



FRANK
ENGELN

Scratch me if you can!

Programmieren mit **Scratch 1.4**, **Lego© WeDo** & **Picoboard**

Ideenbuch

Programmieren mit Scratch 1.4, Lego© WeDo & Picoboard

Kinder oder Erwachsene ohne Computerkenntnisse, können mit Scratch motivierende Programmiererfahrungen machen, weil Schrift nicht die einzige Zugangsmöglichkeit ist, sehr wenig getippt werden muss (für einfache Programme gar nicht), keine Programmierbefehle gelernt werden müssen und viele potentielle Fehler (zum Beispiel Syntax-Fehler) gar nicht möglich sind.

©2010 von Frank Engeln,
Bursibantstraße. 3, D-48429 Rheine Germany
Alle Rechte vorbehalten.
Autor: Frank Engeln

Mein Dank gilt:

VS-Möbel
Hochhäuser Straße 8
97941 Tauberbischofsheim

für die freundliche Unterstützung.

Scratch me if you can!

Einführung

Übergeordnetes Ziel des Informatikunterrichts ist es, Schülerinnen und Schüler bestmöglich auf ein Leben in einer Gesellschaft vorzubereiten, die immer mehr durch den Einsatz von Informations- und Kommunikationstechnologien sowohl im privaten als auch im beruflichen Bereich geprägt ist. Da in allen gesellschaftlichen Bereichen die Nutzung und der Umgang mit Informatiksystemen unumgänglich geworden sind, ist eine solide informatische Grundkenntnis die Voraussetzung für die Teilnahme am beruflichen und gesellschaftlichen Leben.

Die Firma Lego hat schon vor längerer Zeit einen neuen, spielerischen Einstieg in die Informatik gewagt. Der LEGO-„Computer“ (Lego Mindstorms) eignet sich sehr gut als Einstieg für die objektorientierte Programmierung in der Schule. Die Planungen zum Unterricht mit LEGO Mindstorms beziehen sich auf den Informatikunterricht der Sekundarstufe I und II. In vielen Bundesländern wird in den Lehrplänen das Thema "Messen, Steuern, Regeln" empfohlen. In diesem Script soll ein Einstieg **vor** Lego Mindstorms gezeigt werden. Welche Vorteile bietet Scratch in Kombination mit Lego? Scratch ist kostenlos, die Programm CD zu Lego WeDo sehr teuer. Wer schon einige Legoteile besitzt, kann Komponenten wie die Sensoren und den USB-Hub von Lego auch einzeln kaufen.

Eine sinnvolle Ergänzung ist das Picoboard sein: <http://www.picocricket.com/picoboard.html>
Für den Einsatz im Informatikbereich der Sekundarstufe I sprechen die einfache Bedienung und der schülergerechte Aufbau der Benutzeroberfläche. Das Erstellen von Spielen ist sehr motivierend für Schüler und Schülerinnen.

„Die Themen sind aus dem Erfahrungsbereich der Schüler und decken wesentliche Inhalte aus den aktuellen Lehrplänen verschiedener Fächer ab.“

Technik/Sachunterricht:

Einfache Mechanismen, Zahnräder, Flaschenzug, Hebel, Übertragung von Bewegungen und Kräften, Planen und Konstruieren von Funktionsmodellen

Informationstechnik/Informatik:

Umgang und Handhabung von Programmiersoftware, Programmieren von Abläufen

Mathematik:

Messen von Zeit und Entfernung, Addition, Subtraktion, Multiplikation, Division, Schätzen usw.

Sprache und Rechtschreibung:

Schreiben von Erzählungen und Berichten, Geschichten erzählen, erklären, interpretieren, befragen „ aus: *LEGO® Education WeDo™ Quick Start-Anleitung, www.legoeducation.com*

Inhalt

Was ist Scratch?	5
Was ist WeDo?	6
Arbeiten mit Scratch und Lego WeDo	8
Der Umgang mit Scratch	9
Die Objektverwaltung – der Editor	9
Struktogramme	9
Arbeiten in der Grundschule (ab Klasse 4)	13
Scratch & WeDo	17
Unterrichtseinheit Klasse 7 RS (2 Stunden pro Woche)	17
Wir basteln uns einen Joystick – Interaktives Spiel	18
Kleiner Synthesizer	18
Autorennen – Ein Projekt für eine Klasse?	19
Zufallsgrafiken	22
Fraktale	23
Wir bauen eine Wii	24
Picoboard	26
Pico-Schlagzeug	29
Scratch, WeDo und Picoboard	31
Umweltmessgeräte entwickeln (Skala per Motor)	31
Kühlschrankalarm und Alarmanlage	33
Ein Feuermelder bauen	33
Das gesteuerte Energiehaus – Ein Gruppenprojekt	34
Adapterkabel bauen	35
Lichteffekte oder Motorenleistung mit einem Potentiometer steuern	36
LEGO 9686 Naturwissenschaft und Technik (inkl. Motor)	37
Mit Scratch ein Fahrzeug steuern	38
Ideen für den Einsatz in Nicht-Informatik-Fächern	39
Mathematik	39
Links	40
Referenz	42

Was ist Scratch?

Die Programmiersprache Scratch wurde entwickelt, um jungen Leuten das Programmieren näher zu bringen. Scratch ist kostenlos, sehr einfach zu installieren (Quelle: <http://www.scratch.mit.edu>) und läuft auch auf einem USB Stick. Es vergehen nur wenige Minuten bis zum ersten selbst erstellten Programm.

Scratch ist eine visuelle Programmiersprache, entwickelt am Media Lab des Massachusetts Institute of Technology (MIT). Die Programmierung erfolgt per Drag and Drop, bunte Blöcke werden wie Puzzlesteine kombiniert. Die Programmierung ist objektorientiert und ereignisgesteuert.

SCRATCH



Scratch ist so einfach zu erlernen, dass ein Einsatz bereits in der Grundschule möglich und sinnvoll ist, aber nicht nur zum Programmieren an sich sondern aufgrund der Multimediaelemente beispielsweise auch in Bildnerischer Erziehung oder Musik.

«Scratcher» erschaffen mit einfachsten Mitteln virtuelles Leben. Dies stößt auf Anklang, wie die stetig ansteigenden Scratch-Fangemeinden im Internet zeigen.

Fertige Projekte können auf die Homepage von Scratch geladen werden, andere können die Projekte online ansehen, in Seiten einbinden oder herunterladen und weiterentwickeln!

Scratch ist für den Programmieranfang, für den Einstieg gedacht und keine hochwertige Programmiersprache.

Download: http://info.scratch.mit.edu/Scratch_1.4_Download (ca. 33 Mb)

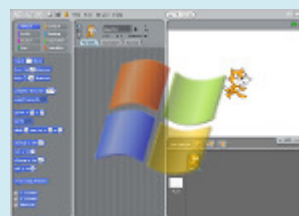
Scratch Installer for Mac OS X
Compatible with Mac OSX 10.4 or later

MacScratch1.4.dmg



Scratch Installer for Windows
Compatible with Windows 2000, XP, Vista

ScratchInstaller1.4.exe



Was ist WeDo?

LEGO Education WeDo ist ein neues Lernkonzept von LEGO Education. Mit WeDo® soll die Robotik endgültig Einzug in die Klassenzimmer halten und es nach Angaben von LEGO® Lehrern ermöglichen, 7- bis 11-jährigen Schülern die entsprechenden Grundlagen auf intuitive und spielerische Weise zu vermitteln.

Im Prinzip handelt es sich dabei um eine Sammlung von 158 LEGO®-TECHNIC-Elementen, einem PF-Motor und zwei Sensoren, einer USB-Verbindung zu einem Computer sowie zugehöriger Software mit einer einfachen graphischen Programmierumgebung, Anleitungen und Aufgaben.



Das Basisset (siehe Foto) kostet ca. 125,00 Euro. Allerdings fehlt die Software. Die Originalsoftware kostet ca. 65,00 Euro. Sehr teuer für eine Einzellizenz. Die Komponenten könne aber auch einzeln gekauft werden.

Die Bauanleitungen für WeDo findet man als Download im Internet unter:

www.nxt-roboter.de/wedo/

Fliegender Vogel, Flugzeug, Fußballer, Kreisel, Krokodil, Löwe, Riese, Schiff, Torwart, Trommelnder Affe, Zuschauer, Zwei Vögel

	<p>LEGO 9581: USB-Hub für WeDo®</p> <p>Der LEGO USB Hub ist die Schnittstelle zwischen Computer, Software und der Hardware. Der Hub hat zwei Anschlüsse an die sowohl Motoren und Sensoren angeschlossen werden können. Über den Hub werden die Modelle gesteuert und mit Spannung versorgt.</p> <p>Der Hub wird automatisch von der Software bei bestehender Verbindung mit dem Computer erkannt. Ca. 40,00 €</p>
	<p>LEGO 9583: WeDo® Bewegungssensor</p> <p>Der WeDo Bewegungssensor kann Gegenstände im Abstand von bis zu 15 cm erkennen. ca. 15,00 €</p>
	<p>LEGO 9584: Kippsensor/Tiltensor für WeDo®</p> <p>Der WeDo Kippsensor unterscheidet Richtungsänderungen in sechs verschiedenen Zuständen. ca. 15,00 €</p>
	<p>LEGO 8883: Power Functions M Motor</p> <p>Mit diesem mittelstarken und mittelgroßen M-Motor kannst man LEGO Kreationen in Bewegung setzen.</p> <p>Es gibt die Motoren auch in S (klein) und XL (groß). Ab ca. 10,00 €</p>
	<p>Die Verlängerungskabel sind in 20 cm und 50 cm Länge erhältlich. (nicht in WeDo enthalten)</p>

Quelle: www.nxt-roboter.de

Um mit Schülern mit Scratch zu programmieren benötigt man den USB-Hub und einen Sensor. Alle andere Hardware kann optional ergänzt werden. Wer keine Lego Bauteile besitzt ist mit dem WeDo Kasten sehr gut ausgerüstet.

Arbeiten mit Scratch und Lego WeDo

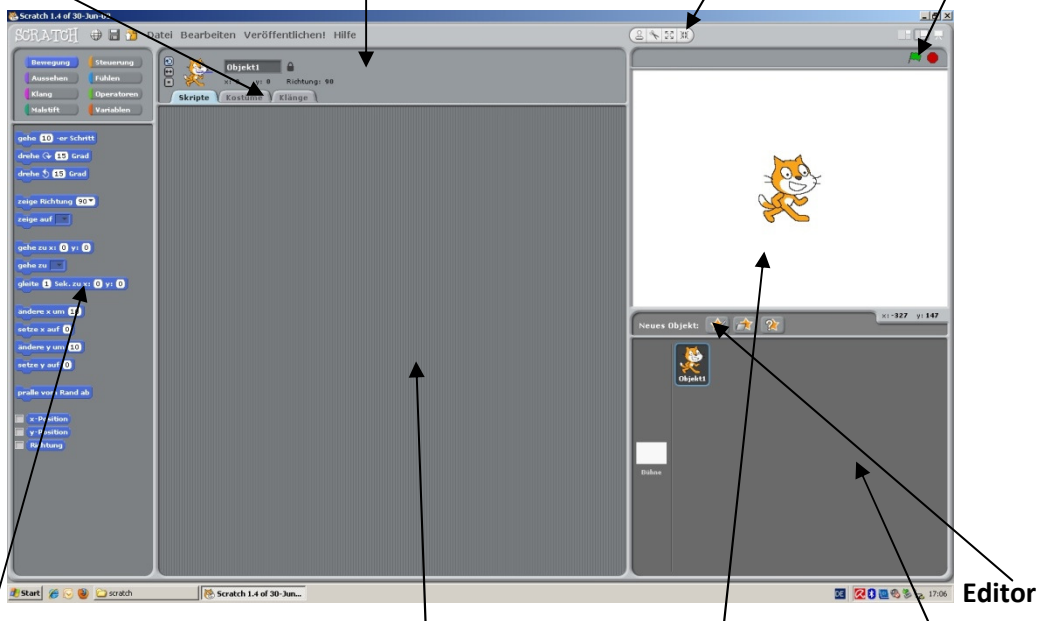
Der Aufbau der Programmierumgebung

Infos zum aktuellen Objekt und Rotationsmodus

Werkzeugbereich

Grüne Flagge

Reiter



Blockpalette

Programmebereich

Bühne

Objektliste

Auf der *Bühne* (480x360 Einheiten) erwachen die Geschichten, Animationen und Spiele zum Leben. In der *Objektliste* stehen die Vorschaubilder der Projekte, der Name und die Anzahl der Programme.

Im Extras- bzw. Language-Menü lässt sich die Beschriftung der „Programmierblöcke“ auf Deutsch Umstellen.

Mittlerer oder rechter Mausklick auf das Skriptfenster hilft beim Aufräumen.

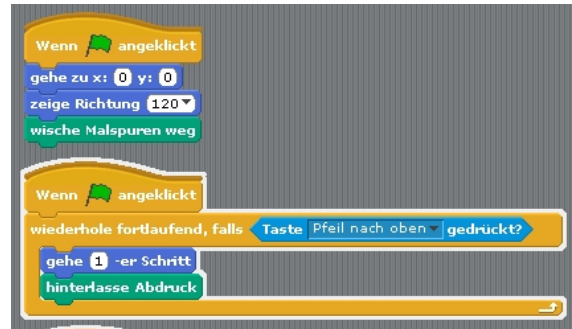
Der mittlere oder rechte Mausklick auf die Blöcke liefert ein Hilfefenster.

Ein mittlerer oder rechter Mausklick auf die Skripte bietet auch die Möglichkeit, Skripte (oder Teile davon) zu kopieren. Zieht man mit der Maus ein Skript auf das Icon eines anderen Objekts im Objekt-Fenster, wird das Skript auch zu diesem Objekt hinzugefügt.



