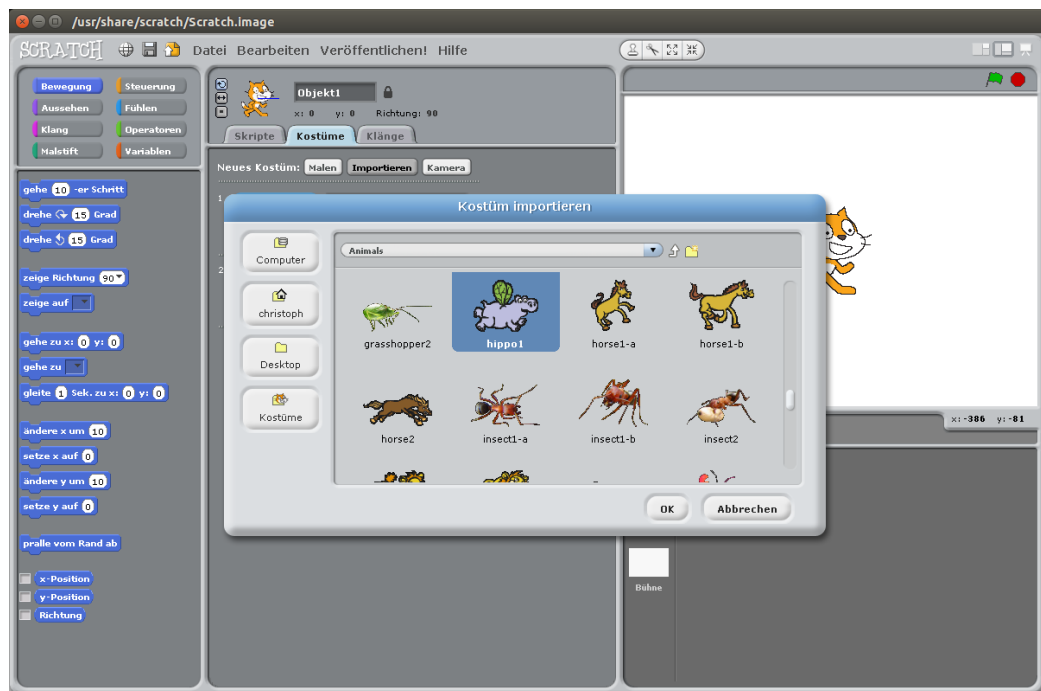


6. Klicke auf *Importieren*

7. Wähle den Ordner (Animals (dt. Tiere), People (dt. Menschen), Things (dt. Dinge))



8. Wähle einen Sprite! (Doppelklick)

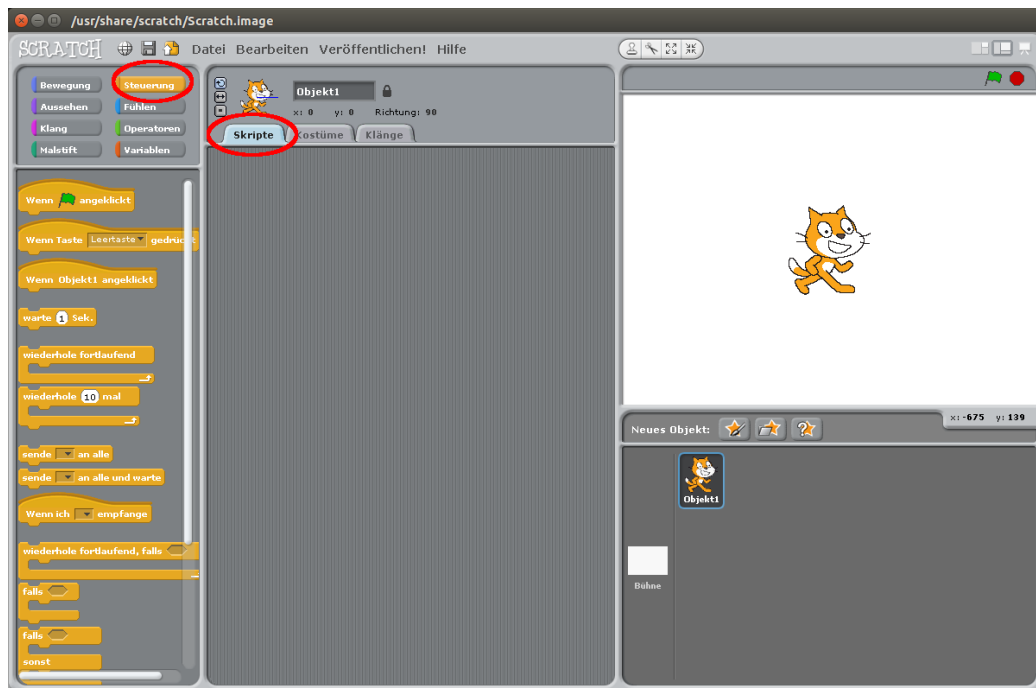


### 1.2.2 Bewege deinen Sprite in 4 Richtungen (Rechts, Links, Hoch, Runter)

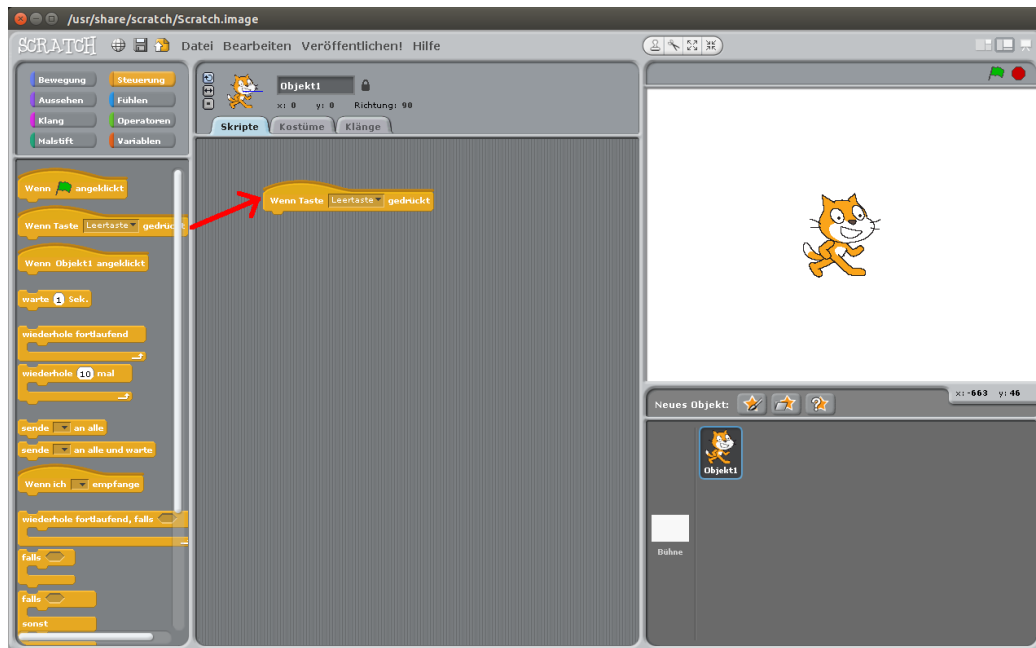
Sprites können nichts von sich aus. Eine Sprite-Aktion kommt direkt aus einem Skript aus dem Skript-Editor. Diese Skripte sind die Anweisungen, was ein Sprite tun soll.

Du ziehst die Anweisung mit deiner Maus aus dem linken Tilespalte in den Skriptbereich. Die Tiles passen wie Puzzleteile ineinander und bilden eine Anweisung.

1. Klicke auf den Sprite-Reiter.
2. Wenn du dein Sprite nach rechts bewegen willst, klicke auf den Steuerung-Button.



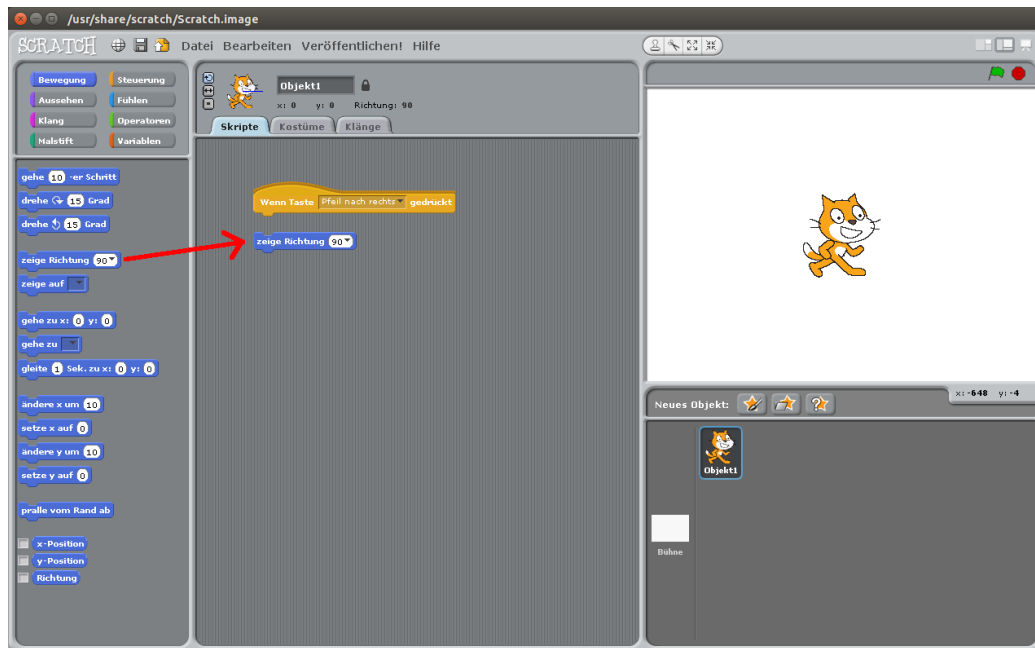
3. Klicke mit der linken Maustaste auf eine Anweisung und halte die Taste gedrückt. Ziehe nun das Tile mit dem Befehl *Wenn Taste Leertaste gedrückt* nach rechts in die Skriptfläche.



