




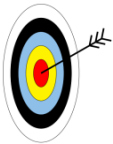
Fühlen und Variablen





Während des gesamten Moduls führen euch die Arbeitsblätter durch die Programmerstellung und geben euch Hinweise.

Achtet dabei einfach auf folgende Symbole, die..

- × euer Arbeiten strukturieren und Teilziele aufzeigen, 
- × euch Hilfen geben, Wichtiges, Schwieriges, etc. kennzeichnen und erklären 
- × Arbeitsaufträge und Aktionen beinhalten! 



Gruppenpuzzle: Fühlen und Variablen

Hier lernt ihr mit dem Programm  , wie ihr *Figuren* (im Beispiel die Katze ) beibringt etwas zu **fühlen** und wofür man in diesem Zusammenhang die **Variablen** braucht.
Neugierig? Na dann los!



Fühlen mit Hilfe von Variablen:

Die Katze soll zunächst warten bis sie berührt wird und daraufhin nach eurem Namen fragen, den ihr dann auch einträgt! Das Ganze jetzt Schritt für Schritt 😊




Dafür legen wir erst einmal die **Startbedingung** fest. Hierzu wählt ihr im Menü „Skripte“ aus den **Ereignissen** die Startbedingung „Wenn-grüne-Fahne-angeklickt“ und zieht sie einfach rechts daneben, in das freie Feld, eure **Programmiersoberfläche**.

Start/Stop




Programmiersoberfläche

Fühlen und Variablen

Die  soll auf euch aufmerksam werden, sobald ihr sie mit dem **Mauszeiger berührt**.

Dazu sucht ihr euch das folgende Puzzleteil unter dem Menüpunkt *Skripte* im Bereich


Steuerung: 

Aufgrund dieses Puzzleteils **wartet** die  solange **bis eine Bedingung** –die ihr noch in das Puzzleteil einfügen könnt- **erfüllt ist**. In unserem Beispiel **wartet** sie also so lange, **bis** ihr sie mit dem **Mauszeiger berührt**.

Hierfür verwendet ihr folgende **Bedingung**, die ihr unter **Fühlen** findet:



Hier könnt ihr den Mauszeiger auswählen.

Sobald ihr die  berührt habt, soll sie jetzt **die Frage** stellen, wer sie denn streicheln möchte. Verändert dazu das folgende Puzzleteil aus **Fühlen** dementsprechend:

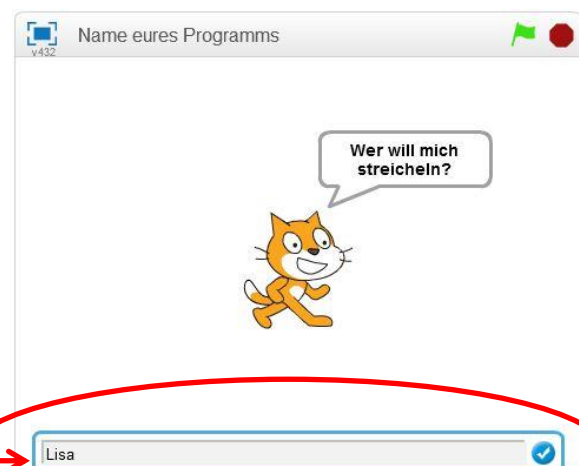


Ihr könnt hier in das Puzzleteil die Frage hineinschreiben.

So soll euer Programm bisher aussehen.



Hier könnt ihr jetzt euren Namen eintragen!



Fühlen und Variablen



Damit sich die 🐱 auch *merkt*, wie oft ihr sie gestreichelt habt (also angeklickt habt) legen wir jetzt einen „Merkspeicher“ an, eine sogenannte **Variable**. Wir geben ihr einen passenden Namen, wie z.B. „Streichelzähler“.

Durch Häkchen setzen wird der *Streichelzähler* - eure Variable - auf der Bühne sichtbar!

The image shows the Scratch 'Neue Variable' dialog box with 'Variablenname: Streichelzähler' and 'Für alle Figuren' selected. To the right, a script block contains: 'setze Streichelzähler auf 0', 'ändere Streichelzähler um 1', 'zeige Variable Streichelzähler', and 'verstecke Variable Streichelzähler'. A red arrow points from the 'Daten' category to the dialog box, and another red arrow points from the dialog box to the script block.