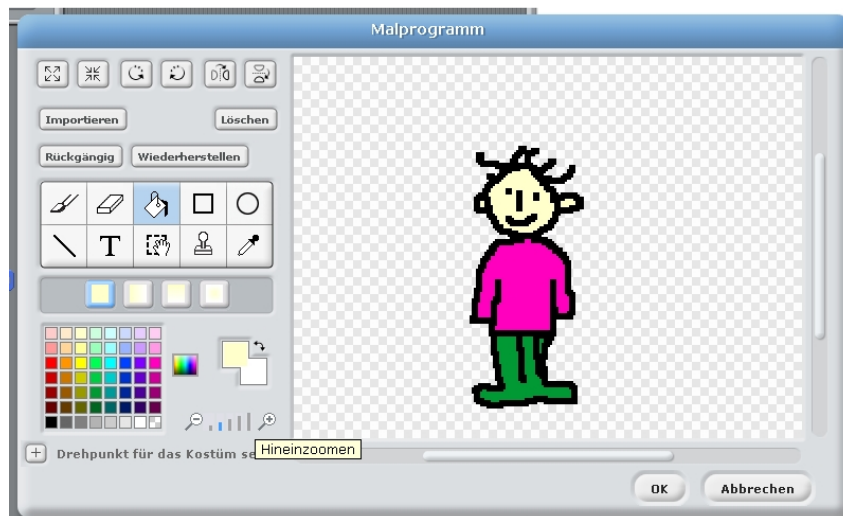
Der Umgang mit Scratch

Um ein Programm zu erstellen zieht man einen *Block* aus der *Blockpalette* in den Programmierbereich (drag & drop). Im Baukastensystem findet man Steuerungs- und Kontrollstrukturen, aus denen zusammengesetzt ein Programm entsteht. Um einen Baustein zu löschen zieht man diesen einfach in die Blockpalette zurück. Neben den Standardbefehlen „falls“, „wiederhole“, „fortlaufend“ und „warte bis“, lässt sich auch das Verhalten eines Objektes durch „fühlen“ (Sensoren) beeinflussen. Die einzelnen Baukastenmodule „Bewegung, Steuerung, Aussehen, Fühlen, Klang, Operatoren, Malstift und Variablen“ sind mit verschiedenen Farben versehen. Dadurch wird die Programmierung für SchülerInnen sehr deutlich. Um neue Kostüme zu erstellen oder um das aktuelle zu ändern, klickt man auf *Kostüme*. Scratch kann die Bildformate „*.jpg, *.gif, *.bmp und *.png“ verarbeiten.



Klicke auf den *Klänge*-Reiter um die Objekt-Klänge zu sehen. Scratch kann „*.mp3, *.wav, *.aif und *.au“ Dateien lesen. Man kann Töne auch selbst aufnehmen.

Die Objektverwaltung – der Editor



Im Grafikeditor kann man Kostüme bearbeiten oder neue Objekte erstellen. Ebenso werden in dem Editor die Bühnenbilder bearbeitet oder erstellt. Man kann selbst zeichnen oder ein Objekt importieren.

Struktogramme

In der Informatik spricht man bei der Beschreibung von Methoden gerne von Algorithmen. Es gibt verschiedene Möglichkeiten, Algorithmen übersichtlich darzustellen.

Eine davon sind sogenannte Struktogramme. Um Struktogramme zu erstellen gibt es im Internet unter <http://eii.ucv.cl/nessi/> einen kostenlosen Editor mit dem Namen Nessi (nach Nessi Schneiderman).

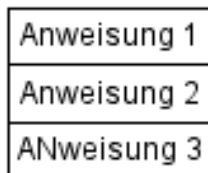
Sequenzen

Als *Sequenzen* bezeichnen wir komplexe Datentypen, die einfach eine Folge von Elementen bzw. Objekten darstellen.

Was wird durch diese Sequenz (Bild rechts) dargestellt?



Struktogramm von Sequenzen:



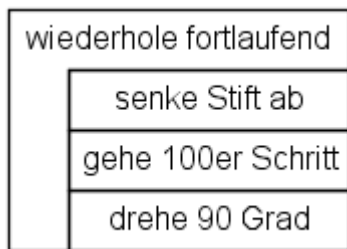
Schleifen

Eine Schleife ist eine Befehlsfolge, die mehrfach durchlaufen wird. Schleifen wiederholen einen Code-Block – den so genannten *Schleifenrumpf* oder *Schleifenkörper* – so lange, wie eine Laufbedingung gültig ist oder bis eine Abbruchbedingung eintritt. Schleifen, deren Laufbedingung immer erfüllt ist oder die ihre Abbruchbedingung niemals erreichen, und Schleifen, die keine Abbruchbedingungen haben, sind Endlosschleifen.

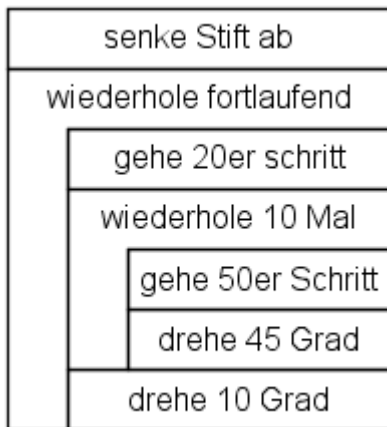
Folgende Schleifen-Bausteine gibt es in Scratch:

	Baustein für Endlosschleifen
	Baustein für Wiederholungen mit fester Anzahl
	Baustein für Wiederholung bis eine bestimmte Bedingung erfüllt wird und die Wiederholung dadurch abgebrochen wird
	Baustein für Endlosschleifen, die abgebrochen wird, falls ein Ereignis eintritt.

Struktogramm für eine Schleife. Hier wird eine Sequenz endlos wiederholt.



Verschachtelte Schleifen können so aussehen:

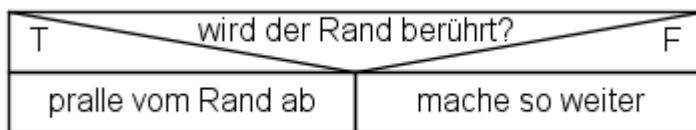


Aufgabe:

Entwerft Struktogramme für das Zeichnen eines Quadrats und eines Oktaeders. Lasst dann ein „Objekt“ die Figuren zeichnen.

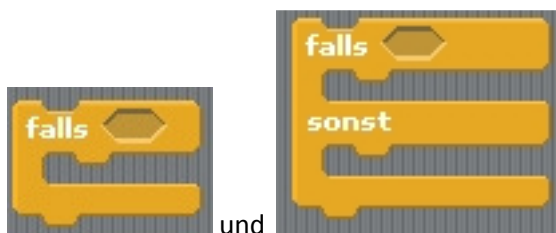
Bedingte Anweisungen

Eine **Bedingte Anweisung** ist ein Programmabschnitt, der nur unter einer bestimmten Bedingung ausgeführt wird. Eine **Verzweigung** legt fest, welcher von zwei (oder mehr) Programmabschnitten, abhängig von einer (oder mehreren) Bedingungen, ausgeführt wird. Hier: Wird der Rand berührt? Ja= pralle vom Rand ab; nein=mache so weiter.



Bedingte Anweisungen und *Verzweigungen* bilden, zusammen mit den Schleifen, die Kontrollstrukturen der Programmiersprachen.

In Scratch gibt es folgende Bausteine:



Block einer unvollständigen und vollständigen Alternative mit möglicher Bedingung.



Variablen

Die Variable bildet das „Gedächtnis“ eines Programmes. Sie ist einfach ein Behälter für Rechnungsgrößen („Werte“), die im Verlauf eines Rechenprozesses auftreten. Bei einem Tennisspiel können hier die Treffer des Schlägers mit dem Ball gezählt werden und nach 15 Punkten ist das Spiel beendet.

Erstellt man eine Variable, muss man sich überlegen, wer alles auf die Variable Zugriff benötigt – wenn nur ein Objekt die Variable benutzen soll, beschränkt man sich auf den Zugriff für nur dieses Objekt (nur für dieses Objekt). Sollen alle Objekte die Variable verändern und benutzen können, verwendet man globale Variablen. (für alle Objekte)



Beispiel: Die Katze soll 10 Mal hin und her springen. Nach 10x springen soll das Programm stoppen. Hier bauen wir einen Zähler (Variable) ein. Damit das Programm nicht zu schnell abläuft habe ich ein „warte 1 Sek.“ eingebaut.

Nachrichten senden und empfangen

Durch Interaktion mit einer Figur/eines Objekts bzw. Objektes können weitere Aktionen bei den anderen Objekten ausgelöst werden. Dazu soll eine Nachricht von einem Objekt zu einem anderen verschickt werden.

Bausteine zum Senden und Empfangen von Nachrichten.

In Scratch wird die Verhaltensweise eines Objektes durch ein Script beschrieben.

Das Ausführen dieser Verhaltensweise kann veranlasst werden durch:

- Erhalten einer bestimmten Botschaft
- Tastendruck (z.B. Leertaste)
- Anklicken des Objektes (Objekt1)
- Start des Programms (Grüne Fahne)



Arbeiten in der Grundschule (ab Klasse 4)

Unterrichtsbeispiel – Schnappi, das Krokodil

Schnappi, das Krokodil - Lehrerkommentar

Schüler bauen und programmieren ein mechanisches Krokodil, das Geräusche macht und dessen Maul von einem Motor gesteuert wird. Das Programm soll den Kiefer öffnen und schließen.

Ziele

Wissenschaft

Beobachte die Kraftübertragung der Bewegung und der Übertragung der Energie durch den Motor. Identifizieren der Riemenscheiben und Riemen und der Verlangsamung Mechanismus bei der Arbeit im Modell. Betrachte die Bedürfnisse von lebenden Tieren.

Technik

Erstelle ein programmierbares Modell um zu zeigen, wie du dein Wissen mit Hilfe von digitalen Werkzeugen und technischen Systemen umsetzen kannst.

Engineering

Erstelle und teste die Krokodil-Bewegung. Erhöhe das Verhalten des Krokodils, indem du den Bewegungsmelder und die Sound-Programmierung mit der Bewegung des Krokodilmauls abstimmt.

Mathematik

Verstehe die Abhängigkeit der Entfernung zwischen einem Objekt (Krokodil) und dem Bewegungssensor.

Bekomme ein Verständnis im Umgang mit Zahlen und Werten (Leistung des Motors, Länge der Drehung es Motors...)

Sprache

Berichte über Krokodile mit Hilfe des Krokodil-Modells. Erkläre die Technik des Modells. Bringe neue Ideen ein, wie das Modell erweitert werden kann.

Trage in mündlicher oder schriftlicher Form mit dem entsprechenden Wortschatz vor.

Wortschatz

Gummiband, Bewegungssensor und Antriebsscheibe, Sound-Wiedergabe, Motor und Programm.

Andere Materialien

Optional für die Erweiterung im Unterricht: Basteln eines Krokodils mit Papier, Pappe, Gras, Steinchen.



Bauanleitung dieses Krokodils unter: http://www.nxt-roboter.de/krokodil_bauanleitung.pdf

Krokodile wirken auf uns oft unheimlich und gefährlich: Die faszinierenden Reptilien sind sehr geschickte Jäger. Doch die meisten Arten sind für den Menschen harmlos.

Mögliche Unterrichtsinhalte zur Erarbeitung

Schnappi, das kleine Krokodil ist formal und inhaltlich ein Kinderlied. Es verwendet – gesungen von einem Kind mit noch unsteter Stimme – die typischen textlichen und melodiösen Merkmale. Der Text ist einfach strukturiert und verwendet die Stilfigur des „mehrere-Anläufe-mit-falschen-Vokalen“ („Schni-schna-schnappi“ wie in „Bi-Ba-Butzemann“), die zu den so genannten Klangfiguren gehört, unterlegt durch eine pentatonische Melodie. In einfacher Reim Form wird von einem gerade geschlüpften Nilkrokodil erzählt, das niedlich, neugierig und frech seine Umgebung erkundet. Deshalb darf dieses Kinderlied nicht mit den – ebenfalls nicht allzu hoch anzusetzenden - Ansprüchen (der Erwachsenen) an Popmusik oder Schlager gemessen werden. (aus www.wikipedia.de)

Liedertext singen: **Schnappi, das kleine Krokodil**

Ich bin Schnappi das kleine Krokodil
Komm' aus Ägypten das liegt direkt am Nil
Zuerst lag ich in einem Ei dann schnei schna schnappte ich mich frei
Schni schna schnappi - schnappi schnappi schnapp
Schni schna schnappi - schnappi schnappi schnapp....

Der ganze Text ist im Internet zu finden, z.B. unter <http://www.langeneggers.ch/Songs/schnappi.htm>

Links zum Thema Krokodil:

http://www.kidsweb.de/quis/krokodil/kroko_2005.html - Krokodil - Quiz (Sachunterricht)

<http://www.schulbilder.org/de-malvorlagen-ausmalbilder-foto-krokodil-i3053.html> - Malvorlage (Ausmalbild)

Eine weitere Möglichkeit ist es ein Hintergrundbild in Scratch malen lassen. Man kann natürlich auch aus dem Internet ein Hintergrundbild importieren.

Beispiel:



Um ein Programm für Grundschüler (4. Klasse) zu erarbeiten muss man kleinschrittig vorgehen. Erst die Werte lesen lassen. In welche Richtung dreht der Motor, wie kann ich ihn anders herum drehen lassen. Was bedeutet etwas wiederholen, wenn-dann Bedingungen, Werte aufschreiben etc.