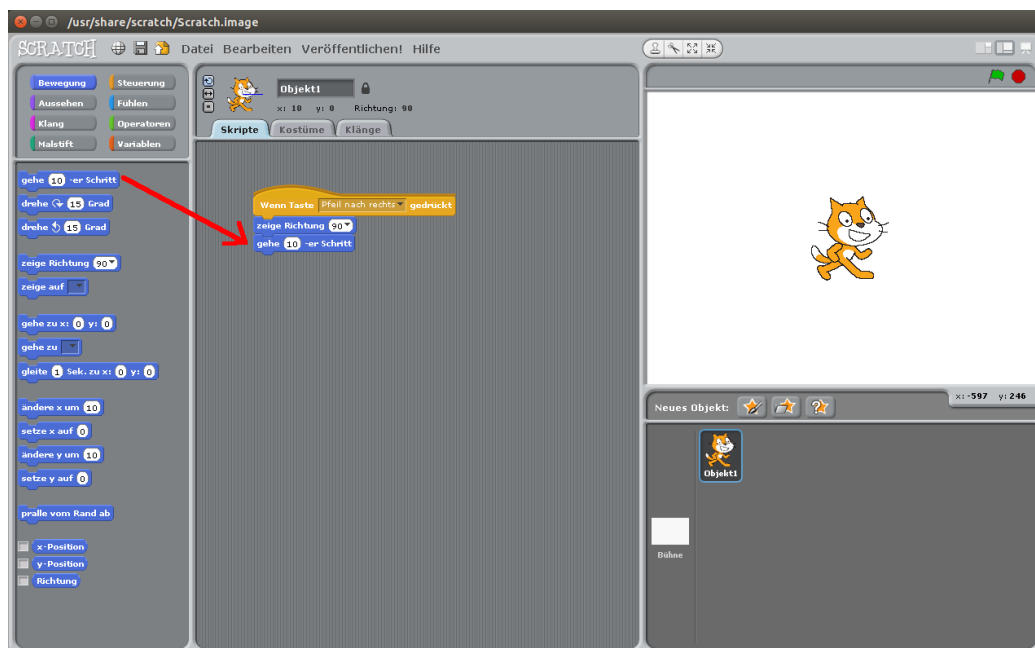
4. Klicke auf das Wort *Leertaste* und wähle *Pfeil nach rechts* aus. (Wir bewegen den Sprite damit nach rechts)



5. Klicke auf den Bewegung-Button, oben links, und ziehe das Tile "zeige Richtung 90" in das Skript-Fenster.



6. Verbinde *Wenn Taste Pfeil nach Rechts* mit *zeige Richtung 90*.
7. Klicke auf das *gehe 10-er Schritte* und ziehe es in das Skript-Fenster.
8. Verbinde die Kacheln wie auf dem Bild.

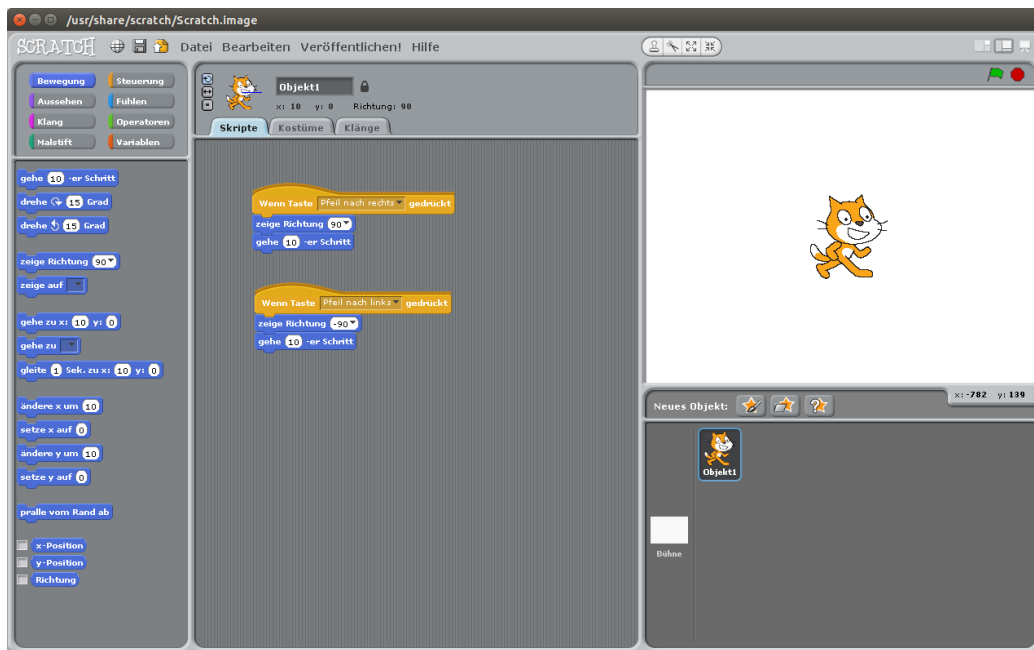


9. Klicke den Pfeil-Button deiner Tastatur und deine Figur bewegt sich.



Baue nun die gleiche Steuerung für die anderen drei Richtungen!

10. Bewegen wir unser Sprite nach links: Ziehe den *wenn Leertaste gedrückt*-Tile in das Skriptfenster .
11. Ändere *Leertaste* zu *Pfeil nach links gedrückt*.
12. Ziehe *zeige Richtung 90* aus dem Bewegungsfenster in das Skript-Fenster.
13. Ändere die 90 zu -90.
14. Ziehe das *gehe 10er Schritte*-Tile in das Skriptfenster und verbinde es mit dem vorherigen Kachel. ( Wenn es bei dir aussieht wie auf dem Bild, ist alles korrekt!)

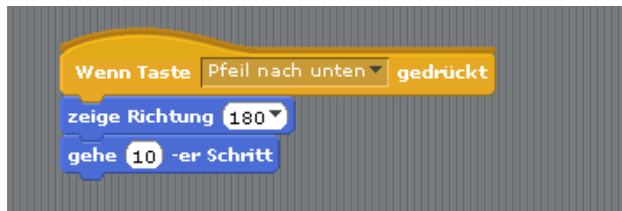


15. Jetzt sollte auch dein linker Pfeil auf der Tastatur funktionieren! Klicke auf den Doppelpfeil, damit sich die Figur auch in die richtige Laufrichtung dreht.
16. Lass deine Figur sich nach unten bewegen. Ziehe und verbinde dazu nun folgende Kacheln:

*Wenn Leertaste gedrückt*  
*Zeige Richtung 90*  
*gehe 10 Schritte*

17. Ändere *Leertaste* zu *Pfeil nach unten*.

18. Jetzt funktioniert auch der Pfeil nach unten auf deiner Tastatur.



19. Lass deine Figur sich nach oben bewegen: Ziehe und verbinde dazu folgende Kacheln:

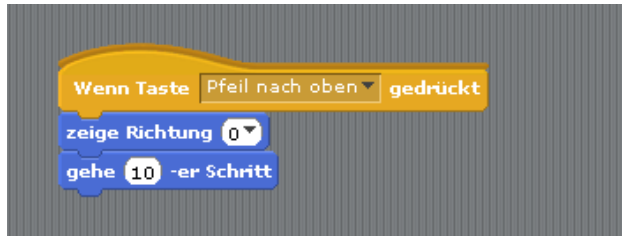
*Wenn Leertaste gedrückt*

*Zeige Richtung 90*

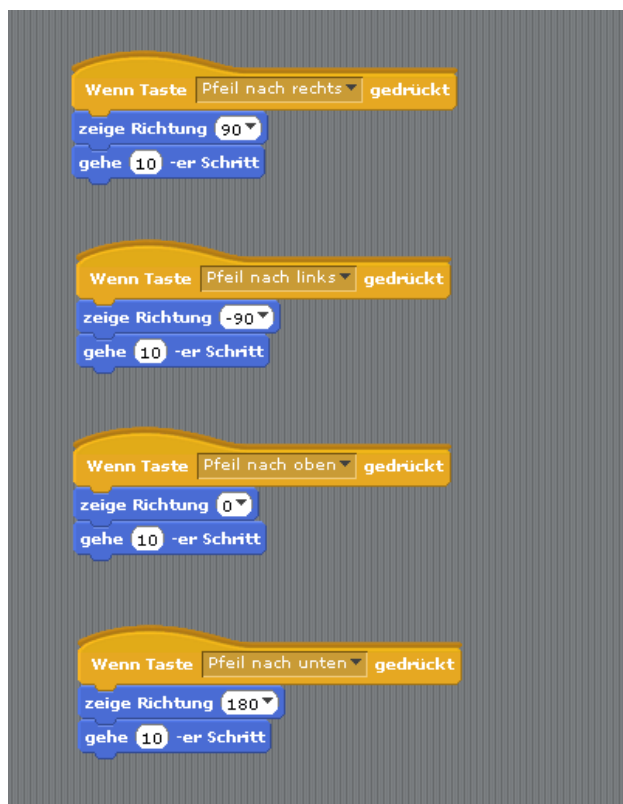
*gehe 10 Schritte*

20. Setze die Richtung auf *0 - Oben*:

21. Ändere *Leertaste* zu *Pfeil nach oben*



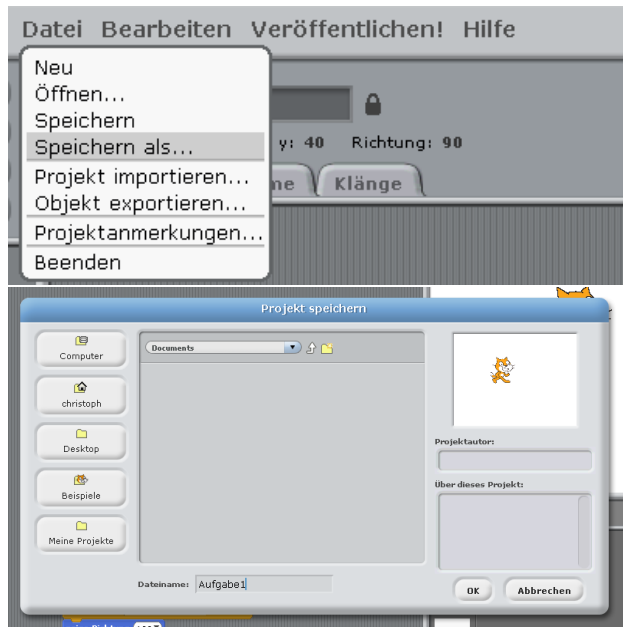
22. Jetzt bewegt sich dein Sprite in alle 4 Richtungen. Überprüfe es !



23. Gib deinem Sprite einen Namen.



## 24. Speichere dein Programm

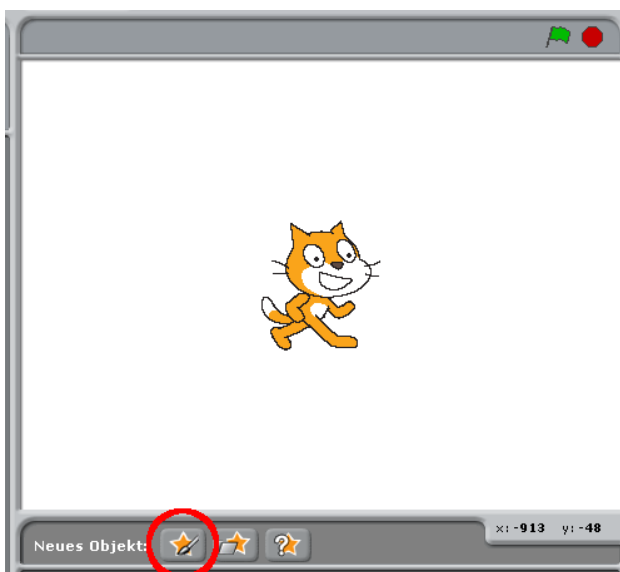


## 2 Spüre die Welt – Bedingte Anweisung und Verzweigung

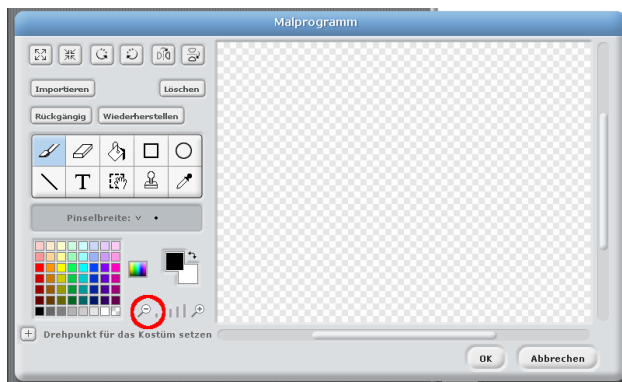
### 2.1 Mal ein Labyrinth

Wir brauchen ein Labyrinth als Barriere für unsere Hauptfigur. Wir erzeugen dazu ein weiteres Sprite, welches als Barriere dient.