Jedes Mal wenn wir auf die grüne klicken, soll die 🐱 wieder bei „0“ anfangen zu zählen. Deshalb setzen wir in diesem Programmteil **die Variable** auf 0, so weiss die 🐱 ab wann sie anfangen soll zu zählen.

Damit dies in eurem Programm funktioniert, **setzt ihr den Streichelzähler auf 0**. Das passende Puzzleteil findet ihr unter **Daten**. Zieht es einfach unter eure vorhandenen Puzzleteile in eure Programmieroberfläche.

Wenn euer Programm so aussieht, habt ihr alles richtig gemacht! Auf zum zweiten Teil!



The image shows a Scratch script block with the following steps: 'Wenn grüne Flagge angeklickt', 'warte bis Mauszeiger berührt?', 'frage Wer will mich streicheln? und warte', and 'setze Streichelzähler auf 0'.

Fühlen und Variablen



Was genau tut euer Programm bisher? Was kann ich **fühlen**?
Was macht die **Variable**?
Erklärt den bisherigen Programmablauf mit euren eigenen Worten!



1.

2.

3.

4.

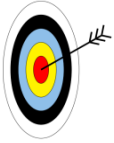


Zur Erinnerung:

Wir wollen dass die Katze merkt, dass wir sie streicheln und ab wann wir dies tun.

Zusätzlich lassen wir unsere Zählungen auf der Bühne mit Hilfe unserer erstellten Variablen (alias „Streichelzähler“) anzeigen.

Fühlen und Variablen




Zählen mit Hilfe von Variablen:

Nachdem wir unsere **Variable** (den Merkspeicher) erstellt haben, wollen wir ihn jetzt auch benutzen. Die Katze soll jetzt zählen, bzw. sich merken, wie oft sie gestreichelt wurde. Das machen wir in einem **zweiten Programmteil**, den wir mit der **Startbedingung** „Wenn-ich-angeklickt-werde“ beginnen.


Dieser Programmteil wird dann jedes Mal aufgerufen, wenn wir die Katze anklicken, also **berühren**.



Aus dem Menü Skripte wählt ihr aus den **Ereignissen** „Wenn-ich-angeklickt-werde“, aus. Und zieht es unter euer erstes Programm.

Zieht nun „ändere-Streichelzähler-um-1“ aus **Daten** eurer erstellten **Variable** unter diesen Block, um die  zählen zu lassen. Dies sollte jetzt wie folgt aussehen:



Als nächstes bringen wir der  bei sich zu bedanken, sobald sie 5-mal gestreichelt wurde.

Hierfür verwenden wir die „falls-dann“- **Anweisung** aus der **Steuerung**.

Falls die Katze 5-mal gestreichelt wurde, **dann** soll sie sich bedanken.



In die **Anweisung** zieht ihr jetzt noch die **Bedingung**, die erfüllt sein soll. In unserem Fall ist das, **falls die Katze 5-mal gestreichelt worden ist**, also der **Streichelzähler gleich 5** ist.

Wir verwenden dazu den **Vergleichs-Operator**:





Er vergleicht, ob der linke Wert gleich dem Rechten ist.

Dies sieht dann so aus:



Fühlen und Variablen




Damit sich die  bei euch fürs Streicheln bedankt, müssen wir in die „falls – dann“ – **Anweisung** nur noch dieses Puzzleteil einsetzen: . In das weiße Feld schreibt ihr dann: „Danke fürs Streicheln!“
Das Puzzleteil findet ihr unter **Aussehen**.
Klar soweit? 😊 Dann testet euer Programm!




Hat euer
Programm
funktioniert?

Wozu habt ihr im ersten Teil
eigentlich euren Namen eingegeben?

- Genau, weil die Katze sich jetzt namentlich bei euch bedanken soll!
- Doch woher weiß die Katze euren Namen noch?
- Weil sie ihn in der bereits vorhandenen **Variable**  gespeichert hat und deshalb jederzeit darauf zugreifen kann.



Verbindet deshalb jetzt noch schnell mit Hilfe eines **Operators** die  mit dem Satz den ihr sagen wollt und dann sollte das Ganze so aussehen:

Sieht der
zweite
Programmteil
so
bei euch aus?



Herzlichen Glückwunsch!
Ihr habt euer Ziel erreicht!