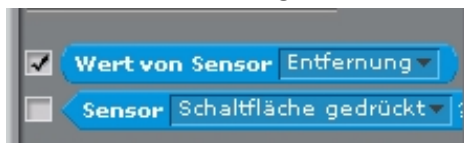
Das Krokodil-Programm mit Scratch

Das Maul soll geöffnet sein. Wenn ein Gegenstand oder die Hand näher als der Wert „20“ angezeigt wird, soll das Krokodil „knurren“ und das Maul schließen und nach 0,3 Sekunden wieder öffnen.

1. USB anschließen, Entfernungssensor und Motor anschließen
2. Scratch starten
3. Auf Fühlen klicken



4. Sensorwert „Entfernung anklicken“



5. Wert vom Sensor ablesen. Die Änderung beobachten



- 6.
7. Programm erstellen (es gibt mehrere Möglichkeiten)
8. Programm testen

Programmbeispiel:



Der Fußballer

2 Schüler an 2 Rechnern. Fußballer und Torwart.



```

Wenn [Flagge] angeklickt
wiederhole fortlaufend
  falls Taste [Pfeil nach rechts] gedrückt?
    ändere x um 4
    setze Motornrichtung auf in diese Richtung
    schalte Motor für 1 Sekunden an
  falls Taste [Pfeil nach links] gedrückt?
    ändere x um -4
    setze Motornrichtung auf in jene Richtung
    schalte Motor für 1 Sekunden an
  
```

Vorbereitung

Die Kinder lieben die Fußball-Sticker der Stars. Warum nicht selbst alternative Bilder herstellen? Damit kann man Geld sparen und dennoch Spaß haben. Passende Bilder dazu finden die Kinder im Internet. Eine Verknüpfung des Sport-Themas mit dem Sachunterricht kann insbesondere durch die Schwerpunkte Landes- und Menschenkunde hergestellt werden. Welche Länder haben sich für die WM 2010 qualifiziert? Wo liegen sie? Wie viele Einwohner haben sie? Wie groß ist das Land?

Die einfache Torwartsteuerung (oben) ohne Picoboard

Welche Flaggen und KFZ-Kennzeichen haben die beteiligten Länder? Was essen die Menschen dort? Ausmalbilder von Flaggen und Fußballern und Fußballerinnen gibt es genug im Internet. Info unter: <http://de.fifa.com/worldcup/index.html>

Kicken auf Zuruf mit dem Picoboard – „Schuss - ?“
Torwart mit dem Regler steuern

Ein mögliches Torwartprogramm mit der Reglersteuerung des Picoboard

Für den Spieler:

```

Wenn [Flagge] angeklickt
zeige Richtung 90
wiederhole fortlaufend
  falls Wert von Sensor [Regler] < 45
    ändere x um 4
    setze Motornrichtung auf in jene Richtung
    setze Motorkraft auf 80
    schalte Motor für 0.3 Sekunden an
    schalte Motor aus
  falls Wert von Sensor [Regler] > 55
    ändere x um -4
    setze Motornrichtung auf in diese Richtung
    setze Motorkraft auf 80
    schalte Motor für 0.3 Sekunden an
    schalte Motor aus
  pralle vom Rand ab
  zeige Richtung 90
  falls Sensor [Schaltfläche gedrückt] ?
    setze x auf 0
    setze y auf 0
    stoppe alles
  
```

```

Wenn Taste [Leertaste] gedrückt
stoppe alles
  
```

```

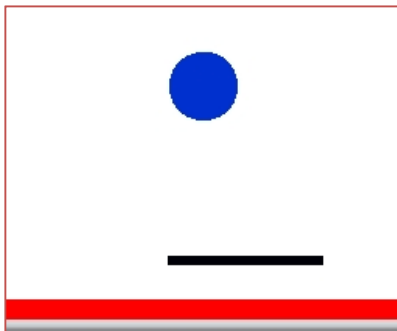
Wenn [Flagge] angeklickt
wiederhole fortlaufend
  falls Wert von Sensor [Klang] > 10
    setze Motorkraft auf 100
    setze Motornrichtung auf in diese Richtung
    schalte Motor für 1 Sekunden an
    schalte Motor aus
  
```

Auf Zuruf schießt der Fußballer.

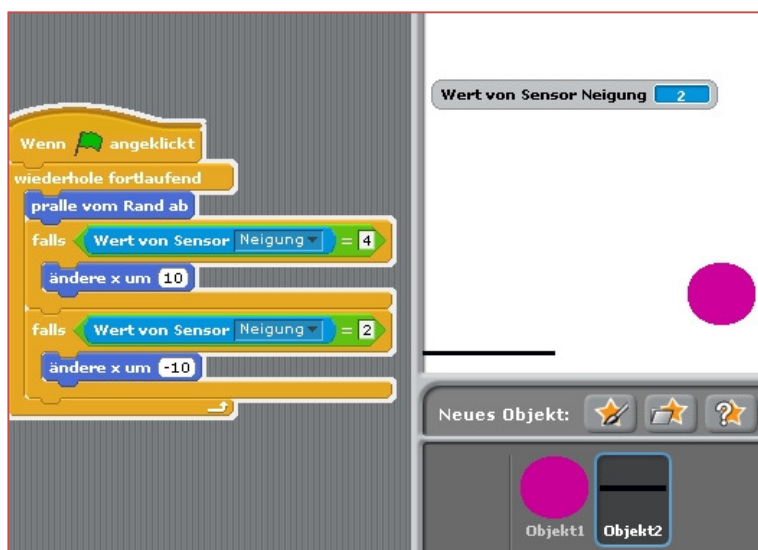
Scratch & WeDo

Unterrichtseinheit Klasse 7 RS (2 Stunden pro Woche)

- 1. Stunde :**
Die Schüler sollen das Programm erkunden.
- 2. Stunde:**
Das Program wird gemeinsam erforscht. Der Lehrer gibt Hilfestellungen. Somit sollen Fachbegriffe eingeführt werden.
Aufgabe: Wir zeichnen einen Ball und lassen diese auf der Bühne umherspringen, (optional: bei Berührung mit dem Mauszeiger soll sich die Farbe „Kostüm“ ändern oder ein Ton „Sound“ erklingen. Dabei soll die Geschwindigkeit und die Richtung geändert werden. Das Projekt soll gespeichert werden.
- 3. Stunde:** Zeichnen, Bühne, Objekte
Aufgabe: Das gespeicherte Projekt „ Ball“ soll die Geschwindigkeit und die Richtung geändert werden. Es soll eine Animation zum Thema „Ballsport“ erstellt werden.
- 4. Stunde:**
Wir erstellen ein einfaches „Tennis- oder Squash Spiel“. Mit den Pfeiltasten „links“ und „rechts“ soll der „Schläger“ (ein einfacher Balken)“ gesteuert werden und der Ball soll bei Berührung seine Richtung leicht ändern. Das „Aus“ ist als roter Strich unterhalb des Schlägers gezeichnet. Bei Berührung soll das Spiel zu Ende sein.



- 5. Stunde:** Die Steuerung durch die Pfeiltasten wird durch die Steuerung mit dem Kippsensor von WeDo ersetzt.



6. **Stunde:** Kontakte zählen – die Variable
Jeder Berührung mit dem Schläger soll gezählt werden. Nach 60 Sekunden soll das Spiel beendet sein.
7. **Stunde:** (eingeschoben - es war vor Weihnachten)
Ein selbsterstellter Stift soll auf der Bühne das „Haus vom Nikolaus“ zeichnen. Hier lernen die Schüler den Umgang mit dem Koordinatensystem.
8. **Stunde:**
Ein interaktives Spiel erstellen.

Wir basteln uns einen Joystick – Interaktives Spiel

Ein Taster vom NXT ergibt einen Wert (nicht gedrückt) 100 und gedrückt von 17, angeschlossen an das Picoboard.



Um den NXT Taster anschließen zu können sollte man sich eigene Adapterkabel bauen. Das ist nicht schwer und man ist flexibler. Beispiele gibt es auf Seite 30.

Bild: Taster von Lego Mindstorms NXT

Einen Joystick kann man aus Lego bauen.

Kleiner Synthesizer

Schließt man mehrere Taster an, so kann man die Kabel mit eingebauten Widerständen nehmen. Dadurch lassen sich auch 8 Taster mit unterschiedlichen Werten auslesen. Bei 8 Tönen lässt sich hier eine kleine Orgel bauen. Hat man noch mehr Taster zur Verfügung schafft man mit den richtigen Kabeln auch alle 12 Töne in einer Oktave. Ein kleiner Synthesizer. Statt der Lego-Taster kann man auch einfache (günstigere) Taster aus dem Elektronikversand nehmen.



Autorennen – Ein Projekt für eine Klasse?

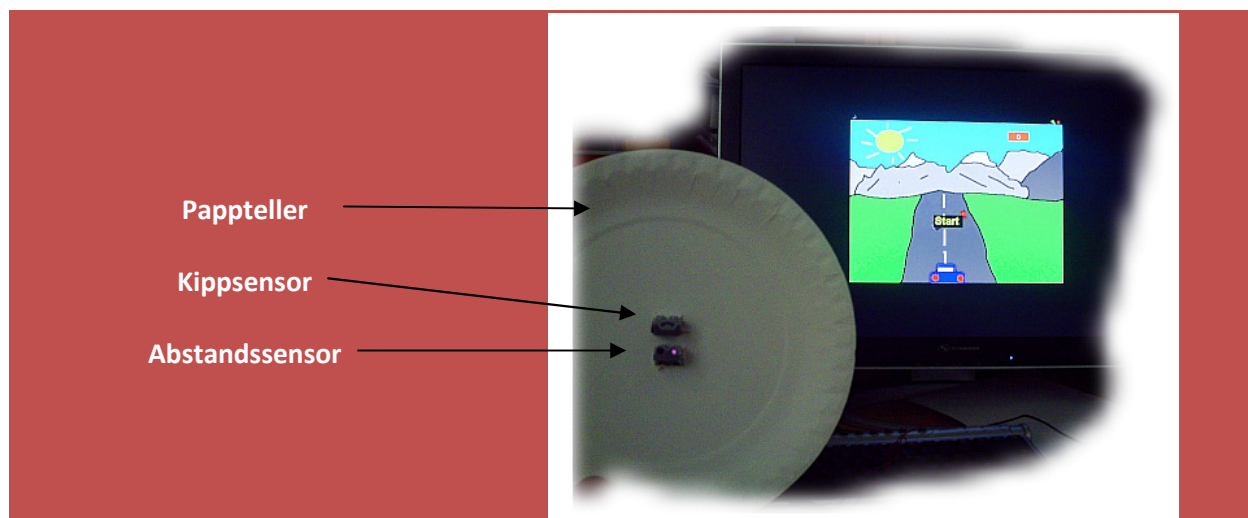
3D Autorennen: Benötigt werden der Tilt-/Kippsensor, der Abstandssensor und ein selbstgebautes Lenkrad.

Folgende Kriterien sollen das Spiel erfüllen:

- Das Auto soll eine Fahrt simulieren
- Der Kippsensor soll den Wagen nach links und rechts steuern
- Der Abstandssensor soll den Wagen nach vorne fahren lassen und zurück
- Es sollen Gegenstände eingesammelt werden, für die es Punkte gibt
- Es gibt einen „Teufel“ – wird dieser berührt ist das Spiel zu Ende
- Die Gegenstände sollen zufällig irgendwann auftauchen
- Fährt das Auto nach vorne, soll es kleiner werden und umgekehrt
- Links und rechts fährt man an Bäumen vorbei
- Bäume, Rasen und Berge dürfen nicht berührt werden ansonsten ist das Spiel zu Ende
- Nach 5 Punkten ändert sich der Hintergrund

Das Lenkrad besteht aus einem einfachen Pappeller. In die Mitte werden beide Sensoren gesteckt. Somit kann nach links und rechts gesteuert werden. Man beginnt mit dem Zeichnen des Hintergrundes. Die Striche auf der Fahrbahn sind ein Objekt mit 2 Kostümen die sich abwechseln, genauso wird das Autofahren animiert. So entsteht der Eindruck eines fahrenden Autos.

„Das Lenkrad“



Beispiel Spiel „Autofahrt“

The screenshot shows the Scratch interface for a game called "autospiel".

- Script Editor (Objekt3):**
 - Script 1:** Triggered by "Wenn ich start empfangen". It sets x to -25, y to -145, shows the object, sets size to 80%, moves forward, and enters a "wiederhole fortlaufend" loop:
 - Change to next costume.
 - If "wird Objekt4 berührt?" or "wird Objekt6 berührt?", say "Spiel vorbei!" for 1 second and send "aus" to all.
 - If "wird Farbe berührt?", send "aus" to all.
 - If "wird Farbe berührt?", send "aus" to all.
 - If "wird Objekt10 berührt?", send "aus" to all.
 - Script 2:** Triggered by "Wenn ich start empfangen". It enters a "wiederhole fortlaufend, falls Wert von Sensor Neigung = 2" loop, changing x by -3.
 - Script 3:** Triggered by "Wenn ich start empfangen". It enters a "wiederhole fortlaufend, falls Wert von Sensor Neigung = 4" loop, changing x by 3.
- Stage:** Shows a car moving on a road. A score of 0 is displayed in the top right. The background features mountains and a sun.
- Object Palette:** Shows various objects including trees, a banana, a car (Objekt3), and a "Start" button.

Das Startskript:

The start script consists of three event-driven blocks:

- Block 1:** "Wenn angeklickt"
 - zeige dich
 - gehe zu x: -22 y: -43
- Block 2:** "Wenn Objekt9 angeklickt"
 - sende start an alle
 - setze zTempo auf 1
- Block 3:** "Wenn ich start empfangen"
 - verstecke dich

Um die Bäume am Straßenrand zu bewegen – sie sollen immer größer werden je näher sie kommen – habe ich folgendes Skript geschrieben:



Für die andere Seite werden die x und y Werte verändert, sowie die Zufallszahlen-Wartezeit. Ähnlich wird bei den Objekten vorgegangen, die auf der Straße einzusammeln sind.