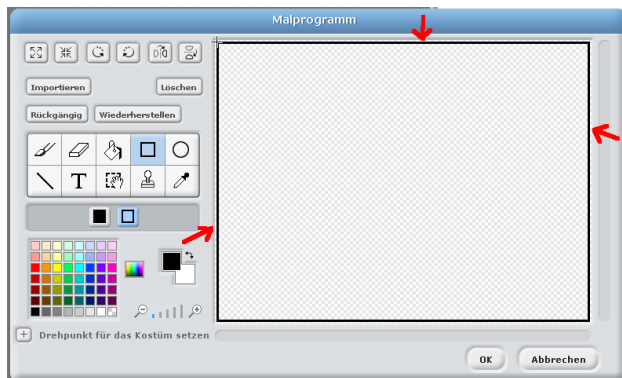
1. Klicke auf das *Neues Objekt malen*-Icon.



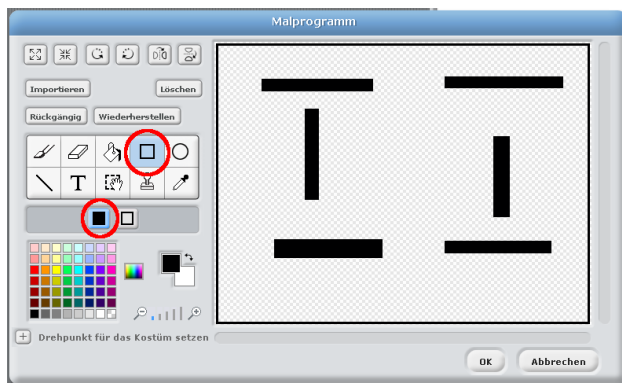
2. Benutze das Zoom-Tool um komplett herauszuzoomen. Klicke auf die Minus-Lupe.



3. Klicke auf das Rechteck-Symbol und male eine Rechteck um die gesamte Bildfläche.



4. Benutze das Rechteck Tool mit ausgewähltem Füllmodus, um einige Hindernisse zu malen.



5. Klicke auf Ok, um deine Zeichnung abzuschließen.
6. Ändere den Namen des Sprites zu Labyrinth.

## 2.2 Abprallen vom Hinderniss

1. Klicke auf deine Sprite-Figur.
2. Ziehe jetzt aus dem Steuerung-Panel, das *Wenn grüne Flagge angeklickt* in das Skript-Panel.



3. Füge nun das *wiederhole fortlaufend* hinzu.



4. Füge nun aus dem Bewegungs-Panel, *gehe 10er Schritt* in die Lücke.



5. Hänge daran nun aus dem Steuerungs-Panel, die Kachel *falls* hinzu.



6. Wechsle nun in das Fühlen-Panel

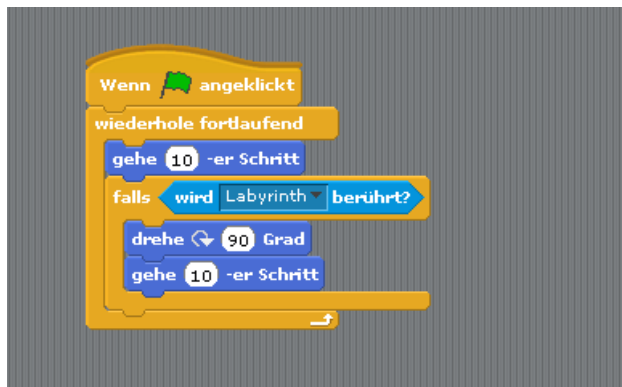


7. Wähle nun die *wird berührt*-Kachel, ziehe es ins Skript-Panel in die Wabe der Falls-Kachel und wähle dort das Labyrinth.



8. Anschließend wählst du aus dem Bewegungs-Panel *drehe um 90 Grad* und *gehe 10-er Schritte* und ziehst es in die Lücke unter *falls*.



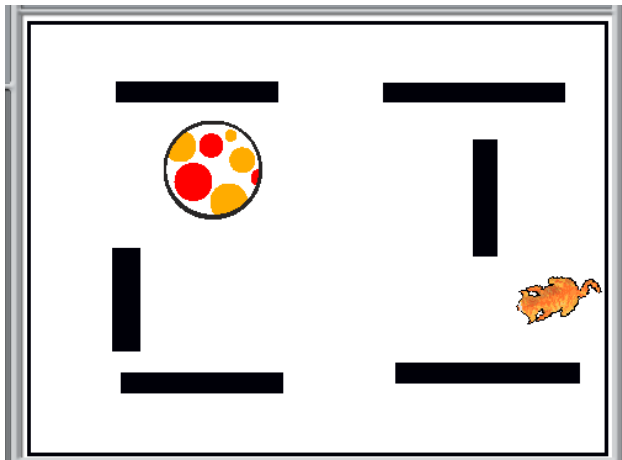


9. Teste dein Programm. Drücke auf die grüne Flagge und starte dein Programm. Beeinflusse die Richtung mit Hilfe der Pfeiltasten deiner Tastatur.

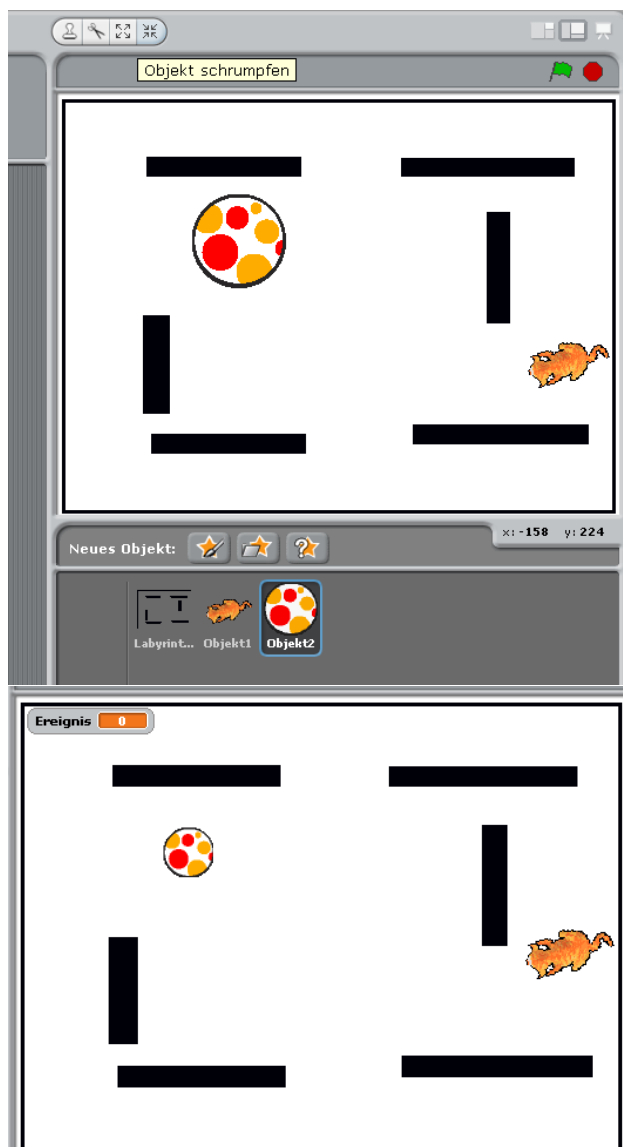
### 3 Etwas zu Essen. Bedingungen, Variablen, Ausblenden, and Sound.

#### 3.1 Erzeuge das Essen!

1. Wähle ein neues Sprite aus einer Datei. Wir haben einen Ball gewählt, du kannst jedoch auch etwas anderes nehmen oder sogar selber zeichnen.



2. Benutze das Verkleinerungs-Tool, um die Größe deines Sprites in Relation zum Rest zu bringen.



3. Benenne dein Sprite um. Meins heißt *Ball*.

]

### 3.2 Erzeuge das Skript für das Ergebnis und das Ausblenden

4. Klicke auf *Variablen* und auf *Neue Variable*, nenne Sie *Ergebnis* und klicke auf *OK*.



5. Ziehe aus dem Steuerungs-Panel, die *Wenn grüner Pfeil angeklickt* und die *wiederhole fortlaufen*-Kachel in das Skript-Panel und verbinde beide.

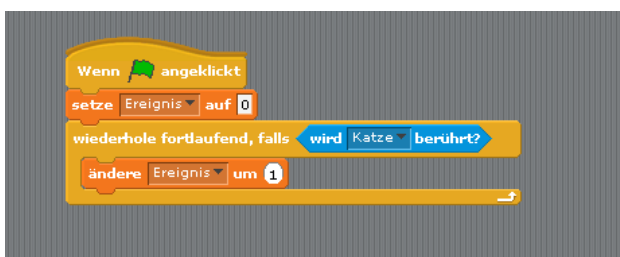


6. Füge folgendes noch hinzu:

Setze das Ergebnis auf 0 und füge es zwischen der *Wenn grüner Pfeil angeklickt* und *wiederhole fortlaufend* ein.

Füge aus dem Fühlen-Panel die *wird berührt*-Kachel in die Wabe der *wiederhole fortlaufend*-Kachel ein.

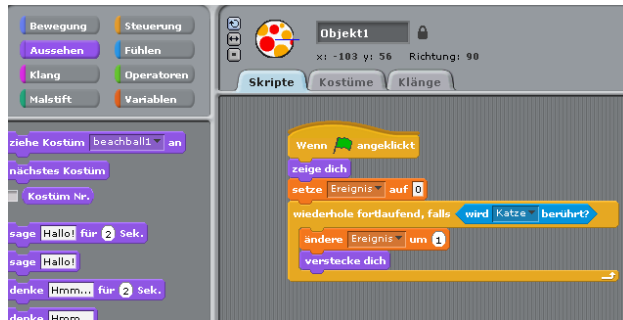
Füge in die Freie Fläche der *wiederhole fortlaufend*-Kachel aus dem Variablen-Panel die *ändere Ergebnis um 1*-Kachel ein.



7. Füge nun aus dem Aussehen-Panel eine *zeige dich* und *verstecke dich*-Kachel hinzu.

Die *zeige dich*-Kachel direkt unter das *Wenn grüner Pfeil geklickt* und *Setze Ergebnis auf 0*.

Die *verstecke dich*-Kachel unterhalb *ändere Ergebnis um 1*



8. Überprüfe deine Einstellung mit dem Bild und klicke auf den grünen Pfeil.
9. Der Ball sollte ausgeblendet werden und das Ergebnis um 1 erhöht sein.