Wenn 5 mehr als 5 Punkte erreicht sind, wechselt der Hintergrund zur Nachtfahrt. Das Spiel kann nun erweitert werden um die Zunahme der Geschwindigkeit oder der Anzahl der Gegenstände. Es können andere Strecken folgen etc. Man kann hieraus ein Projekt für eine ganze Klasse machen. Auch kann differenziert gearbeitet werden. Einige sind sehr gut im entwerfen von Objekten und Bühnenbildern, andere in der Programmierung.

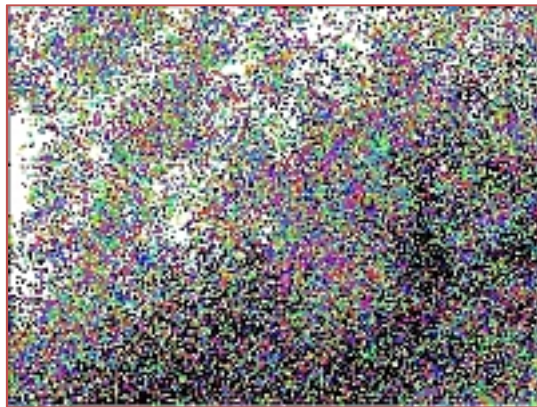
Zufallsgrafiken

Ein interessantes Spiel in Scratch sind Zufallsgrafiken. Hier können interessante Grafiken und Bilder entstehen. Der Phantasie muss man freien Lauf lassen.

Beispielprogramm für ein Objekt:

```
Wenn Taste Leertaste gedrückt
wische Malspuren weg
verstecke dich
setze farbeÄndern auf 3
setze länge auf 2
setze x auf 1
setze y auf 1
setze Stiftdicke auf 1
hebe Stift an
gehe zu x: x y: y
wiederhole fortlaufend
  hinterlasse Abdruck
  ändere Farbe -Effekt um farbeÄndern * x
  gehe länge * Zufallszahl von -10 bis 20 -er Schritt
  zeige Richtung Zufallszahl von 1 bis 300
  drehe länge * drehen Grad
  pralle vom Rand ab
  ändere farbeÄndern um 1
  ändere x um Zufallszahl von -10 bis 10
  ändere y um Zufallszahl von -10 bis 10
  falls wird Farbe berührt?
    setze Stiftfarbe auf
  falls wird Farbe berührt?
    setze Stiftfarbe auf
  falls wird Farbe berührt?
    setze Stiftfarbe auf
  falls wird Objekt4 berührt?
    ändere Farbe -Effekt um 25
    gehe zu x: 133 y: -144
Wenn Taste Pfeil nach unten gedrückt
hebe Stift an
stoppe alles
```

Das Ergebnis mit drei Objekten und veränderten Werten:



Weitaus „hochwertiger“ auch mathematischer ist es Fraktale zu erzeugen.

Fraktale

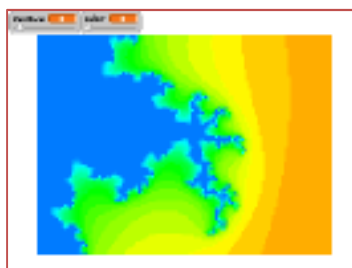
Folgende Begriffe tauchen immer wieder auf: Chaos, Mandelbrot, Julia, Apfelmännchen. Julia war leider keine hübsche Dame, sondern ein französischer Mathematiker (Gaston Julia), der die mathematischen Grundlagen für Fraktale entwickelte, lange bevor es möglich war, diese durch Computer grafisch darzustellen.

Der amerikanische Mathematiker Benoit Mandelbrot griff diese Grundlagen auf, und setzte sie 1980 erstmals grafisch um. Das bekannteste Ergebnis war das sogenannte „Apfelmännchen“, das seinen Namen seiner Ähnlichkeit mit aus Äpfeln gebastelten Männchen verdankt.

Für den Anfang: Auf Basis verschiedener, geeigneter Formeln und Konstanten werden Grafiken berechnet, die zu ihren variierenden Randpunkten (Grenzwerten) immer weiter ins Detail gehen, quasi in einen unendlichen Mikrokosmos ständig neuer Verästelungen und "Elementarteilchen" führen, ähnlich einem Baum, der immer feinere Äste und Verzweigungen aufweist, schließlich Blattstrukturen, die ihrerseits wieder Verästelungen von Adern aufweisen usw. Im Endeffekt entstehen Bilder, die zum Teil an Organisches erinnern, zum Teil aber auch an Galaxien und Spiralnebel.

Die Grundfigur - Je nach verwendeter Formel und Konstanten, entstehen unterschiedliche, meist in irgendeiner Form symmetrische Grundgebilde. Hier kann man nur ausprobieren und experimentieren.

Gute Beispiele findet man hier: <http://scratch.mit.edu/tags/view/fractal>



Wir bauen eine Wii

Was ist eine Wii? Definitionen von **Wii** im Web:

Die Wii ist eine fernsehgebundene (stationäre) Videospiel-Konsole der japanischen Firma Nintendo, die seit Ende 2006 auf dem Markt ist. Das Spielsystem aus Hardware und Software misst dabei die dreidimensionalen Bewegungen des Nutzers. (aus: www.wikipedia.de)

Wii-Fernbedienung

Die Wii-Fernbedienung funktioniert kabellos und reagiert auf Ihre Bewegungen. Dadurch ermöglicht sie eine völlig intuitive, natürliche Art des Spielens. Der ergonomische Controller verwandelt Ihre ganz realen Bewegungen in Bewegungen auf dem Bildschirm. Sie wollen eine Trommel schlagen oder einen Tennisschläger schwingen? Warum sollten Sie dazu einen Knopf drücken? Machen Sie es doch ganz einfach! Dank der Wii-Fernbedienung versetzen Sie ganz normale Bewegungen ins Zentrum Ihres Spiels. (aus: www.amazon.de)

Ganz genau können wir keine Wii bauen, aber dennoch einen Stick, der unsere erstellten Spiele über Funk steuern soll.



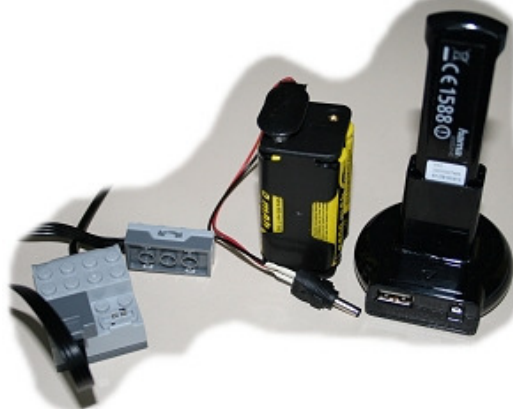
Es handelt sich um ein Set bestehend aus zwei Wireless USB Dongles. Im Prinzip ersetzt das Set ein USB 2.0 Kabel, d.h. es verbindet zwei Geräte mit bis zu 480 MBit/sec. Das ist schneller als die WLAN Übertragung.

Der computerseitige (Host-)Dongle wird einfach an einen freien USB-Port angeschlossen und ist mit 5V Spannung bedient, doch der geräteseitige (Client-)Dongle benötigt eine zusätzliche 5V Stromspeisung via Netzgerät, da das Peripheriegerät an seinem USB-Anschluss entweder gar keinen Strom oder lediglich 500mA zur Verfügung stellt. Für eine stabile Funkverbindung ist

dies wohl zu wenig - deshalb das Netzteil. Ersetzt man jedoch das Netzteil durch ein Batterie-Pack (hier 4x 1,2V AA NiMH **Akkus** = 4,8V), so erhält man eine komplett kabellose Lösung. Um viel Lötarbeiten zu ersparen habe ich einen Batteriehalter mit dem Stecker vom Netzteil versehen.



Um nun Scratch fernzusteuern, werden der USB-Hub, die Sensoren, das Akku-Pack und der Host in einen selbstgebauten Lego-Joystick gepackt.

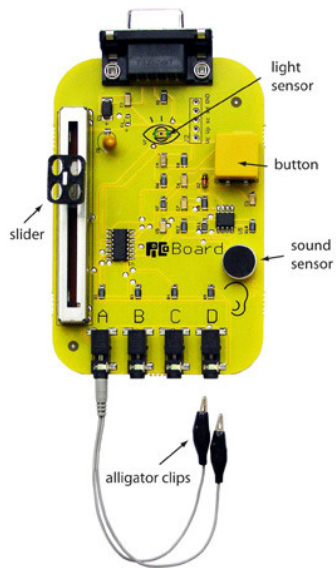




Ein einfacher
Legojoystick.
Ganz links sieht man
die Anschlüsse. Um
nicht immer mit dem
Akkupack verbunden
zu sein habe ich die
Stellen offen
gelassen. Ganz oben
kommen die
Sensoren. Das
erleichtert das
Auswechseln von
Sensoren.

Rechnet man alle Kosten zusammen kommt auf ca. 90,00 € mit 4 Akkus.

Man kann alle möglichen Verpackungen nehmen, um eine Fernbedienung zu bauen.



Picoboard

Mit dem Picoboard kann das Scratch-Programm fühlen und reagieren. Das Programm verlässt den Rechner und wird interaktiv. Das Picoboard hat verschiedene Sensoren: Sound-, Tast-, Regel- und Lichtsensor. Ebenso stehen 4 Ausgänge (A, B, C, D) zur Verfügung mit Kabelanschlüssen.

Mit dem Picoboard werden also verschiedene Sensoren angesprochen. Mit dem USB-Hub können Lego-Motoren

angesteuert werden. Man kann also einen Motor mit einem Soundsensor steuern oder mit dem Taster den Motor einschalten. Für interaktive Spiele oder Anwendungen eine sinnvolle Bereicherung. Es gibt dadurch eine Vielzahl an Möglichkeiten für den Einsatz im Physik- und Informatikunterricht.

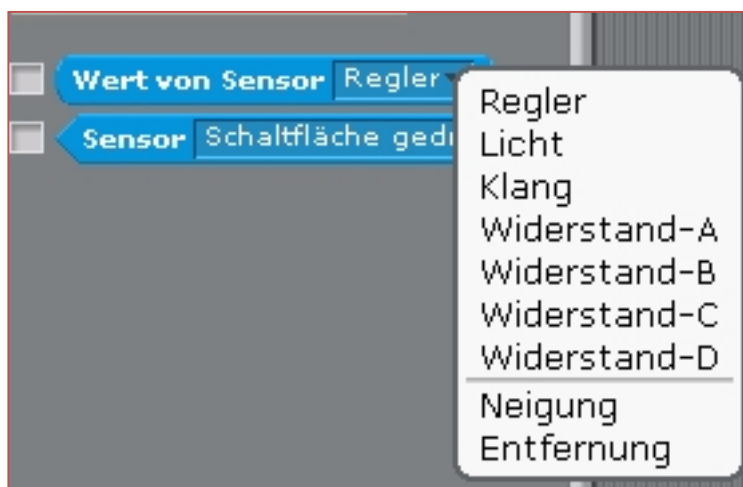
Das Picoboard wird mit einem USB Kabel und 4 Y-Kabeln mit Krokodilklemmen geliefert. Die Kabel können für „eigene“ Sensoren verwendet werden, wie z.B. einen Hautwiderstand. Das Picoboard mit dem Mikrokontroller und Kabeln kostet ca. 50 Dollar (ca. 35 €). Es müssen zuerst die Treiber von der Homepage heruntergeladen werden, damit Scratch das Board erkennt.

Bestellung & download der Treiber unter: <http://www.picocricket.com/picoboard.html>

WeDo und das Picoboard können via USB parallel betrieben und genutzt werden.

Arbeiten mit dem Picoboard

Wenn die Treiber installiert sind und das Board per USB angeschlossen ist, kann losgelegt werden. Das Schiebepotentiometer gibt einen Wert zwischen 0 und 100 aus. Ein Klick vor den Block „Regler“ und man kann auf der Bühne den Wert ablesen.



Um ein Objekt zu bewegen ändert man in der Bewegung die Koordinate durch ablesen des Sensorwertes „Regler“.



Genauso kann der Wert nun durch andere Sensorwerte ersetzt werden.

Der Lichtsensor

Um diesen Sensor anzuwählen muss man in der Kategorie „Fühlen“ den Lichtsensor anwählen. Wenn die Checkbox angeklickt wird, kann der Wert für diesen Sensor abgelesen werden. Legt man nun langsam die Hand über das Picoboard – der Sensor ist an dem aufgedruckten Auge zu erkennen -, so ändert sich der Wert. Mit dem Lichtsensor kann nun ein Objekt auf der Bühne oder ein Motor angesteuert werden.



Um mehr Empfindlichkeit zu bekommen benutzt man Operatoren. Hier wird der Sensorwert um den Faktor 5 multipliziert um eindeutige Bewegungen zu bekommen.



Der Soundsensor

Der Soundsensor misst die Lautstärke eines Klanges. Die Software gibt einen Wert von 0-100 zurück. Je lauter der Klang umso höher der Wert. Als Beispiel wird bei jedem Klatschen in die Hände das Kostüm gewechselt. Die Sensorwerte können dazu genutzt werden um z.B. grafische Effekte zu erzwingen.

Ein Objekt (Objekt) wird mit dem Wert des Soundsensors bewegt.



Ein Objekt (Objekt) ändert das Aussehen, vorausgesetzt es gibt ein weiteres Kostüm für dieses Objekt.