### 3.3 Füge einen Klang hinzu, wenn der Ball deinen Sprite berührt

10. Klicke auf deine Figur und wähle den Klang-Panel.



11. Klicke auf Importieren und wähle einen Klang aus der Ordnerliste.



12. Du kannst den ausgewählten Sound testen, indem du das Play-Zeichen klickst.



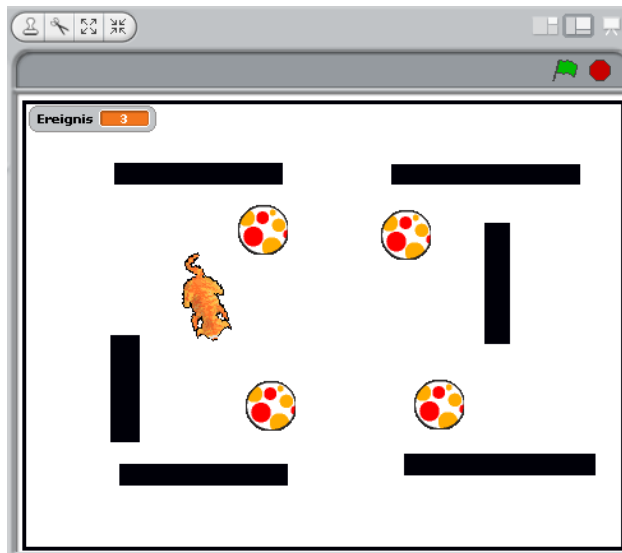
13. Klick auf den Klang-Panel, wähle die *spiele Klang*-Kachel und ziehe sie unter die *verstecke dich*-Kachel in dem Ball Sprite Skript-Editor.



14. Klicke auf den grünen Pfeil und teste ob du etwas hörst, sobald deine Figur den Ball berührt.

### 3.4 Kopiere Objekte

15. Sobald du zufrieden bist mit deinen Ball-Sprites und den dazugehörigen Skripten, klicke auf das Stempel-Tool, um das gewünschte Sprite zu kopieren. So kannst du die Bälle vermehren und das dazugehörige Verhalten gleich mit.
16. Kopiere die Bälle und ordne Sie innerhalb des Labyrinths an.



### 3.5 Füge Hintergrundmusik hinzu

17. Du kannst Musik in dein Spiel einfügen. Scratch spielt alle Lieder mit der Dateiendung .mp3 und .wav ab.
18. Klicke auf das Bühnen Icon.
19. Klicke auf das Klang-Panel und befolge Anweisungen aus Schritt 3.
20. Ziehe die *Wenn grüne Flagge geklickt*-Kachel in das Skript-Panel.
21. Hänge daran, die *wiederhole fortlaufend*-Kachel.
22. Füge in diese den Klang ein.
23. Drücke die grüne Flagge und teste dein Spiel.

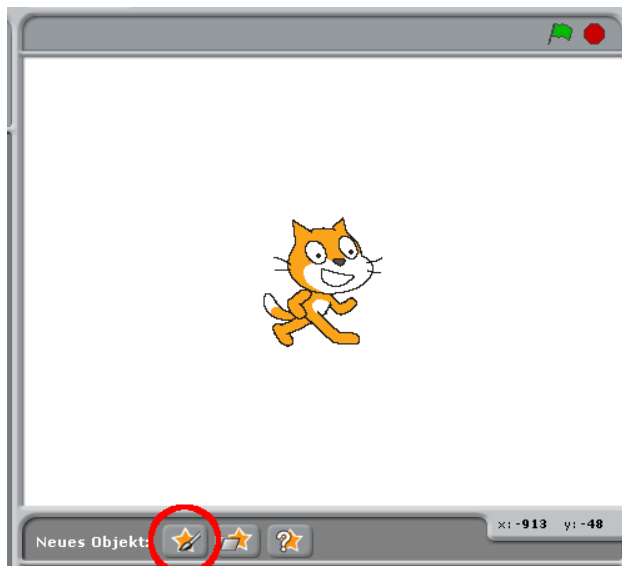
## 4 Führe die Katze

In dieser Aufgabe bauen wir ein kleines Spiel. Es soll eine Katze auf einer Rennbahn geführt werden ohne, dass die Wand berührt werden darf. Einige von euch kennen ein ähnliches Spiel unter dem Namen *heißer Draht*.

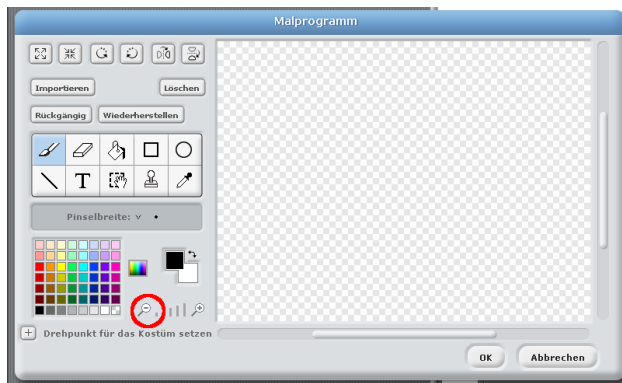
### 4.1 Male eine Rennbahn

Wir brauchen eine Rennbahn für unsere Hauptfigur. Wir erzeugen dazu ein weiteres Sprite.

1. Klicke auf das *Neues Objekt malen*-Icon

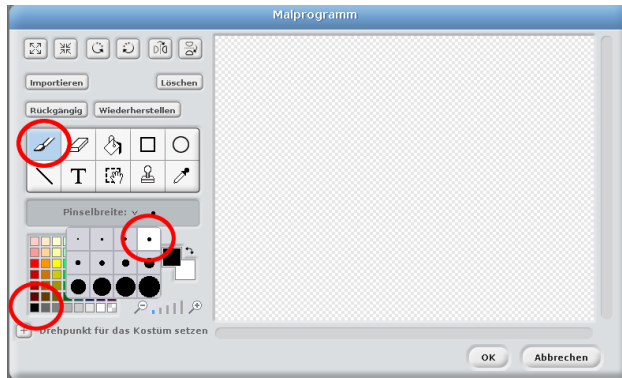


2. Benutze das Zoom-Tool um komplett herauszuzoomen. Klicke auf die Minus-Lupe.

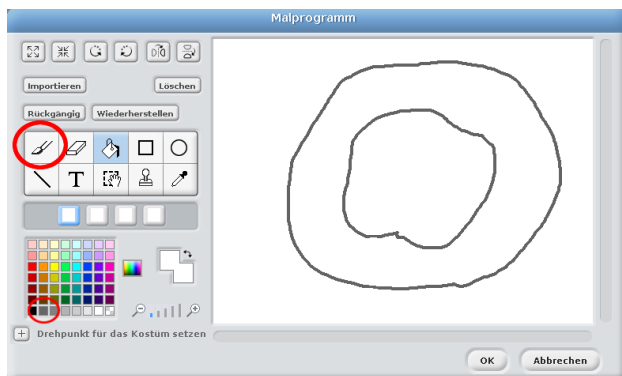




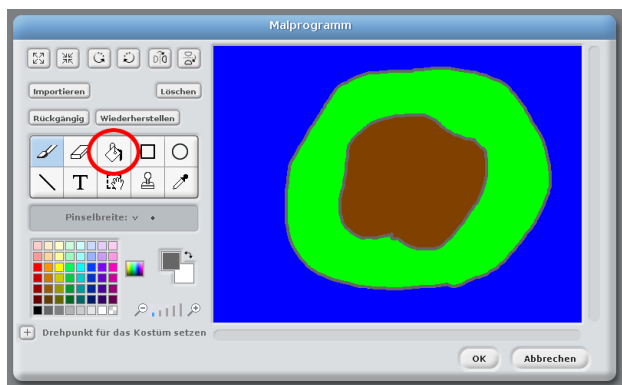
3. Klicke auf den Pinsel um diesen auszuwählen, wähle einen dickeren Pinsel und die Farbe grau.



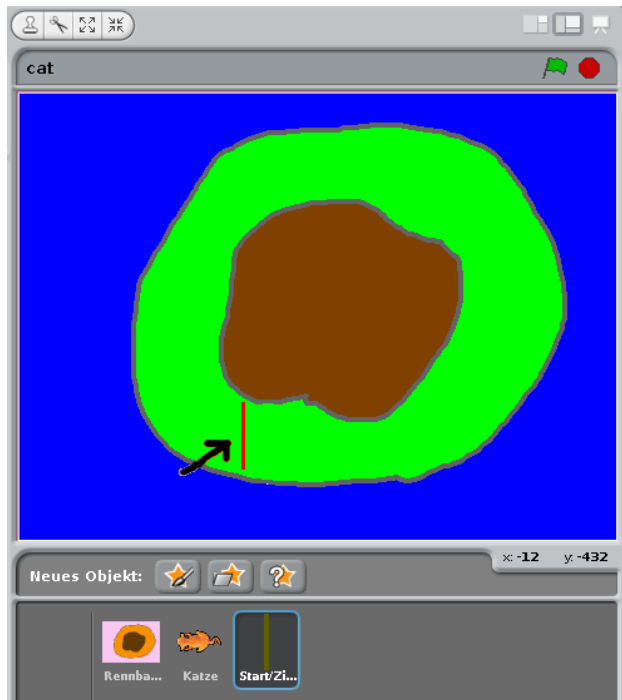
4. Benutze den Pinsel um eine Rennbahn zu malen.



5. Da das jetzt einer Rennbahn noch nicht so ähnlich sieht füllen wir die Rennfläche mit der Farbe *hellgrün* und den Berg in der Mitte mit der Farbe *braun* und alles andere Hellblau. Du kannst die Flächen mit dem Pinsel anmalen, leichter ist es jedoch mit dem *Farbeimer*. Dazu einfach den Farbeimer und die Farbe auswählen und dann auf die gewünschte Fläche klicken.



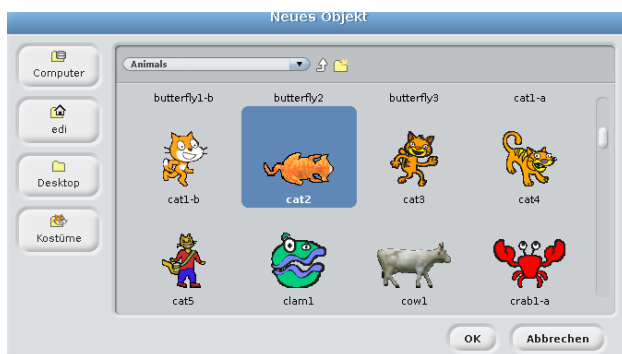
- Zum Schluss fügen wir noch ein weiteren Sprite hinzu und zwar eine *Start/Ziel-Linie* in der Farbe *rot*, diese erstellen und auf die gewünschte Position schieben.



- Ändere den Namen des Sprites zu *Rennbahn* und den Namen des Sprites für die Start/Ziel-Linie zu *Start/Ziel*.