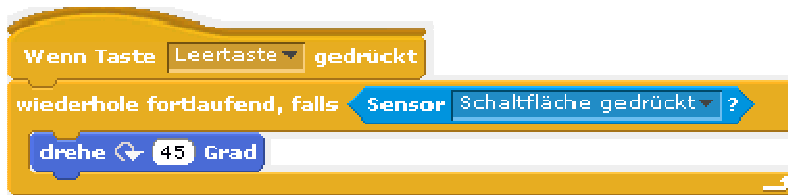


Mit dem Soundsensor kann man grafische Effekte hervorrufen.



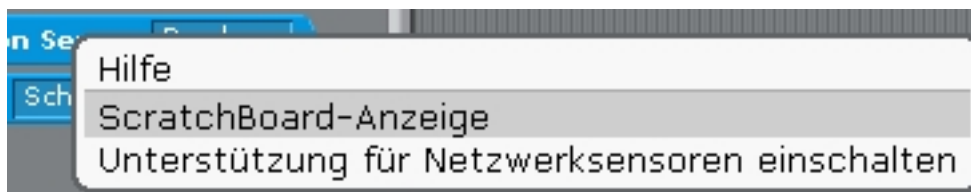
Der Taster

Der Tastsensor gibt den Wert „wahr“ oder „falsch“ wieder. Ist der Taster gedrückt (wahr) oder nicht (falsch).



Wenn der Taster „wahr“ liefert, wird das Objekt (Objekt) um 45 Grad im Uhrzeigersinn gedreht.

Alle Sensoren anzeigen lassen



Um sich alle Werte des Boards anzeigen zu lassen, klickt man in der Kategorie „Fühlen“ mit der rechten Maustaste auf einen Sensorwert.

Nach der Bestätigung erscheint das linke Fenster auf der Bühne.

Fehlermeldungen

Wenn das Picoboard nicht auf Anhieb funktioniert kann es an den fehlenden Treibern liegen. Diese gibt es auf der Internetseite von dem Picoboard (Vorwort).

Wenn dennoch keine Werte angezeigt werden:

- USB Kabel überprüfen
- Das Programm starten, das USB Kabel entfernen und wieder einstecken
- Das USB Kabel entfernen, einstecken und in der Kategorie „Fühlen“ das Sensorwert Regler anklicken
- Das Programm oder Scratch erneut starten

Eigene Sensoren



Um die Ausgänge des Picoboard (Widerstände)A-D zu nutzen ein kleines Beispiel.

Pico-Schlagzeug

Eine interessante Variante des Schlagzeug spielens. Um ein akustisches in ein elektronisches Schlagzeug umzuwandeln benötigt man hier nicht viel. Die Felle dämmen und auf den Dämmstoff leitendes Material befestigen. Schon kann man über Kopfhörer in der Wohnung üben. Eine Schlagzeug Idee kann auch so aussehen:

Man nehme 4 Pappteller, klebe hinten Aluminiumfolie und einen Draht auf die Rückseite der Teller, nehme einen Schaschlik Spieß oder Nähnadel aus Metall, befestige die anderen Leitungen daran und lege los. Für die Basedrum haben wir etwas Aluminiumfolie um den Fuß gewickelt.

Im Bild sieht es dann so aus:



„Basedrum“



Das Programm kann dann so aussehen (für eine Trommel):



Ein Problem: Wir haben nur 4 Ausgänge und bei einem Schlagzeug gibt es mehrere Trommeln. Ebenso braucht jeder Ausgang einen Anschluss für den „Drumstick“. Man spielt aber nur mit 2 Schlagzeugstöckern. Eine Abhilfe können hier selbst gebaute Kabel sein.

Ich habe z.B. einen Mikrolinkenstecker genommen, 3 Krokodilklemmen und 2 Widerstände. Wenn man daraus 4 Kabel baut für die 4 Ausgänge, dann hat man statt der 4 Ausgänge jetzt 12 Ausgänge. Man kann also die Anzahl der Ausgänge am Picoboard durch Widerstände erhöhen. Für ein

Schlagzeug einfach genial. In dem nächsten Beispiel wurden ebenso selbstgebaute Kabel verwendet. Die Kosten sind sehr gering.



Scratch, WeDo und Picoboard

Mit den Sensoren kann z.B. die Motorleistung kontrolliert werden. Ein Beispiel um die Motorleistung mit dem Regler zu kontrollieren:

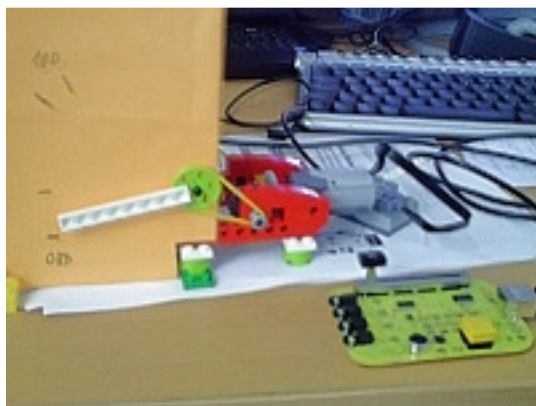


Klick man mit der rechten Maustaste auf einen Block „Wert von Sensor ...“, kann man sich auf der Bühne alle Werte des Picoboard anzeigen lassen.

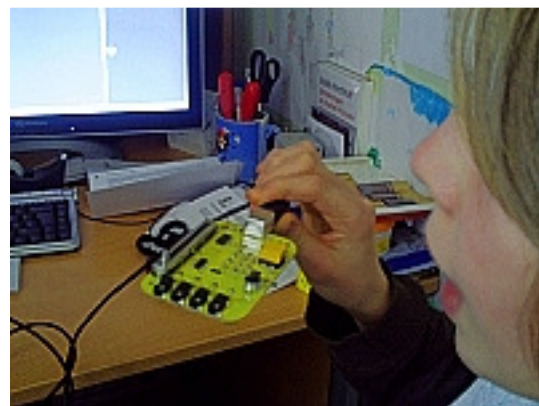
Umweltmessgeräte entwickeln (Skala per Motor)

Beispiel: ein Lautstärkemessgerät mit Scratch, Wedo und dem Picoboard.

Als Zeigergerät für die Lautstärkenwerte habe ich das Krokodil (Bauanleitung im Internet - siehe Seite 13) etwas umgebaut. Vielmehr das Maul entfernt und ein Legobauteil als Zeiger an das Rad geheftet. Ein einfacher Pappdeckel fungiert als Skala. Das Messgerät ist in diesem Beispiel nicht geeicht und die Werte sind erst einmal frei erfunden.



1.



2.



3.

1. Ohne Ton ist der Zeiger auf 0
2. Ein Ton wird vom Picoboard gemessen
3. Der Zeiger zeigt die Lautstärke.

Gleichzeitig wird in dem Programm der Pegel angezeigt (zum Vergleich).

Das Programm (nächste Seite):

```

Wenn  angeklickt
  sende start an alle

Wenn ich start empfangen
  warte 1 Sek.
  gehe zu x: -66 y: -150
  wiederhole fortlaufend
    falls Wert von Sensor Klang = 0
      gleite 1 Sek. zu x: -65 y: -150
    falls Wert von Sensor Klang > 1 und Wert von Sensor Klang < 30
      gleite 1 Sek. zu x: -65 y: -50
      setze Motornrichtung auf in diese Richtung
      setze Motorkraft auf 20
      schalte Motor für 0,2 Sekunden an
      schalte Motor aus
      warte 1 Sek.
      setze Motornrichtung auf umkehren
      setze Motorkraft auf 20
      schalte Motor für 0,2 Sekunden an
      schalte Motor aus
      warte 1 Sek.
    falls Wert von Sensor Klang > 31 und Wert von Sensor Klang < 60
      gleite 1 Sek. zu x: -65 y: 30
      setze Motornrichtung auf in diese Richtung
      setze Motorkraft auf 20
      schalte Motor für 0,4 Sekunden an
      schalte Motor aus
      warte 1 Sek.
      setze Motornrichtung auf umkehren
      setze Motorkraft auf 20
      schalte Motor für 0,4 Sekunden an
      schalte Motor aus
      warte 1 Sek.
    falls Wert von Sensor Klang > 61 und Wert von Sensor Klang < 80
      gleite 1 Sek. zu x: -65 y: 60
      setze Motornrichtung auf in diese Richtung
      setze Motorkraft auf 20
      schalte Motor für 0,6 Sekunden an
      schalte Motor aus
      warte 1 Sek.
      setze Motornrichtung auf umkehren
      setze Motorkraft auf 20
      schalte Motor für 0,6 Sekunden an
      schalte Motor aus
      warte 1 Sek.
    falls Wert von Sensor Klang > 81 und Wert von Sensor Klang < 100
      gleite 1 Sek. zu x: -65 y: 150
      setze Motornrichtung auf in diese Richtung
      setze Motorkraft auf 20
      schalte Motor für 0,8 Sekunden an
      schalte Motor aus
      warte 1 Sek.
      setze Motornrichtung auf umkehren
      setze Motorkraft auf 20
      schalte Motor für 0,8 Sekunden an
      schalte Motor aus
      warte 1 Sek.
    falls Sensor Echaltfläche gedrückt
      stoppe alles
  
```


Kühlschrankalarm und Alarmanlage

(wenn Licht angeht, dann Sound) oder eine LegoFigur wackelt. Hier eignet sich der Trommelnde Affe aus Wedo recht gut (Bauanleitung im Internet unter www.nxt-roboter.de/wedo/).

Inspiriert von www.amazon.de: *Kühlschrankschwein oder Diätschwein*

Dieses niedliche Kühlschrankschwein kann Ihnen helfen die Kühlschranktür, doch besser wieder zu schließen und nicht mehr zu naschen! Das Diät-Schweinchen reagiert auf den Lichtsensor im Kühlschrank, wenn die Tür sich öffnet und das Licht angeht, dann grunzt es ganz niedlich.....



Mit dem gleichen Prinzip kann man eine Alarmanlage mit dem WeDo Annäherungssensor oder dem Tiltensor bauen.

Ein Feuermelder bauen

Als **Brandmelder** werden technische Geräte oder Anlagen zum Auslösen eines Alarms im Falle eines Brandes in Wohnungen, öffentlichen Einrichtungen, Verkehrsmitteln oder Industrieanlagen bezeichnet. Dabei wird unterschieden zwischen *automatischen* Brandmeldern, die den Brand anhand seiner physikalischen Eigenschaften erkennen, und *nicht-automatischen* Brandmeldern, die von Hand betätigt werden müssen. Sinn des Brandalarms ist das Initiieren von Maßnahmen zur Brandbekämpfung, der Warnung von Personen innerhalb eines Gebäudes und, zum Sach- und Personenschutz, meist die Alarmierung von zuständigem Sicherheitspersonal oder auch der Feuerwehr.

Im 19. Jahrhundert wurde der Begriff **Feuertelegraph** für verschiedene elektrische, mechanische und akustische Brandmeldeanlagen verwendet. (www.wikipedia.de)

Unser Feuermelder kann mit Scratch, Wedo und dem Picoboard gebaut werden. Ich habe einen Temperaursensor an das Picoboard angeschlossen. Bei Erreichen eines bestimmten Wertes soll Alarm ausgelöst werden und z.B. der WeDo Affe trommelt auf Blech.

Folgende Werte haben sich ergeben:

Die Skala der Werte reicht von 0-100. Das Board zählt von 100= kalt bis 0= heiß.

Beispiel: Die Außentemperatur von 0 Grad Celsius hat den Wert 80. Die Zimmertemperatur den Wert 54 und wenn ich kurzzeitig ein Feuerzeug unter den Sensor halte einen Wert von 25.

Das gesteuerte Energiehaus – Ein Gruppenprojekt

Um mit Scratch, dem Picoboard und Lego WeDo ein ganzes Haus zu regeln, benötigt man ein schönes Spielehaus. Das Playmobil-Haus eignet sich darum, da es sehr groß und von einer Seite offen ist. Dieses Haus kann nun mit verschiedenen Sensoren und Motoren „verbessert“ werden.

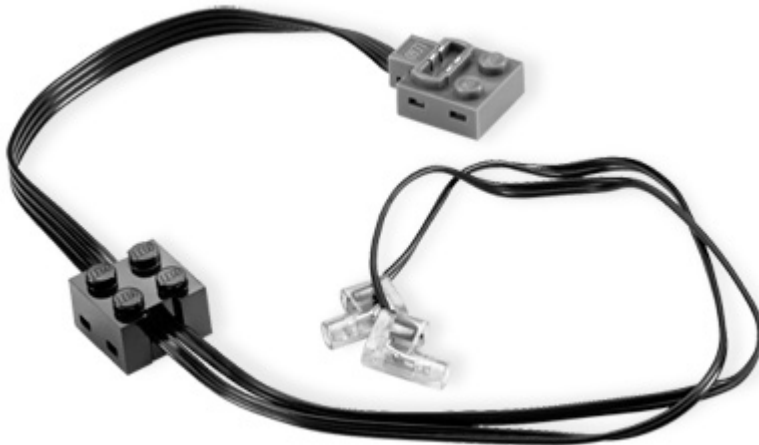
- die Temperatursteuerung soll im Gebäude optimiert wird. Beispielsweise werden Wohnräume hauptsächlich zwischen 7 und 10, bzw. 17 und 23 Uhr geheizt
- Die Lichtsensorgesteuerten Rollläden bewirken eine optimale Sonneneinstrahlung
- Integriert in diese intelligente Steuerung sind selbstverständlich Wassersensoren, Regenmelder, Windmesser, Bewegungsmelder sowie andere Sensoren für eine Alarmanlage
- Die Beleuchtung kann an – oder abgeschaltet werden
- Türklingel
- Das Sonnensegel soll ausfahren bei Sonne (WeDo)
- Die Fenster schließen bei Regen
- Automatischer Türöffner
- Zeitlichtsteuerung
- Erweiterungen selbst bauen

Und vieles mehr.



Foto: [PLAYMOBIL® Neues Wohnhaus Traumhaus Haus 4279](#)

Statt Lego Motoren: Lampen



LEGO® Power Functions LED

Durch Anbringen der hellen Leuchtdioden erhalten Modelle leuchtende Augen, strahlende Scheinwerfer und vieles mehr. Der Fantasie sind keine Grenzen gesetzt. (Beim Test funktionierten diese nur mit dem Picoboard an den Widerstandsausgängen)

Bestellen bei www.Lego.de Artikel-Nr.: 8870 Preis: 7,50 €

Adapterkabel bauen

- Power Functions Legokabel auf Mikroklinke mono. Damit kann man dann Power Functions Teile an das Picoboard anschließen.

2,5 mm (selten): für besonders kleine Geräte verwendet, wie Headsets für Mobiltelefone. Findet auch Verwendung zur Datenübertragung bei manchen Taschenrechnern oder für Kabelauslöser bei Fotoapparaten. Diese Steckergröße ist bekannt als „Mikroklinke“.



Klinkenstecker 2,5mm Mono mit Knickschutz – bei www.reichelt.de nur 0,08 €

- Legokabel Verlängerungskabel in der Mitte durchtrennen und an dem anderen Ende Krokodilklemmen anlöten. So kann man mit dem Wedo USB-Hub experimentieren und auch andere z.B. LED etc. anschließen.

Krokodilklemmen werden zur schnellen Herstellung von elektrischen Verbindungen beispielsweise in Elektronikwerkstätten oder -Labors benutzt. Sie können je nach Federkraft und Material Ströme bis etwa 15 Ampere übertragen. 10-teiliger Verbindungsleitungs-Satz

Kostet z.B. bei www.reichelt.de 1,15 €



- Power Function Verlängerungskabel auf RCX Kabel. Damit lassen sich Bauteile und Sensoren des RCX an WeDo anschließen.
- Adapterkabel für das Picoboard. Um verschiedenen Werte zu erhalten werden in die Kabel Widerstände eingebaut. So kann man pro Ausgang mehrere Kabel anschließen und das Board hat mehrere Anschlussmöglichkeiten statt nur 4.

Lichteffekte oder Motorenleistung mit einem Potentiometer steuern



Drehpotentiometer, mono, linear, 100 Ohm

Werte von 0.4 bis 34.5 in der Anzeige. Je nach Potentiometer können die Werte natürlich abweichen. Durch Operatoren lassen sich die Werte beeinflussen.



LEGO 9686 Naturwissenschaft und Technik (inkl. Motor)

Eine optimale Ergänzung oder auch für den Anfang ist dieses Set.

Dieser Universalbaukasten für den naturwissenschaftlichen Unterricht lässt Schüler in Kleingruppen Naturgesetze experimentell erforschen. Das Set für 2-3 Schüler enthält 396 Elemente, darunter Zahnräder, Räder, Achsen, Nocken, Riemen, Windsegel und Motor mit Batteriebox. Zudem farbige DIN A5 Bauanleitungshefte zu 10 Modellen mit technischen Grundprinzipien sowie 18 Modellprojekten. Zur leichteren Orientierung ist der Kasten mit Sortiermagazin ausgerüstet und enthält farbige Elementeübersichten. Inklusive stabile blaue Aufbewahrungsbox. Erweiterungsset LEGO® 9641 optional erhältlich. Mit fünf zusätzlichen Bauanleitungen für pneumatische Modelle. Unter anderem kann, zusammen mit dem Baukasten 9686, ein Kompressor gebaut werden. Nachfolger des erfolgreichen 9632.

Konstruieren & Erforschen realer Maschinen und Mechanik Erforschen von Kräfteverhältnisse: z.B. Erfassung und Kalibrierung von Wind Dieser Universalbaukasten für den naturwissenschaftlichen Unterricht lässt Schüler in Kleingruppen Naturgesetze experimentell erforschen. Das Set für 2-3 Schüler enthält 396 Elemente, darunter Zahnräder, Räder, Achsen, Nocken, Riemen, Windsegel und Motor mit Batteriebox. Zudem farbige DIN A5 Bauanleitungshefte zu 10 Modellen mit technischen Grundprinzipien sowie 18 Modellprojekten. Zur leichteren Orientierung ist der Kasten mit Sortiermagazin ausgerüstet und enthält farbige Elementeübersichten. Inklusive stabile blaue Aufbewahrungsbox. Erweiterungsset LEGO® 9641 optional erhältlich. Mit fünf zusätzlichen Bauanleitungen für pneumatische Modelle. Unter anderem kann, zusammen mit dem Baukasten 9686, ein Kompressor gebaut werden.

Bei www.nxt-roboter.de



Mit Scratch ein Fahrzeug steuern

1 Motor fährt langsam vorwärts, Lenkung durch Scratch

Als Beispiel nehme ich den neuen Lego ©Technic Buggy Nr. 8084. Für den Antrieb baut man einen Power Functions Motor ein, der über die Batteriebox betrieben wird. Dieses ist der Antrieb für das einfache Fahren. *(Bild: Power Functions Batteriebox)*



An der Lenkung baut man nun einen zweiten Power Function Motor. Dieser wird mit dem USB Hub und dem Funk USB in die Fahrerkabine eingebaut. Nun kann das Fahrzeug von Scratch aus gelenkt werden. Das Stoppen geht natürlich nicht. Dazu benötigt man 2 USB Funk und 2 PC. Zumindest kann ein Abstandssensor angebracht werden um Hindernissen auszuweichen. Ein toller Experimentierspaß. Je größer das Fahrzeug, umso mehr Möglichkeiten hat man. Bei dem Baukran kann man mit vielen Motoren agieren. Es muss auch nicht unbedingt Funk sein.

Geeignete Fahrzeuge sind z.B.:

Rechts: großer Geländewagen



Raupenbagger



Hier kann der Schüler seiner Phantasie freien Lauf lassen.

Ideen für den Einsatz in Nicht-Informatik-Fächern

Zum Teil aus: <http://wiki.zum.de/Scratch>

Mathematik

- Kreisbewegungen können mittels **Sinus und Kosinus** dargestellt werden.
- **Gleichungssysteme** zu Bewegungsaufgaben können untersucht werden.
- Spirograph nachprogrammieren.
- Darstellung von räumlichen Figuren und deren Bewegung.
- dreidimensionale Darstellung von vierdimensionalen Körpern.
- Zeichnen von Fraktalen.
- Anleitung zum Lösen von Gleichungen mit Bruchtermen
- Eigenschaften von Vierecken

Physik

- Fallkurven
- Beschleunigte Bewegungen (mit Power Function Motoren / WeDo umsetzbar)
- Reflektion von Lichtstrahlen
- Kanone abschießen, Basketball werfen
- Springende Bälle
- Lissajous Figuren

Biologie

- Kreislauf des Wassers (ein Frosch möchte wissen, wo das Wasser herkommt und macht sich auf die Reise) (mit WeDo umsetzbar)
- Bewegungen von Tieren
- „Verhaltensforschung“ (WeDo Annäherungssensor)

Deutsch

- Tipps zur "Erörterung" (ein Fisch schwimmt von Pflanze zu Pflanze und denkt sich: zuerst Argumente sammeln, einteilen - pro und contra, dann Einleitung, ...) (mit WeDo umsetzbar)

Erdkunde

- Kreislauf der Sonne
- Planetenbahnen
- Wetter

Musik

- Instrumente zeichnen, Klänge zuordnen
- Elektronisches Schlagzeug
- Gehörbildungsspiele

Links

<http://www.funlearning.de/>

http://de.wikipedia.org/wiki/Scratch_%28Programmiersprache%29

<http://scratch.mit.edu/> Originalseite Scratch Software

<http://www.olinger.net/iWeb/berufliches/Scratch.html>

<http://www.et.hs-wismar.de/kinderuni/de/index.html>

<http://www.brandhofer.cc/?cat=7>

www.nxt-roboter.de Online Shop Lego Mindstorms Education & WeDo

www.mindstormsforum.de Internetforum mit Unterforum für WeDo

Die Kinder Uni Wismar mit einer tollen Bauanleitung für ein Klavier:

http://www.et.hs-wismar.de/kinderuni/files/Das_Klavier.pdf

FAQ

Wieso werden auf der Homepage keine Projekte angezeigt?

Um Scratch-Projekte auf der Webseite ansehen zu können, benötigt man das „Java Runtime Environment“. Auf der Java-Seite <http://www.java.com/de/download/> kann man seine Java-Installation überprüfen oder Java herunterladen.

Wie kann ich Projekte von der Scratch-Seite herunterladen?

Um Projekte herunterladen zu können, musst man auf der Scratch-Seite ein kostenloses Benutzerkonto eingerichtet haben. Man sucht sich ein Programm aus und klickt auf den Downloadlink. Der Downloadlink befindet sich rechts in der orangen Box.

Welches sind die Systemvoraussetzungen für Scratch

- Anzeige: 1024 x 768 Pixel ab 16Bit Farbtiefe
- Betriebssystem: PC: ab Windows 98, Mac: ab Mac OS X 10.3
- Festplatte: mindestens 120 MB freier Speicher
- Die Prozessorgeschwindigkeit der meisten Prozessoren reicht für Scratch locker aus. Auf älteren Computern wird Scratch vielleicht ein wenig langsamer laufen, aber es läuft.

Wie kann ich meine Scratch-Projekte auf meiner Webseite einfügen oder sie auf CD brennen?

Wie bringe ich ein Scratch-Projekt auf eine CD, einen USB-Stick oder ins Internet?

Es gibt da mehrere Möglichkeiten:

1. In Scratch hast man die Möglichkeit Projekte zu veröffentlichen. Dieses geschieht durch den Menüpunkt „Veröffentlichen!“. Dafür ist eine einmalige Registrierung auf der Scratch-Homepage notwendig. Das Projekt ist danach öffentlich zugänglich, kann zu Galerien hinzugefügt werden und kann auch in andere Internetseiten eingebunden werden.
2. Am einfachsten ist es, die Projektdatei (mit der Endung *.sb) mitsamt folgender Dateien auf eine CD zu brennen / auf einen USB-Stick zu kopieren:
 - Scratch.exe
 - Scratch.image
 - Scratch.ini
 - ScratchPlugin.dll
 - Mpeg3Plugin.dll
 - license.txt (diese Dateien befinden sich im Scratch-Programmordner)

Um ein Projekt später von CD oder USB-Stick zu starten, klickt man doppelt auf Scratch.exe, danach auf „open“ und wähle das Projekt aus.

3. Für Fortgeschrittene: Es gibt noch einen anderen Weg, wie man Projekte auf einer CD oder im Internet veröffentlicht. Dafür benötigt man die folgenden Dateien:
<http://scratch.mit.edu/static/misc/ScratchApplet.jar>
<http://scratch.mit.edu/static/misc/soundbank.gm>

Das Scratch-Projekt (mit der Endung *.sb) kopiert man in dasselbe Verzeichnis wie die Datei ScratchApplet.jar und die HTML-Seite (mit der Endung *.html) mit folgendem Inhalt:

Code:

```
<html>
<body>
<applet id="ProjectApplet" style="display:block" code="ScratchApplet" codebase="./"
archive="ScratchApplet.jar" height="387" width="482"> //kein Zeilenumbruch!
<param name="project" value="meinProjekt.sb">
</applet>
</body>
</html>
```

Weitere Hilfe unter : <http://scratch.mit.edu/forums/viewtopic.php?pid=176504>

Referenz



Das Programm wird gestartet, sobald die grüne Flagge angeklickt wird.

Das Programm wird gestartet, sobald die Leertaste gedrückt wird.

Das Programm wird gestartet, sobald das Objekt1 angeklickt wird.

Warte die angegebene Zeit in Sekunden, dann wird der nächste Block ausgeführt.

Führt die eingefügten Blöcke als Endlosschleife aus.

Führt die eingefügten Blöcke x-mal aus.

Sendet eine Nachricht an alle Objekte. Diese führen daraufhin eine Aktion aus ohne auf die Fertigstellung der Aktion zu warten.

Sendet eine Nachricht an alle Objekte und wartet bis die Aktionen ausgeführt sind.

Wenn eine Nachricht empfangen wird, werden die angedockten Blöcke ausgeführt.

Die Endlosschleife mit einer Bedingung verknüpft. Wenn die Bedingung erfüllt ist werden die eingefügten Blöcke ausgeführt.

Wenn-Bedingung. Wenn die angegebene Bedingung erfüllt ist (wahr), dann werden die eingefügten Blöcke ausgeführt.