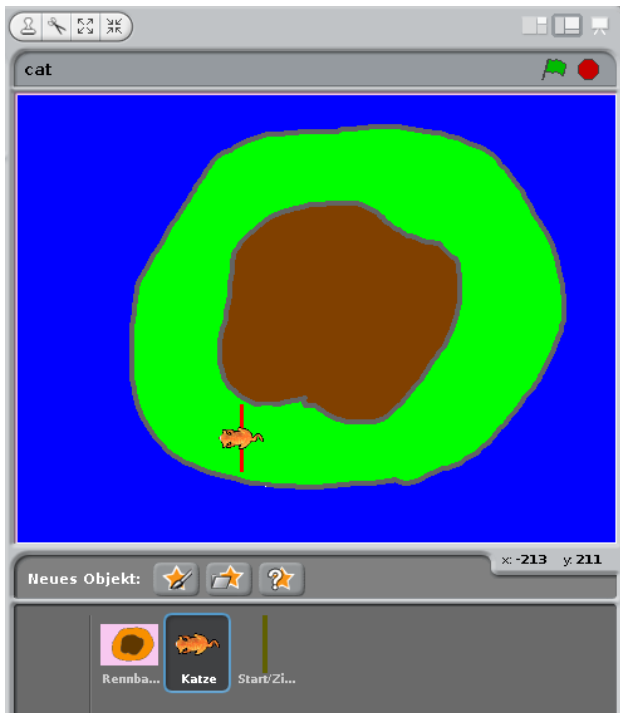
## 4.2 Katze einfügen

- Füge den Sprite *Katze* aus einer Datei hinzu.



- Ändere den Namen des Sprites zu *Katze*.

- Benutze das Verkleinerungs-Tool, um die Größe deiner Katze soweit zu verkleinern, dass sie zur Größe der Bahn passt und ohne anzustoßen auf der Bahn laufen kann.



### 4.3 Erzeuge das Script für die Steuerung der Katze

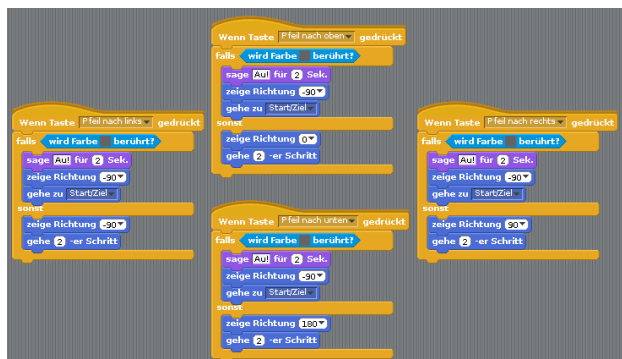
- Klicke auf den Sprite Katze.
- Ziehe folgende Kacheln in dein Skript-Panel:
  - Aus dem Steuerungs-Panel die Kachel *Wenn Taste Leertaste gedrückt* und *falls...sonst* und setze beide zusammen.
  - Aus dem Fühlen-Panel die Kachel *wird Farbe berührt*. Diese Kachel setzt du als Bedingung in die *falls...sonst*-Kachel
  - Aus dem Aussehen-Panel suchst du die Kachel *sage Hallo! für 2 Sek.* raus und fügst diese in den Bereich *falls*, der *falls...sonst*-Kachel ein.
  - Im Bewegungs-Panel findest du nun die vier letzten Kacheln für die Steuerung der Katze: *zeige in Richtung 90* und *gehe zu* fügst du in den *falls*-Bereich, gleich nach der Kachel *sage Hallo! für 2 Sek.* ein. In den *sonst*-Bereich ziehst du die beiden Kacheln *zeige in Richtung 90* und *gehe 10-er Schritt*.



3. Bei der Kachel *Wenn Taste Leertaste gedrückt* klickst du auf den Text *Leertaste* und wählst aus der Liste *Pfeil nach oben*.
4. Bei der Kachel *wird Farbe berührt* auf die Farbe klicken und dann auf den grauen Rand unserer Rennbahn.
5. Bei der Kachel *sage Hallo! für 2 Sek.* auf den Text *Hallo!* klicken und diesen zu *Au!* ändern.
6. Bei der Kachel *zeige in Richtung 90* im *falls*-Bereich auf die Zahl *90* klicken und aus der Liste *(-90) links* auswählen.
7. Bei der Kachel *gehe zu* auf den kleinen Pfeil klicken und unseren Sprite *Start/Ziel* aus der Liste auswählen.
7. Bei der Kachel *zeige in Richtung 90* im *sonst*-Bereich auf die Zahl *90* klicken und aus der Liste *(0) oben* auswählen.
8. Bei der letzten Kachel *gehe 10-er Schritt* soll die Zahl *10* durch eine *2* ersetzt werden.



9. Das Ganze jeweils für die drei restlichen Richtungen *rechts*, *unten* und *links* wiederholen und dabei nicht vergessen bei der Kachel *zeige in Richtung 90* jeweils die gewünschte Richtung auswählen und die richtige Taste zu ändern.



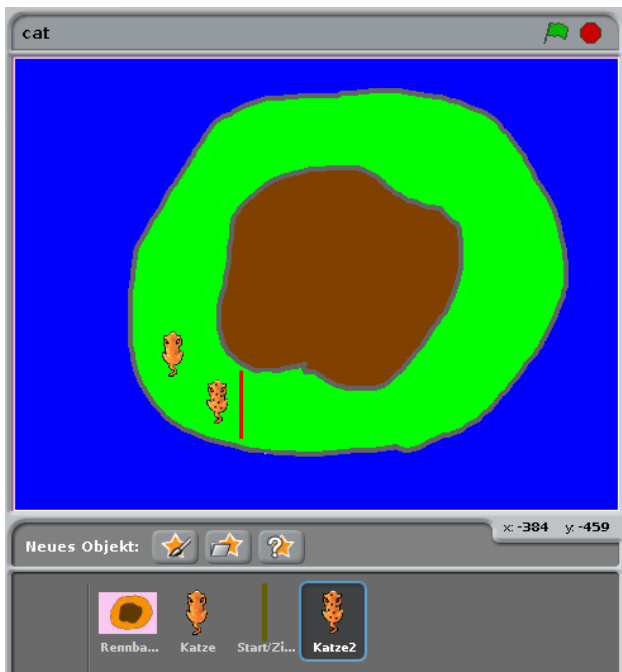
10. Um die Katze beim Start des Spiels zur Startlinie zu bringen, füge die Kacheln *Wenn Fahne angeklickt* aus dem Steuerungs-Panel, aus dem Bewegungs-Panel *zeige Richtung 90* und *gehe zu* ein und setze diese zusammen.
11. Bei der Kachel *zeige in Richtung 90* im *sonst*-Bereich auf die Zahl *90* klicken und aus der Liste *(-90) links* auswählen.
12. Bei der Kachel *gehe zu* auf den Pfeil klicken und aus dem Menü und *Start/Ziel* auswählen.



#### 4.4 Zusatzaufgabe: Erweitere das Spiel damit zwei Spieler spielen können

Du kannst einen weiteren Sprite hinzufügen und dir weitere vier Tasten auf der Tastatur aussuchen um diesen auch eine Steuerung zu geben. Hier die kurze Beschreibung wie man das machen könnte:

1. Die vier Tasten für die Steuerung überlegen z.B.
  - w = oben
  - d = rechts
  - s = unten
  - a = links
2. Kopiere die Katze und passe die Skripte an, sodass die von dir ausgewählten Tasten zur Steuerung der Katze benutzt werden]
3. Ändere das Aussehen einer Katze um es leichter zu machen sie zu Unterscheiden (wir haben ihr ein paar Punkte gegeben).



## 5 Werde zum Tennisspieler

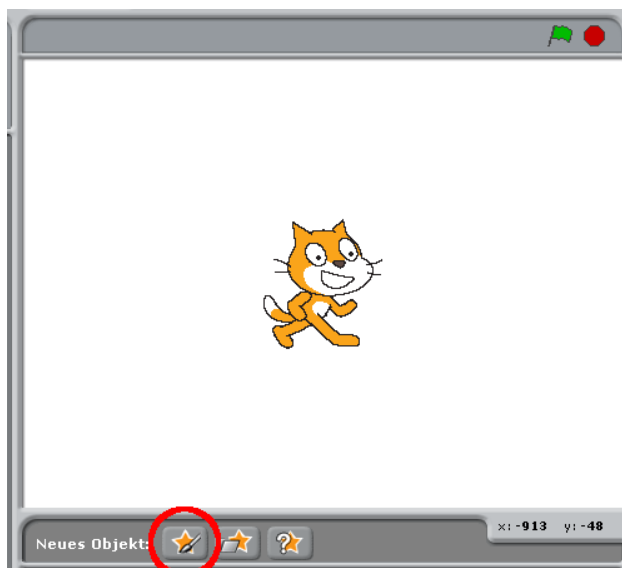
In dieser Aufgabe bauen wir ein kleines Tennisspiel, das auch unter dem Namen *Pong* bekannt ist. Ziel des Spieles ist es, einen Ball mit dem Schläger zu treffen. Verfehlt man den Schlag und trifft der Ball hinter einem auf den Rand, so erhält der Gegner einen Punkt. Wer als erster 10 Punkte erlangt hat, hat gewonnen.

### 5.1 Programmiere einen Spieler

Wir brauchen für das Spiel zwei Spieler. Beide verhalten sich gleich, daher reicht es einen Spieler zu malen und ihn im Anschluss zu kopieren. Wir erzeugen dazu ein weiteres Sprite.

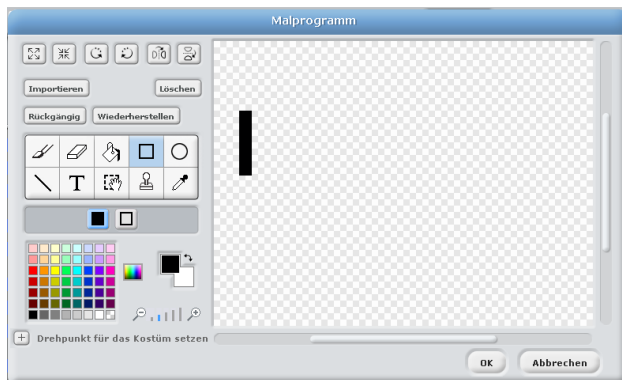
#### 5.1.1 Male den Spieler

1. Klicke auf das *Neues Objekt malen*-Icon

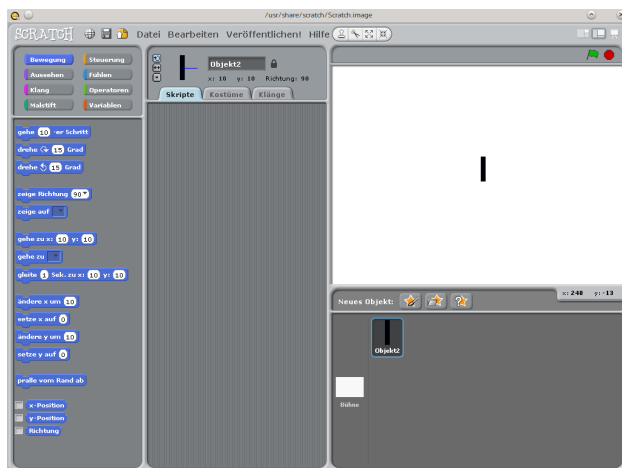


2. Klicke auf das Quadrat, um das Rechteck-Tool auszuwählen.
3. Prüfe ob das ausgefüllte Quadrat unter den Werkzeugen selektiert ist.
4. Stelle sicher, dass die Farbe Schwarz ausgewählt ist
5. Zeichne nun durch gleichzeitiges Klicken und Ziehen mit der Maus ein Rechteck so wie es in folgendem Bild dargestellt ist.



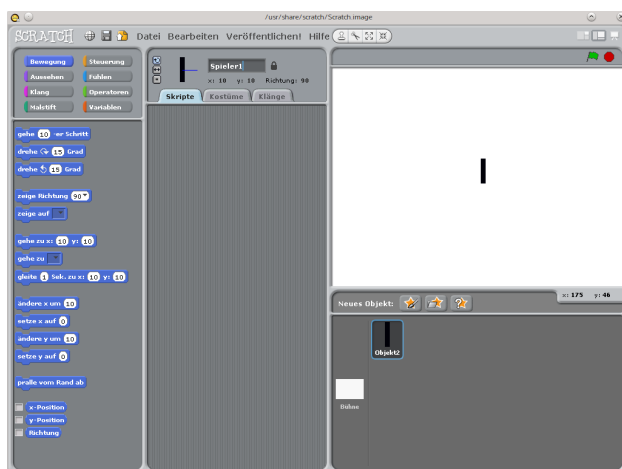


6. Abschließend beende den Dialog durch betätigen der OK-Taste



7. Nun soll das neue Sprite noch den Namen *Spieler1* erhalten

8. Als letztes lösche das Objekt mit der Katze, sofern es noch vorhanden ist

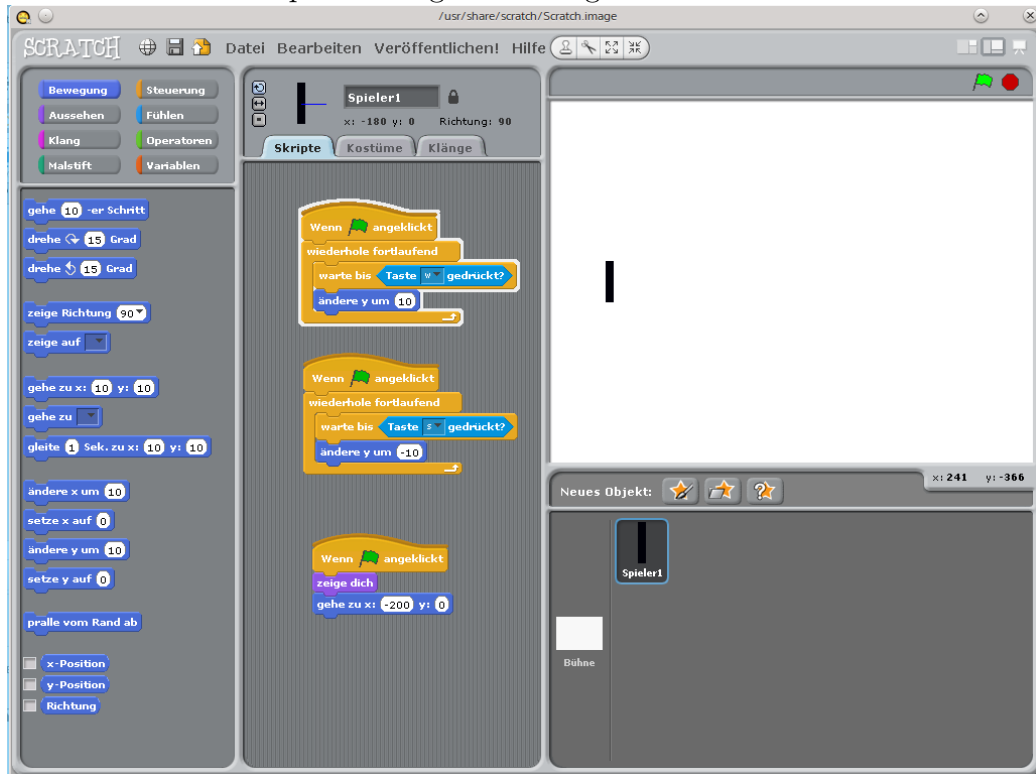


### 5.1.2 Programmiere das Verhalten des Spielers

Nachdem nun der Spieler gemalt wurde muss sein Verhalten programmiert werden: Der Spieler soll durch betätigen der Taste *w* nach oben und durch betätigen der Taste *s* nach unten bewegt werden können.

1. Klicke auf den Sprite *Spieler1*.
2. Für die Bewegung nach oben ziehe folgende Kacheln in dein Skript-Panel:
  - (a) Aus dem Steuerungs-Panel die Kachel *Wenn Taste Leertaste gedrückt* und *wiederhole fortlaufend* und setze sie zusammen.
  - (b) Füge nun aus dem Steuerungs-Panel die Kachel *warte bis* in die Schleife ein.
  - (c) Aus dem Fühlen-Panel die Kachel *Taste ... gedrückt?*. Diese Kachel setzt du als Bedingung in die *warte bis*-Kachel und wählst als Taste *w* aus
  - (d) Aus dem Bewegung-Panel suchst du die Kachel *ändere y um ...* raus und fügst diese unter die Bedingung *warte bis* ein. Setze den Wert in der Kachel auf *10*
3. Für die Bewegung nach unten dupliziere nun die eben angelegten Anweisungen und passe diese an:
  - (a) passe die Kachel *warte bis*-Kachel so an, dass auf die Taste *s* gewartet wird
  - (b) ändere die Kachel *ändere y um ...* auf *-10*
4. Damit der Spieler immer an der richtigen Stelle startet, ziehe folgende Kacheln in dein Skript-Panel:
  - (a) Aus dem Steuerungs-Panel die Kachel *Wenn Taste Leertaste gedrückt*
  - (b) Aus dem Aussehen-Panel die Kachel *zeige dich* und füge sie an *Wenn Taste Leertaste gedrückt*
  - (c) Aus dem Bewegung-Panel die Kachel *gehe zu x: ... y: ...*, fügst diese an die Kachel *zeige dich* und trägst für *x -200* und für *y 0* ein.

Damit ist der erste Spieler fertig. Dein Programm sollte nun so aussehen:



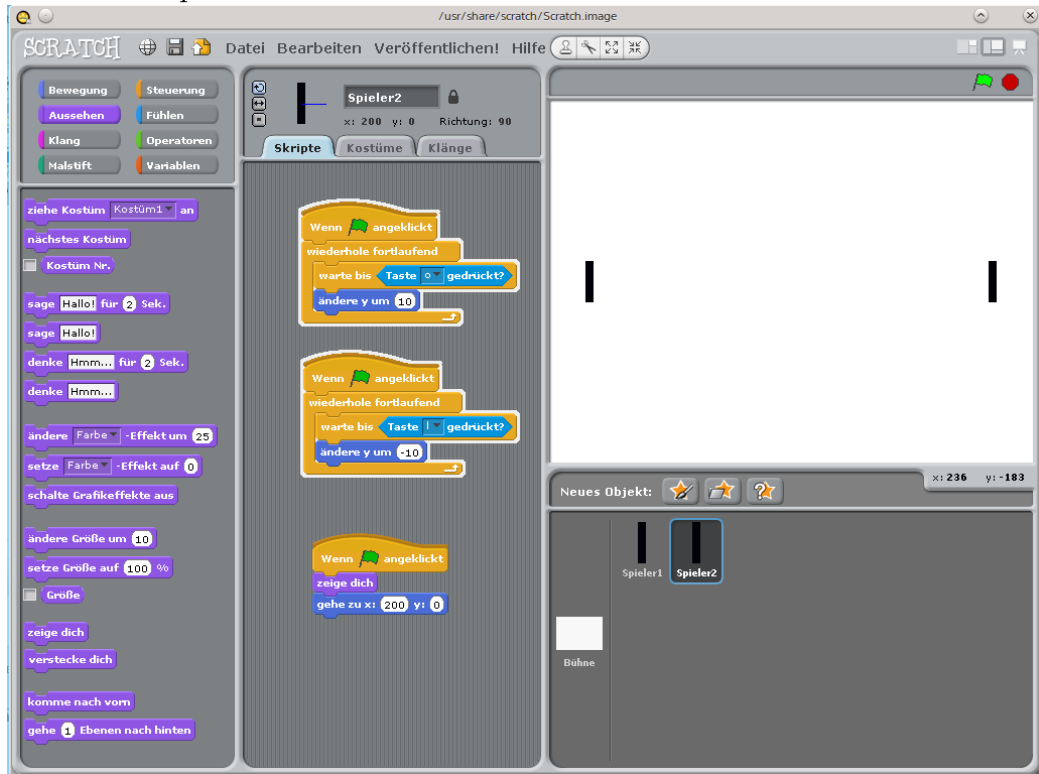
Probiere ob du deinen Spieler mit den Tasten *w* und *s* steuern kannst.

### 5.1.3 nun der zweite Spieler

Den zweiten Spieler können wir durch kopieren des ersten erstellen. Anschließend müssen wir nur noch die Position auf dem Spielfeld und die Tasten für die Bewegung anpassen.

1. Wähle den Sprite *Spieler1* und dupliziere ihn über den Menüeintrag im Kontextmenü (Das Menü erscheint, wenn du die rechte Maustaste auf dem Sprite betätigst)
2. Selektiere nun das neu erstellt Sprite und ändere seinen Namen auf *Spieler2*
3. Ändere nun in der Kachel *Taste w gedrückt?* die Taste auf *o*
4. In der Kachel *Taste s gedrückt?* änderst du die Taste auf *l*
5. Abschließend ändere die *x*-Position in der Kachel *gehe zu x: -200 y: 0* auf *200*

Der zweite Spieler sollte nun so aussehen:



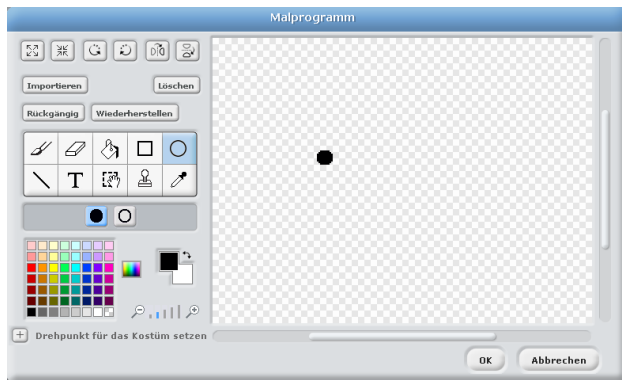
Probiere aus, ob du beide Spieler mit den Tasten *w,s* und *o,l* bedienen kannst.

## 5.2 Der Ball kommt ins Spiel

Nun muss das Verhalten des Balls programmiert werden: Der Ball soll vom Rand abprallen. Wenn ein Spieler den Ball trifft, soll der Ball zurückgeschlagen werden. Trifft der Ball hinter einem Spieler auf den Rand, erhält der andere Spieler einen Punkt.