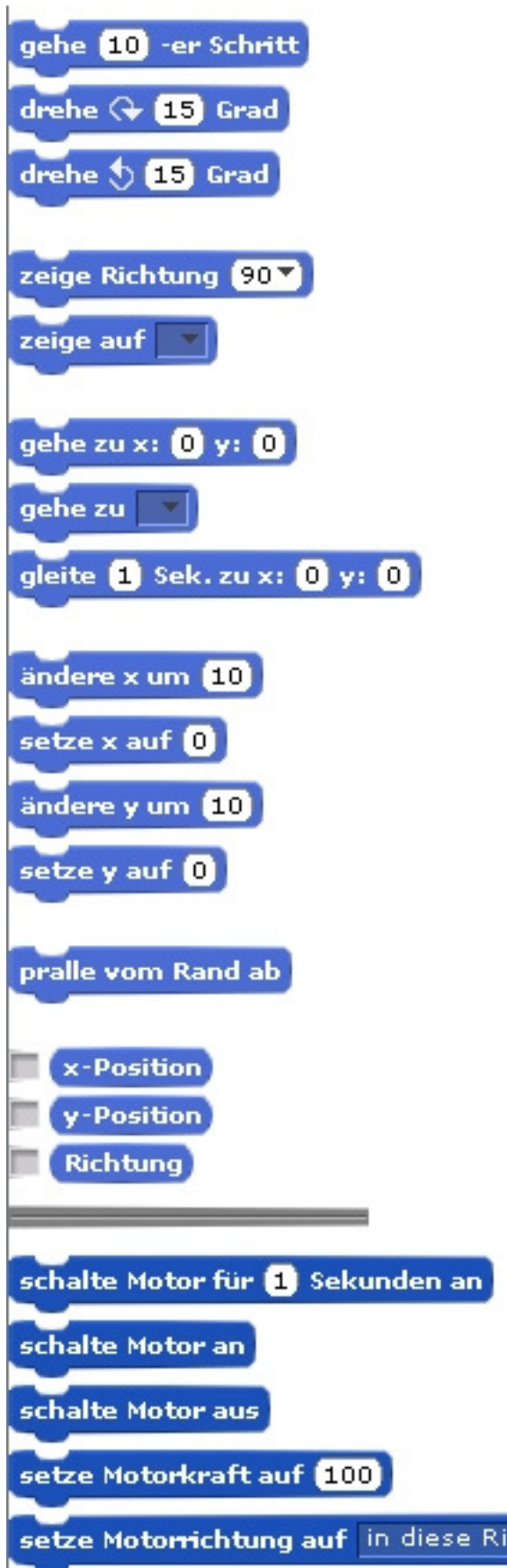
Wie die vorherige Bedingung mit dem Unterschied, dass eine nicht erfüllte Bedingung (falsch) durch die unten eingefügten Blöcke ausgeführt wird.

Der Programmfluss wird unterbrochen bis eine bestimmte Bedingung erfüllt ist.

Führt die eingefügten Blöcke als Endlosschleife aus, wenn eine Bedingung erfüllt ist (wahr).

Stoppt das Programm.

Stoppt alle Objektprogramme.



Bewegt ein Objekt vorwärts oder rückwärts (+ oder -).

Dreht ein Objekt im Uhrzeigersinn.

Dreht ein Objekt gegen den Uhrzeigersinn.

Ein Objekt soll in eine bestimmte Richtung schauen. 0= nach oben; 90=nach rechts; 180=nach unten und -90=nach links.

Ein Objekt soll sich in Richtung des Mauszeigers oder eines anderen Objektes drehen.

Bewegt das Objekt zu der angegebenen X- und Y-Koordinate auf der Bühne.

Ein Objekt soll in die Richtung des Mauszeigers oder eines anderen Objektes gehen.

Das Objekt soll zu der angegebenen X- und Y-Koordinate auf der Bühne in der angegebenen Zeit gleiten.

Ändert die X-Position des Objektes um den eingegebenen Wert.

Setzt die X-Position auf den eingegebenen Wert.

Ändert die X-Position des Objektes um den eingegebenen Wert.

Setzt die X-Position auf den eingegebenen Wert.

Dreht das Objekt um 180 Grad, wenn der Rand berührt wird.

Zeigt die X-Position des Objekts an.

Zeigt die Y-Position des Objekts an.

Zeigt die Richtung des Objekts an.

Wenn der USB-Hub von WeDo angeschlossen ist werden diese Blöcke sichtbar.

Die Motorleistung von 0-100.

Die Drehrichtung des Motors.

ziehe Kostüm **Kostüm2** an

nächstes Kostüm

 Kostüm Nr.

sage **Hallo!** für **2** Sek.

sage **Hallo!**

denke **Hmm...** für **2** Sek.

denke **Hmm...**

ändere **Farbe** -Effekt um **25**

setze **Farbe** -Effekt auf **0**

schalte Grafikeffekte aus

ändere Größe um **10**

setze Größe auf **100** %

 Größe

zeige dich

verstecke dich

komme nach vorn

gehe **1** Ebenen nach hinten

Ändert das Aussehen des Objektes durch die Wahl eines „Kostüms“.

Das Objekt wechselt zum nächsten Kostüm (wenn vorhanden).

Gibt auf dem Bildschirm die Kostümnummer aus.

Eine Sprechblase mit dem Text erscheint für die angegebene Zeit.

Eine Sprechblase mit dem Text wird angezeigt. Ohne Text verschwindet die Sprechblase wieder.

Eine Denkblase mit dem Text erscheint für die angegebene Zeit.

Eine Denkblase mit dem Text wird angezeigt. Ohne Text verschwindet die Denkblase wieder.

Verschiedene Grafikeffekte werden durch den Wert am Objekt verändert. (Farbe, Fischaug, Wirbel, Pixel, Mosaik, Helligkeit, Durchsichtigkeit)

Einen Grafikeffekt für das Objekt mit einem Wert festlegen.

Alle Grafikeffekte eines Objekts ausschalten.

Ändert die Größe des Objekts durch den angegebenen Wert.

Setzt die Größe des Objektes auf einen Prozentwert.

Zeigt die Objektgröße in Prozent zur Originalgröße an.

Objekt auf der Bühne zeigen.

Objekt verstecken (ausblenden).

Legt das Objekt in den Vordergrund (über andere Objekte).

Ein Objekt in der angegebenen Zahl nach „hinten“ gehen lassen. Ein Objekt kann dadurch hinter einem anderen Objekt versteckt werden.



Der Wert „wahr“ wird zurückgegeben, wenn der Rand oder der Mauszeiger berührt wird.

Der Wert „wahr“ wird zurückgegeben, wenn die ausgewählte Farbe berührt wird.

Der Wert „wahr“ wird zurückgegeben, wenn ein Pixel des Objekts die andere Farbe berührt.

Eine Sprechblase die ein Antwortfeld bereitstellt. Wird die erwartete Antwort eingegeben, wird der Wert „wahr“ ausgegeben. Die Antwort kann man sich auf der Bühne anzeigen lassen.

Die X- und Y- Position des Mauszeigers.

Der Wert „wahr“ wird zurückgegeben, wenn die Maustaste gedrückt wird.

Der Wert „wahr“ wird zurückgegeben, wenn die Leertaste (Auswahlfeld beachten) gedrückt wird.

Gibt den Abstand zu dem angegebenen Objekt oder Mauszeiger zurück.

Setzt die Stoppuhr auf 0.

Gibt den Wert der Stoppuhr zurück. Dieser kann angezeigt werden.

Auswahl von Objekt- oder Bühneneigenschaften.

Gibt die Lautstärke des Mikrofontons zurück (1-100).

Der Wert „wahr“ wird zurückgegeben, wenn der gemessene Ton lauter als 30 von 100 ist.

Auswahl der verschiedenen Sensoren.

Der Wert „wahr“ wird zurückgegeben, wenn die Schaltfläche gedrückt oder einer der Widerstände des Picoboard verbunden ist.

spiele Klang

Spielt den angegebenen Ton/Sound ab und springt zum nächsten Block weiter. Der Ton wird dennoch ganz ausgespielt.

spiele Klang ganz

Spielt erst den ganzen Ton/Sound ab und geht dann weiter zum nächsten Block.

stoppe alle Klänge

Stoppt alle Klänge.

spiele Schlagzeug für S

Spielt den ausgewählten Schlagzeug Sound oder Percussion Instrument für die angegebene Zeit ab. (Von 35 – 81)

spiele Pause für Schläge

Länge einer Klangpause angeben.

spiele Note für Schläge

Gleiche wie oben, nur öffnet sich bei Mausklick eine Klaviatur und man kann einen Ton auswählen für eine bestimmte Dauer.

setze Instrument auf

Ein Instrument oder Effekt aussuchen. (Von 1-128)

ändere Lautstärke um

Die Lautstärke ändern.

setze Lautstärke auf %

Die Lautstärke festsetzen in Prozent.

Lautstärke

Zeigt die Lautstärke in Prozent an.

ändere Tempo um

Die Geschwindigkeit setzen.

setze Tempo auf Schläge/Min.

Das Metronom in Scratch.

Tempo

Metrum anzeigen lassen.

wische Malspuren weg

senke Stift ab

hebe Stift an

setze Stiftfarbe auf 

ändere Stiftfarbe um 10

setze Stiftfarbe auf 0

ändere Farbstärke um 10

setze Farbstärke auf 50

ändere Stiftdicke um 1

setze Stiftdicke auf 1

hinterlasse Abdruck

Entfernt alles Gezeichnete.

Setzt den Malstift des Objektes auf die Bühne.

Hebt den Stift wieder an.

Auswahl der Stiftfarbe.

Die Stiftfarbe kann um den eingegebenen Wert geändert werden.

Die Stiftfarbe auf 0=rot bis 100=blau setzen.

Die Intensität der Farbe um den eingegebenen Wert ändern.

Die Intensität der Farbe auf den eingegebenen Wert setzen.

Die Stiftdicke um den eingegebenen Wert ändern.

Die Stiftdicke auf einen Wert festsetzen.

Einen „Stempel“ oder Abdruck des Objekts auf der Bühne hinterlassen.



Addiert zwei Zahlen.

Subtrahiert zwei Zahlen.

Multipliziert zwei Zahlen.

Dividiert die erste Zahl durch die zweite.



Ergibt eine Zufallszahl aus dem angegebenen Bereich.



Wenn der erste Wert kleiner ist als der zweite, wird „wahr“ ausgegeben.

Wenn der erste Wert gleich dem zweiten ist, wird „wahr“ ausgegeben.

Wenn der erste Wert größer ist als der zweite, wird „wahr“ ausgegeben.



Wenn beide Bedingungen erfüllt sind, wird „wahr“ ausgegeben.

Wenn eine der beiden Bedingungen erfüllt ist, wird „wahr“ ausgegeben.

Wenn die Bedingung nicht erfüllt ist, wird „wahr“ ausgegeben.



Verbindet zwei Eingaben.

Gibt das erste Zeichen von Welt aus.

Gibt die Länge der Eingabe aus.



Gibt den Rest einer Division der ersten Zahl durch die zweite aus.

Rundet die Zahl auf die nächste Ganzzahl.



Errechnet die Wurzel aus dem eingegeben Wert.



Mathematische Operatoren



Eine Variable anlegen.

Eine angelegte Variable löschen.

Die Variable „Beispiel“ und den Wert auf der Bühne zeigen.

Die Variable auf den Wert 0 setzen.

Die Variable um den eingegebenen Wert ändern.

Die Variable auf der Bühne zeigen und verstecken.

Eine neue Liste anlegen und löschen.

Liste mit Namen „test“.

Füge etwas zur Liste „test“ hinzu.

Entferne den ersten Eintrag, einen beliebigen oder alles von der Liste.

Füge „etwas“ als erstes, beliebiges oder als letztes zur Liste.

Ersetze das erste, ein beliebiges oder das letzte Element von der Liste „test“ durch „etwas“.

Zeigt ob das erste, ein beliebiges oder das letzte Element von der Liste „test“ ist.

Zeigt wie viele Einträge in der Liste sind.

Berichtet ob „etwas“ in der Liste vorkommt.

Kurzer persönlicher Erfahrungsbericht.

Ich unterrichte an einer Haupt- und Realschule Informatik und Musik. Wir haben von der 6. Bis zur 10 Klasse Informatik als Wahlpflichtkurs zweistündig.

Scratch habe ich in allen Stufen bereits mit großem Erfolg erprobt. Für die Klassen 6 war es kein Problem sich in kürzester Zeit mit dem Programm zurechtzufinden. Einfache Animationen, Trickfilme und Sound ist hier das Reizvolle für die Schülerinnen und Schüler. In Klasse 7 haben wir dann thematisch gearbeitet. „Erstellt eine kleine Animation zum Thema Weltraum“. Kleine Comics und einfache Spiele werden erstellt. In der Klasse 8 haben wir dann etwas komplexere Spiele erstellt. Variablen und Operatoren. Auch zeichnen in drei Dimensionen konnte so geübt werden.

Klasse 9 und 10 erstellen dann Projekte. Alle Erfahrungen aus anderen Spielen werden zusammen getragen und wir versuchen ein oder zwei große Spiele zu erstellen.

Dazu kommt ab Klasse 7 dann schon die Fernsteuerung. Lego WeDo und ab Klasse 8 das Picoboard.

Die Schüler haben sehr viel Spaß an dem Programm und auch viele Ideen. Ein gelungener Einstieg. Natürlich arbeiten wir auch mit anderer Software in Informatik, dennoch kann man sagen: „Scratch kommt gut an“.

Ich habe eine Programmier-AG eingerichtet. Überwiegend kommen Schüler der Klasse 6 und 7. Auch hier „scratches“ wir was das Zeug hält.

Die Internetseite von Scratch finde ich ein wenig überladen. Es ist zu viel und auch recht wenig Überzeugendes – in Relation gesehen - dabei.

Ich veröffentliche die Projekte auf unserer internen Informatikhomepage.