### 5.2.1 Male einen Ball

Zunächst klicke auf das *Neues Objekt malen*-Icon, um einen Sprite für den Ball zu erzeugen. Du malst einen Ball mit Hilfe des Werkzeugs *Ellipse*:

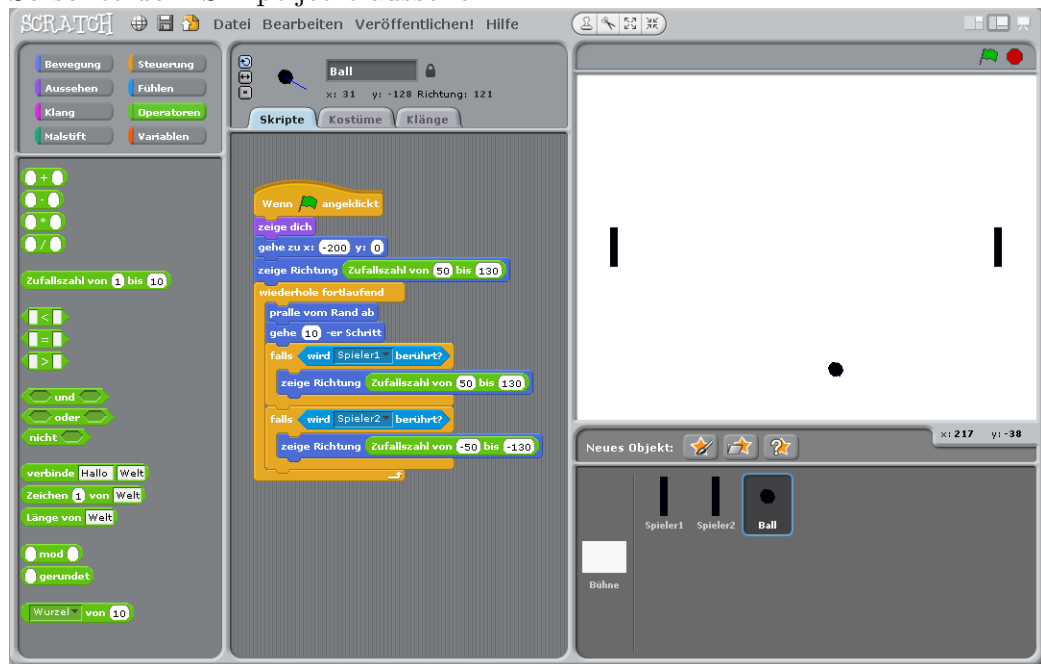


### 5.2.2 Der Ball fliegt

1. Selektiere nun das neu erstellte Sprite und ändere seinen Namen auf *Ball*.
2. Füge nun die Kachel *Wenn grüne Fahne angeklickt* dem Skript für den Ball zu.
3. Als nächsten initialisiere den Ball. Initialisieren bedeutet, beim Programmstart Dinge festzulegen, damit das Programm immer gleich abläuft. In unserem Fall bedeutet dies, beim Spielstart eine Ausgangssituation für den Ball festzulegen. So startet der Ball immer an der gleichen Position:
  - (a) Füge die Kachel *zeige dich* an die Kachel *Wenn grüne Fahne angeklickt*
  - (b) Als nächsten Befehl füge die Kachel *gehe zu x: y:* an die Kachel *zeige dich*. Setze den Wert für *x* auf *-200* und für *y* auf *0*
  - (c) Zum Spielbeginn soll der Ball in eine zufällige Richtung fliegen. Dies erreichst du indem, du die Kachel *zeige Richtung* verwendest und als Wert die Kachel *Zufallszahl von: bis:* einfügst. Trage für *von 50* und für *bis 130* ein.
  - (d) Füge die Kachel *zeige Richtung* nun an die Kachel *gehe zu x: y:* an.
4. Nach der Initialisierung kann nun der Ball zum Programmstart in eine zufällige Richtung zeigen, aber sich noch nicht bewegen. Das ist noch kein richtiges Spiel, daher sollen die nun folgenden Anweisungen immer wieder ausgeführt werden. Das erreichst du indem du die Kachel *wiederhole fortlaufend* an die letzte Kachel deines Skripts anfügst.

5. Nun kommen die Dinge, die der Ball immer wieder ausführen soll:
- Als erstes soll der Ball von der Wand abprallen, wenn er sie berührt. Dazu ziehst du die Kachel *pralle von Rand ab* in die Schleife *wiederhole fortlaufend*.
  - Damit sich der Ball bewegt, fügst du die Kachel *gehe ... er Schritt* an die Kachel *pralle von Rand ab*. Setze den Wert in der Kachel auf *10*. Dieser Wert bestimmt wie schnell der Ball fliegt.
  - Jetzt fehlt noch das Schlagen des Balls durch die Spieler. Durch den Schlag ändert der Ball zufällig seine Richtung:
    - Füge die Kachel *falls* an die letzte Kachel in der Schleife ein.
    - Verwende als Bedingung für die Kachel *wird ... berührt?* und wähle aus der Liste der Objekte dafür *Spieler1*
    - Füge nun in die Kachel *falls* die Kachel *zeige Richtung ...* ein.
    - Als Wert für die Richtung verwendest du die Kachel *Zufallszahl von: bis:*.
    - Trage für *von 50* und für *bis 130* ein.
  - Tue das Gleiche für den zweiten Spieler.
  - Für den zweiten Spieler muss der Wertebereich für die Zufallszahl von *-50 bis -130* gehen.

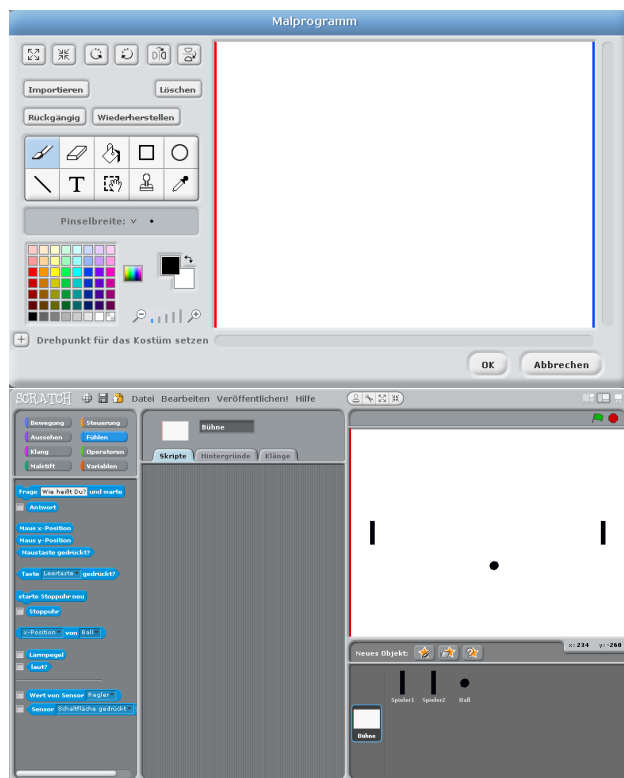
So sollte dein Skript jetzt aussehen:



## 5.3 Der Hintergrund

Nun soll der Hintergrund verändert werden. Zum einen soll der Hintergrund um Auslinien erweitert werden. Berührt der Ball diese so erhält ein Spieler einen Punkt. Ferner sollen zwei weitere Hintergründe gemalt werden: Einen der anzeigt, dass *Spieler 1* gewonnen hat und einen der anzeigt, dass *Spieler 2* gewonnen hat.

1. Selektiere die Bühne und wähle anschließend die Registerkarte *Hintergründe*
2. Klicke nun für den *Hintergrund 1* die Schaltfläche *bearbeiten*
3. Füge auf der linken Seite des Hintergrunds mit dem Werkzeug *Rechteck* eine rote Linie ein
4. Füge auf der rechten Seite des Hintergrunds mit dem Werkzeug *Rechteck* eine blaue Linie ein
5. Schließe den Dialog mit *ok*

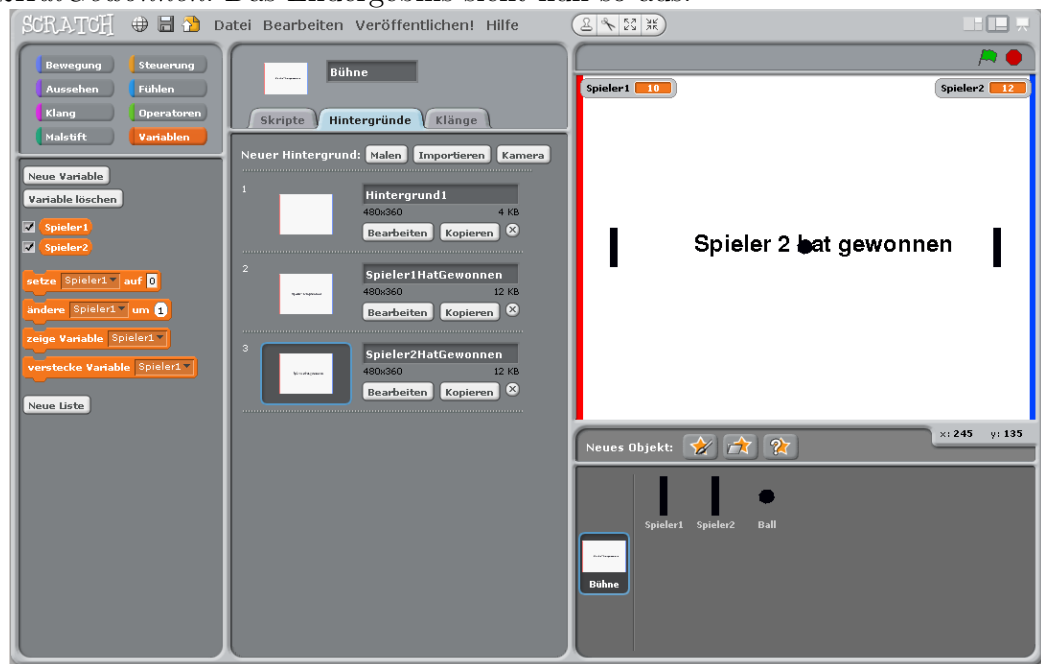


6. Nun klicke auf der Registerkarte *Hintergründe* die Schaltfläche *male*

7. Verwende das Werkzeug *Text*
8. Schreibe den Text *Spieler 1 hat gewonnen*
9. Schließe den Dialog mit *ok*
10. Benenne diesen Hintergrund *Spieler1HatGewonnen*



Wiederhole die letzten Schritte, um einen weiteren Hintergrund mit dem Text *Spieler 2 hat gewonnen* anzulegen. Nenne diesen Hintergrund *Spieler2HatGewonnen*. Das Endergebnis sieht nun so aus:

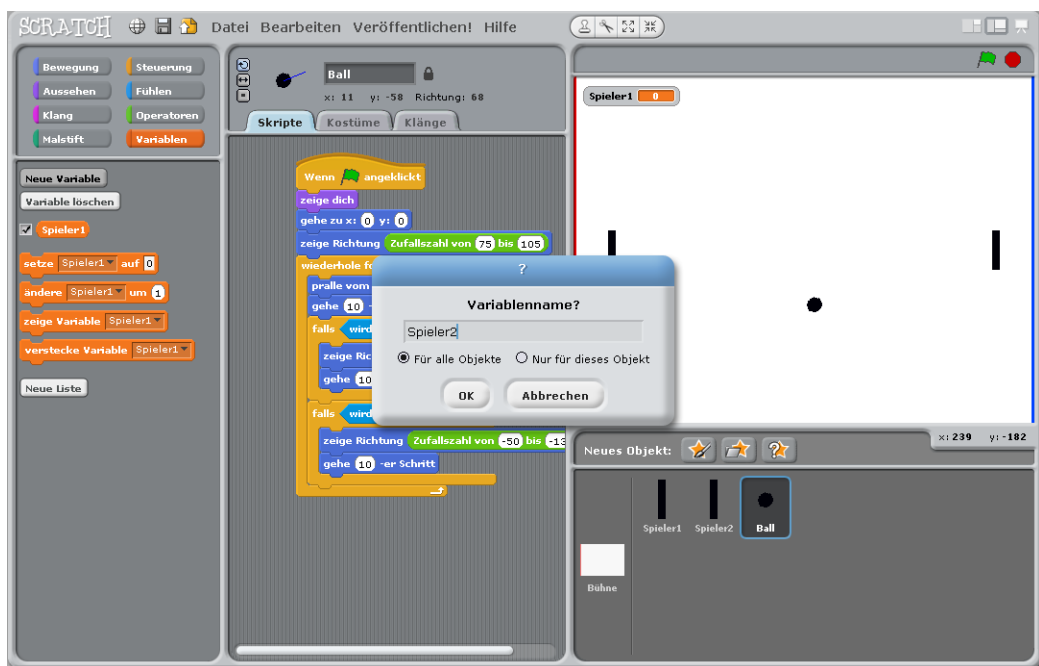




## 5.4 Wer hat gewonnen?

Als letzten Schritt soll der Spielstand gezählt werden. Der Spieler, der als erster 10 Punkte erzielt, hat gewonnen. Dafür müssen wir für jeden Spieler eine Variable anlegen. Variablen ermöglichen einen Wert hinein zu schreiben und zu einem späteren Zeitpunkt diesen Wert wieder auszulesen. Dies ist vergleichbar mit einer Tasche in die man ein Teil legt und später wieder herausnehmen oder einfach nur nachsehen kann, welches Teil in der Tasche ist.

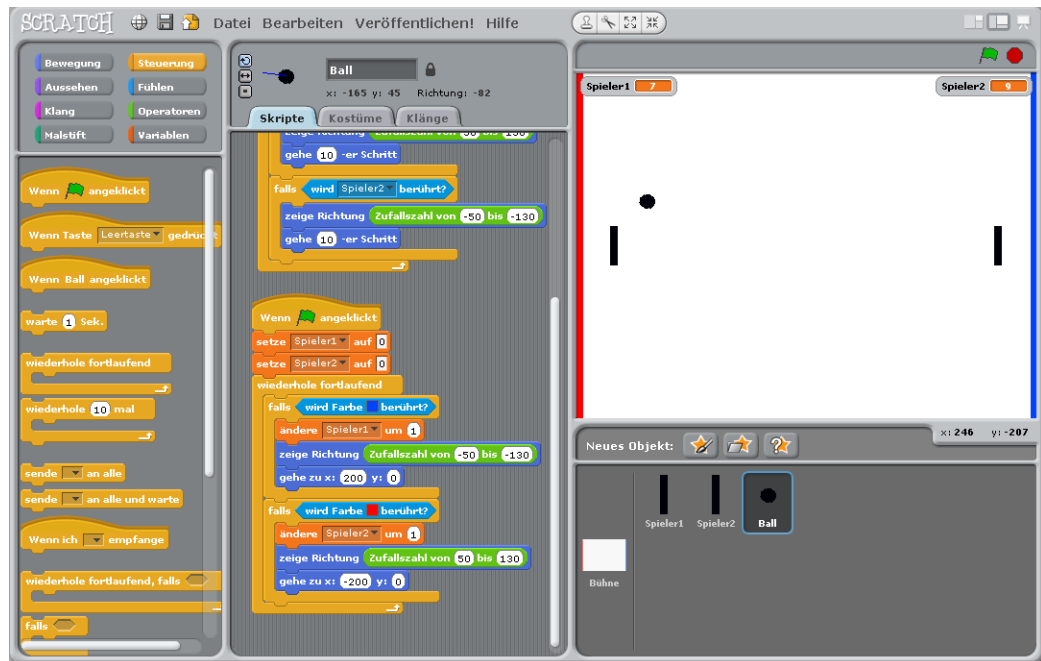
1. Klicke im Bereich Variablen auf die Schaltfläche *Neue Variable*
2. Benenne in dem sich öffnenden Dialog die Variable *Spieler1* und schließe den Dialog mit *Ok*
3. Klicke nun nochmals auf die Schaltfläche *Neue Variable*
4. Benenne diesmal die Variable *Spieler2* und schließe den Dialog mit *Ok*
5. Du kannst mit der Maus die Position der Variablen auf der Bühne verändern



Nun müssen die Variablen zum Spielbeginn mit  $0$  initialisiert und immer dann hochgezählt werden, wenn der Ball die Auslinien berührt:

6. Füge zum Skript des Balls eine weitere Kachel *Wenn grüne Fahne angeklickt* ein

7. Füge an diese Kachel aus dem Bereich Variablen die Kachel *setze ... auf ...*
8. wähle aus der Liste dieser Kachel die Variable *Spieler1* und setze den Wert auf *0*
9. füge an diese Kachel erneut eine Kachel *setze ... auf ...*
10. wähle dieses Mal die Variable *Spieler2*, setze den Wert ebenfalls auf *0*
11. Nun wird in einer Schleife geprüft, ob die Auslinien berührt werden. Füge dazu die Kachel *wiederhole fortlaufend* an die letzte Kachel
12. Innerhalb der Schleife füge nun zunächst eine Kachel *falls* ein
13. Als Bedingung dieser Kachel wähle die Kachel *wird Farbe ... berührt?*
14. Die zu prüfende Farbe ist in diesem Fall *blau*
15. Innerhalb der Kachel *falls* füge eine Kachel *ändere ... um ...* und wähle für diese Kachel die Variable *Spieler1* aus der Liste. Als Wert für *um* setze *1*
16. Nachdem ein Ball im Aus landet, muss der Spieler erneut mit zufälliger Richtung abschlagen. Füge an die Kachel für die Erhöhung der Variable eine Kachel *zeige Richtung ....* Als Wert wähle mithilfe der Kachel *Zufallszahl von ... bis ...* eine Zufallszahl im Bereich *-50* bis *-130*
17. Lasse an diese Kachel eine Kachel *gehe zu x: ... y: ...* folgen und setze für *x* den Wert *200* und für *y* den Wert *0* ein
18. Dupliziere nun die *falls*-Kachel mit den enthaltenen Anweisungen und passe diese an:
  - (a) Ändere in der Kopie die Farbe in der Bedingung auf *rot*
  - (b) Wechsel die zu erhöhende Variable auf *Spieler2*
  - (c) Erzeuge die Zufallszahlen in dem Bereich *50* bis *130*
  - (d) Abschließend trage für den *x*-Wert der *gehe zu x: ... y: ...* Kachel den Wert *200* ein

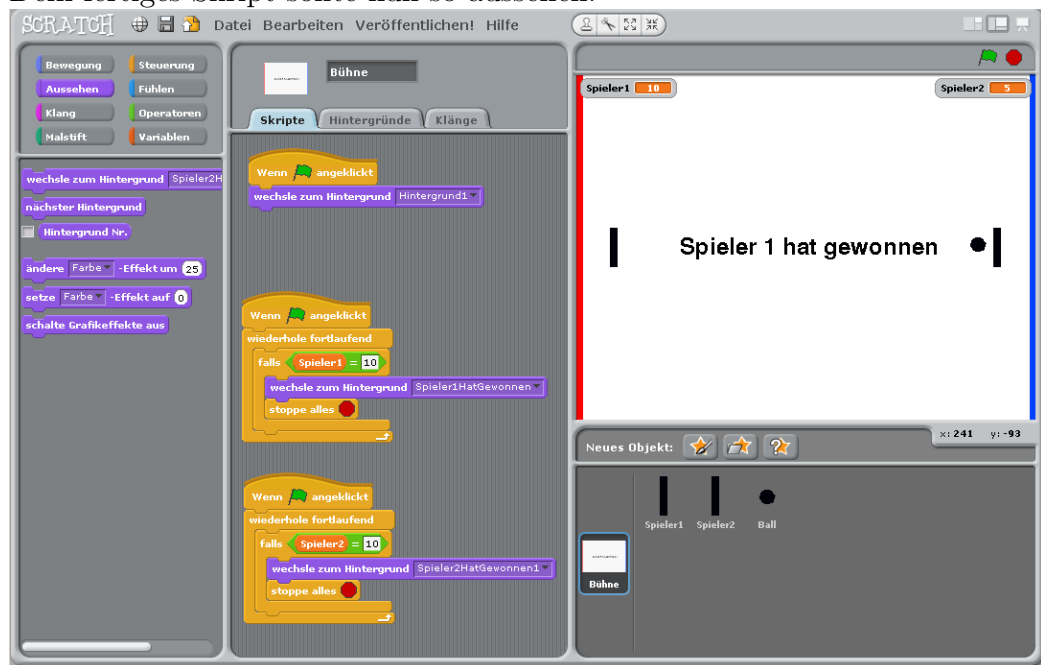


Soweit so gut! Nun muss das Spiel nur noch enden, wenn ein Spieler 10 Punkte erreicht. Wenn dies geschieht soll der Hintergrund so gewechselt werden, dass der Spieler angezeigt wird, der gewonnen hat. Du erreichst dies, indem du ein Skript für den Hintergrund erstellst. Dieses initialisiert den Hintergrund bei Spielstart und wechselt diesen sobald ein Spieler 10 Punkte erreicht:

19. Wähle die Bühne und füge dem Skript-Register die Kachel *Wenn grüne Fahne angeklickt* ein
20. Füge an diese Kachel die Kachel *wechsle zu ...* ein und wähle in der Liste den *Hintergrund1*
21. Nun füge erneut die Kachel *Wenn grüne Fahne angeklickt* ein
22. Füge an diese eine Schleife mittels der Kachel *wiederhole fortlaufend*
23. In die Schleife fügst du nun eine Kachel *falls*.
24. Als Bedingung fügst du nun die Operator-kachel *=* ein. Den einen Wert des Operators setzt du auf *10* für den anderen verwendest du die Variable *Spieler1*. Hierdurch wird der *falls*-Block nur betreten, wenn die Variable *Spieler1* den Wert *10* besitzt
25. In den *falls*-Block fügst du nun die Kachel *wechsle zu ...* ein und wähle in der Liste den Hintergrund *Spieler1HatGewonnen*

26. Füge an diese Kachel die Kachel *stoppe alles, um das Spiel anzuhalten*
27. Dupliziere nun den gesamten Block ab der Kachel *Wenn grüne Fahne angeklickt*
28. Verwende in dem neuen Block in der *falls*-Bedingung die Variable *Spieler2*
29. Passe im *falls*-Block den anzuzeigenden Hintergrund an, so dass der Hintergrund *Spieler2HatGewonnen* angezeigt wird.

Dein fertiges Skript sollte nun so aussehen:



Nun ist es Zeit das Spiel auszuprobieren.

## 5.5 Erweiterungen

Das Pong-Spiel ist nun fertig; dennoch kannst du es erweitern:

1. Versuche das Programm so zu erweitern, dass der Ball schneller wird sobald ein Spieler den Ball getroffen hat. Du brauchst dafür eine weitere Variable, die die Geschwindigkeit des Balls enthält.
2. Die Auslinien sind durch die rote und blaue Linie im Hintergrund dargestellt. Gezählt wird, wenn der Ball die rote oder blaue Farbe berührt. Überlege, wie du das Programm umbauen müsstest, wenn die Auslinien

selbst Objekte sind. Diese Objekte sollen merken können sobald sie der Ball trifft und die Punkte entsprechend anpassen.

3. Nachdem die Auslinien Objekte geworden sind, kannst du aus dem Tennisspiel ein Fußballspiel machen. Hierzu brauchst du nur die Auslinien auf Torlänge zu kürzen.