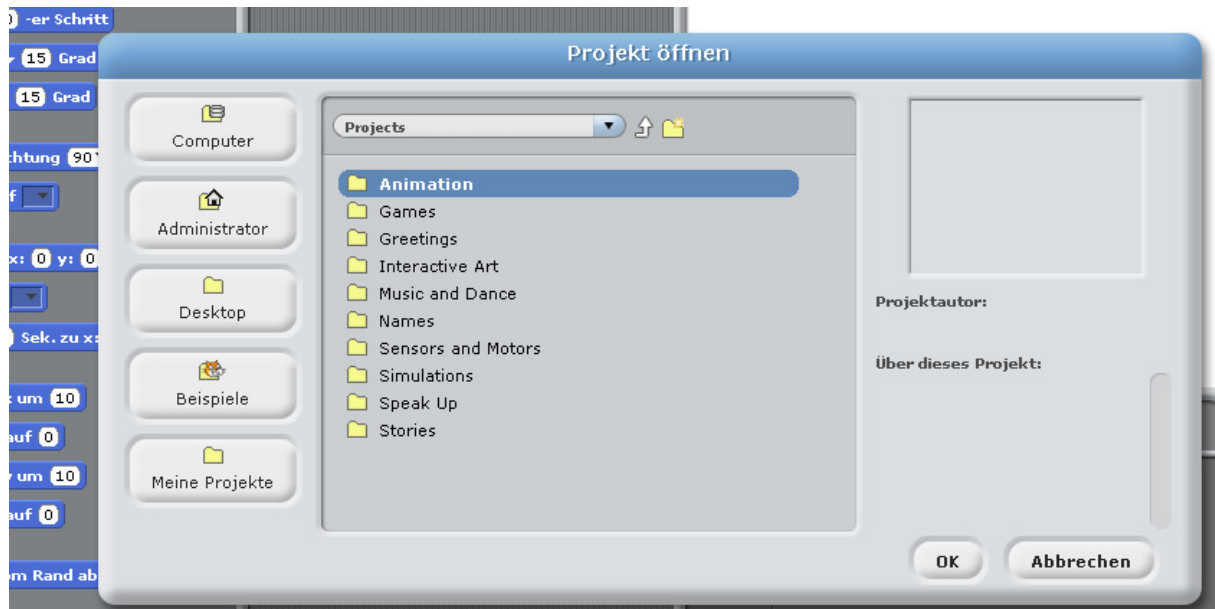
Frank Engeln

Aufgaben zu Scratch

Um das Programm kennen zu lernen sollst du erst einmal Beispiele ansehen.

Aufgabe:

1. Starte das Programm und sehe die Beispiele an. Du findest diese unter: Projekt öffnen – Beispiele - *(Order). (Zeit ca. 20 Minuten)

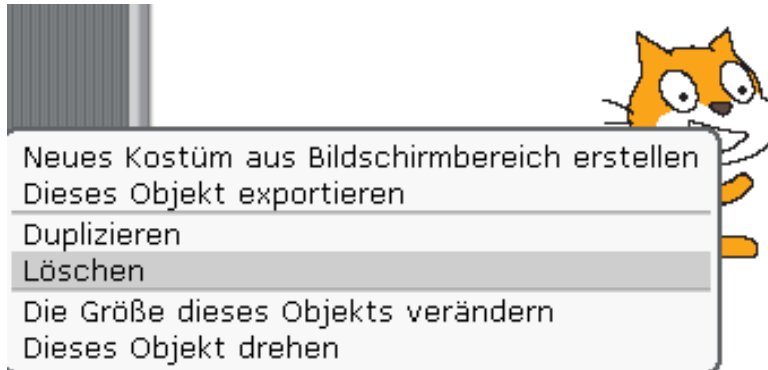


2. Schreibe auf, welches Projekt dir besonders gut gefällt. Begründe!
3. Könnt ihr euer gewähltes Beispiel beschreiben – was passiert und wie ist es aufgebaut?
4. Vergleicht eure Ergebnisse in der Klasse untereinander. Gibt es besonders beliebte Beispiele? Wenn ja, warum?

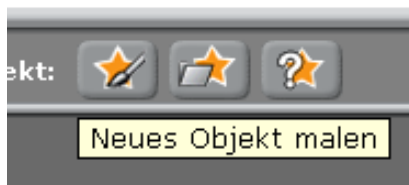
Einfache Bewegung einer Kugel (Kreis)

Um ein Objekt zu bewegen zeichnen wir erst einmal eine Kugel.

Aufgabe 1: Lösche die Katze auf der Bühne indem du mit der rechten Maus auf das Objekt klickst. Wähle „Löschen“ und bestätige dieses mit einem Linksklick.



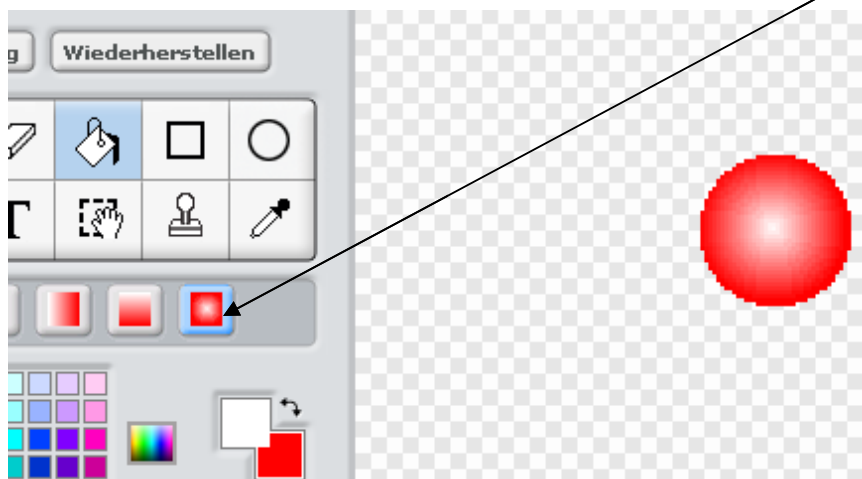
Klicke auf den linken Stern „Neues Objekt malen“



Es öffnet sich der Editor. Wähle mit der Maus den Kreis und halte die „Shift“ Taste während des zeichnens um einen Kreis zu erstellen (ohne „Shift“ zeichnet man eine Ellipse).



Nun ist der Kreis gezeichnet. Fülle ihn nun mit deiner Lieblingsfarbe als Farbverlauf von innen nach außen.



Aufgabe 2: Lasse die Kugel von links nach rechts über die Bühne „fliegen“.

Schreibe das Programm ab und ändere die Werte.



Das Programm startet, wenn die grüne Flagge angeklickt wird.

Eine Schleife die der Kugel sagt, mache 5 Mal einen 10er Schritt nach rechts.

Eine Schleife die der Kugel sagt, mache 5 Mal einen 10er Schritt nach links.

Aufgabe 3:



Die Kugel soll sich im Quadrat über die Bühne bewegen. Dazu nimmt man eine Drehung von 90⁰ Grad. Um die Ausgangsrichtung zu erhalten klicke auf



Aufgabe 4:

Unsere Kugel soll nun etwas zeichnen. Dazu soll sie immer an einem bestimmten Punkt anfangen. Wir können den Punkt mit den x und y Koordinaten angeben. Ebenso die Richtung, in die das Objekt „gucken“ soll.



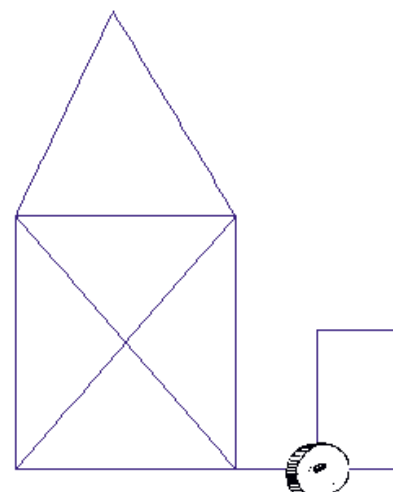
Um ein Objekt etwas zeichnen zu lassen benötigen wir den Malstift. Anhand der Koordinaten können wir bestimmen, zu welchem Punkt der Kreis zeichnen soll.

senke Stift ab

gehe zu x: -180 y: -130

Versuche nun das „Haus vom Nikolaus zeichnen zu lassen.

Für Fortgeschrittene: Erstelle ein Oktaeder.

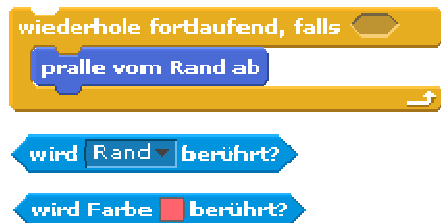


Lösungsmöglichkeit „Haus vom Nikolaus“:

```
Wenn  angeklickt
wische Malspuren weg
setze x auf -85
setze y auf -145
wische Malspuren weg
warte 1 Sek.
wiederhole 1 mal
  senke Stift ab
  setze Stiftfarbe auf 
  gleite 1 Sek. zu x: 50 y: -145
  gleite 1 Sek. zu x: 50 y: 10
  gleite 1 Sek. zu x: -85 y: 10
  gleite 1 Sek. zu x: -25 y: 135
  gleite 1 Sek. zu x: 50 y: 10
  gleite 1 Sek. zu x: -85 y: -145
  gleite 1 Sek. zu x: -85 y: 10
  gleite 1 Sek. zu x: 50 y: -145
  →
gleite 1 Sek. zu x: 150 y: -145
gleite 1 Sek. zu x: 150 y: -60
gleite 1 Sek. zu x: 100 y: -60
gleite 1 Sek. zu x: 100 y: -145
```

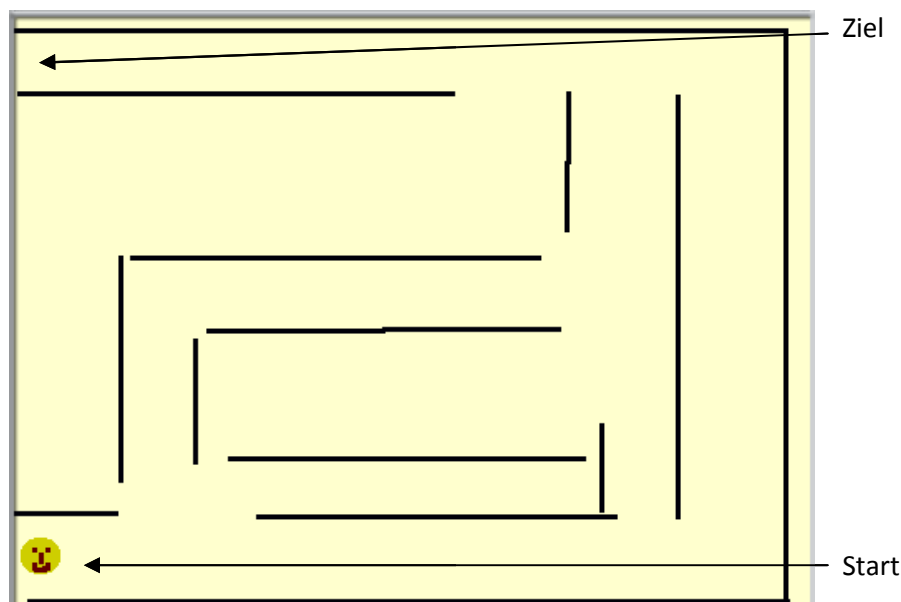

Pralle vom Rand

Aufgabe: Zeichne eine „Mauer“ um deine Bühne in Rot. Die Mauer ist ein dicker roter Strich. Wenn sich dein Kreis bewegt, soll er vom Rand abprallen. Nutze folgende Bausteine zusätzlich:



Aufgabe: Ändere das Programm indem der Kreis seine Farbe ändert, wenn er eine bestimmte Farbe (des Randes) berührt oder das Programm beendet wird.

Aufgabe: Erstelle ein Labyrinth und ein Objekt, welches durch das Labyrinth gesteuert wird.



Die Figur soll mit den Pfeiltasten gesteuert werden. Folgende Bausteine (und einige mehr) brauchst du:

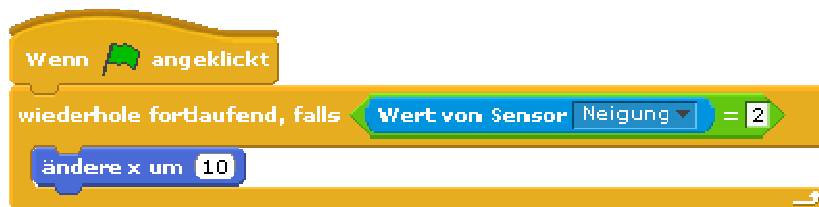


Bewegung mit Lego WeDo

Wenn du die ersten Aufgaben gemacht hast, schließe den USB Hub von Lego WeDo an den Computer an. Verbinde den Neigungssensor mit dem USB Hub.

Aufgabe 1: Ändere die Aufgabe 2 „Lasse die Kugel von links nach rechts über die Bühne „fliegen““ so ab, dass du die Steuerung mit dem Neigungssensor machst.

Dazu braucht man folgende Bausteine:



Aufgabe 2: Zeichne ein Quadrat mit dem Neigungssensor.

Aufgabe mit dem Abstandssensor

Aufgabe: Vergrößere und verkleinere den Kreis mit dem Abstandssensor.

Der Kreis aus Aufgabe 1 soll bei Annäherung größer und beim Abstand kleiner werden. Ähnlich wie beim Neigungssensor brauchst du nur folgende Bausteine:



Steuerung mit dem Picoboard

Aufgabe: Vergrößere und verkleinere den Kreis mit dem Regler.

Stelle den Regler in die Mitte des Boards (50.0).

Der Kreis aus Aufgabe 1 soll bei „> 55“ größer und beim Wert „< 45“ kleiner werden. Zwischen Beim Wert 50.0 bleibt der Kreis in der alten Größe Du brauchst nur folgende Bausteine:



Aufgabe: Ändere das Programm so ab, dass du die Kreisgröße mit anderen Sensoren steuerst.

Aufgabe: Steure die Richtung auf der x-Achse mit dem Soundsensor, die Steuerung mit dem Regler auf der y-Achse und stoppe das Script beim Druck auf die Taste.

