

# Objektorientiertes Modellieren mit SMALLTALK/SQUEAK

Beispiele für den Anfangsunterricht  
Teil 1

von Rüdiger Baumann

Die Informatik (in der Schule) ist keine Lehre, sondern eine Tätigkeit.  
*in Anlehnung an: Wittgenstein (1918), 4.112*

Wer der Empfehlung der *Bildungsstandards für die Sekundarstufe I* folgt, die objektorientierte Sichtweise „als durchgängiges, grundlegendes Prinzip“ anzuwenden (AKBSI, 2008, S.27), und außerdem die Grundsätze konstruktivistischen Lernens ernst nimmt, kommt (neben LOGO) um SMALLTALK bzw. SQUEAK nicht herum. Nachdem für letzteres in der Etoys-Version bereits etliche didaktische Publikationen vorliegen (z.B. Allen-Conn/Rose, 2009), ist es an der Zeit, auch SQUEAK in seiner Eigenschaft als SMALLTALK-Dialekt für den Anfangsunterricht zu erschließen.

Das im Folgenden empfohlene Vorgehen stellt einerseits eine Alternative zu den derzeit gängigen Lehrbüchern (die den Anfangsunterricht mit Standardsoftware bestreiten und den Einsatz einer Programmiersprache erst für Jahrgangsstufe 9 oder 10 vorsehen) dar – und bietet andererseits die Möglichkeit, die Bildungsstandards, die ja bekanntlich keinen unterrichtlichen Weg vorschreiben, mit Leben zu erfüllen.

---

## Voraussetzungen und Ziele

Es wird vorausgesetzt, dass Schüler und Schülerinnen (Jahrgangsstufe 5 oder 6) bereits ein halbes Schuljahr mit Etoys gearbeitet haben und einige der „mächtigen Ideen“ (*powerful ideas*) gemäß Seymour Papert und Alan Kay verinnerlicht haben (siehe Allen-Conn/Rose, 2009, unter anderem auf Seite V). Im Einzelnen nehmen wir an, dass die Lernenden über folgende Kompetenzen verfügen: Sie können

- ▷ Objekte mithilfe des Malwerkzeugs erstellen, benennen und handhaben (verschieben, vergrößern/verkleinern, duplizieren etc.);
- ▷ ein Skript als Folge von Anweisungen an ein (selbsterzeugtes oder bereits vorhandenes) Objekt durch Aneinanderfügen von Kacheln aufbauen, testen und erläutern;
- ▷ die algorithmischen Grundbausteine „Test“ (Verzweigung mit einer einfachen Bedingung) und „Wiederhole n-mal“ (Zählschleife) in einem Skript erfolgreich verwenden;
- ▷ Variablen, Parameter und Wertzuweisungen in einfachen Beispielen verständlich und zielgerichtet einsetzen.

Ferner wird angenommen, dass die Lernenden über einen intuitiven Objektbegriff verfügen, wie er sich beim Umgang mit den Gegenständen der Lebenswelt parallel zum Spracherwerb einstellt. Sie können insbesondere zwischen Objekten (im Sinne von Dingen, Gegenständen, Lebewesen, Geräten usw.) der realen Welt und solchen „im Computer“ unterscheiden. In diesem naiven Sinn denken sie sowohl *algorithmisch-imperativ* (Objekte verstehen Befehle), *deskriptiv-prädikativ* (man kann Objekte durch Eigenschaften und Fähigkeiten beschreiben) und *objektorientiert* (richtet man an ein Objekt eine Bitte, so weiß dieses – sofern es die Bitte versteht – von selbst, was es zu deren Erfüllung tun muss).

Die Lernenden, so wird unterstellt, verstehen die einschlägigen Begriffe (z.B. auch „Problem“ und „Programm“) handlungsorientiert, d.h. nicht aufgrund einer Begriffsdefinition, sondern durch erfolgreiches zielgerichtetes Tun; sie weisen das Verstehen der Begriffe durch korrekten Wortgebrauch nach. Objekte können in SQUEAK als Prototypen erzeugt werden, d.h. man benötigt zu deren Bildung keine Klassen und damit auch nicht den Klassenbegriff. Der tritt erst in einer späteren Phase, wenn die Abstraktionsfähigkeit weiter ausgebildet ist, in Aktion.

## Computereinsatz zum gestaltenden Lernen

Die didaktisch-methodische Alternative des hier empfohlenen Vorgehens zu dem der gängigen Lehrbücher besteht darin, dass die Lernenden Objekte nicht als Erklärungsmittel für Anwendungsprogramme, sondern als etwas erfahren, das man erschaffen und gestalten, d.h. mit Eigenschaften und Fähigkeiten ausstatten kann, um damit interessante (selbstgestellte oder vorgegebene) Aufgaben zu lösen. „Wir ließen uns von der Forderung leiten, Kinder bereits im frühen Alter zu intensiver Beschäftigung anzuregen, ihnen Möglichkeiten zum aktiven Tätigsein und zum vielseitigen Umgang mit Spielzeug zu geben. Auch der Computer mit geeigneten Programmen kann ein Spielzeug in diesem Sinne sein“, heißt es in einem Projekt *Computereinsatz zum Knobeln und Gestalten*, das im Jahr 1988 in Leipzig mit Vorschulkindern durchgeführt wurde (Gutzer/Quapp, 1990, S.48).

Entscheidend ist hier natürlich, was man unter „geeigneten Programmen“ versteht. Für einen an Bildungsstandards orientierten Informatikunterricht „ist es wichtig, dass Werkzeuge eingesetzt werden, die das Augenmerk auf die grundlegenden Aspekte der objektorientierten Modellierung lenken und die es erlauben, wenige einfache syntaktische Mittel auf vielfältige Weise zu kombinieren“ (Füller, 2004, S.43). Papert greift in seiner Lerntheorie des Konstruktivismus das Prinzip des Lernens als Konstruktionsprozess auf, fügt aber die Idee hinzu, dass Lernen durch das *eigenhändige Konstruieren eines Artefakts* (z.B. eines Computerprogramms) unterstützt wird. Der Vorteil des Schaffens eines realen Produkts ist die Möglichkeit, dieses auszuprobieren, es zu zeigen, zu diskutieren und bewundern zu lassen (vgl. Zahn, 2009). Mit solchen Erwägungen ist der Einsatz von SMALLTALK/SQUEAK gerechtfertigt. Glücklicherweise ist dann auch nicht mehr strittig, dass „schon von der ersten Stunde an in die objektorientierte Sichtweise eingeführt werden sollte“ (Koerber, 2004).

## Vom visuellen zum textuellen Programmieren

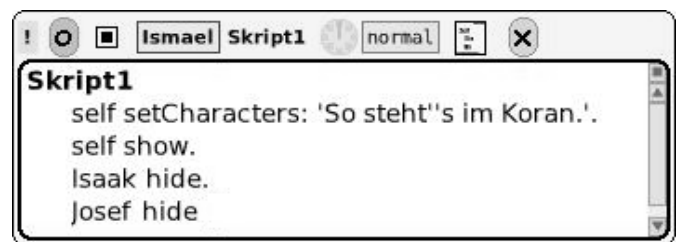
An die Kompetenzen im visuellen Programmieren (mit Etoys) anknüpfend, wird zunächst schrittweise



**Bild 1:** Text nach Anklicken des mittleren Knopfs (Isaak).



**Bild 2:** Ein Skript in Kachelform (oben) und in Textform (unten).



zum Schreiben von SMALLTALK-Programmen in Textform übergeleitet.

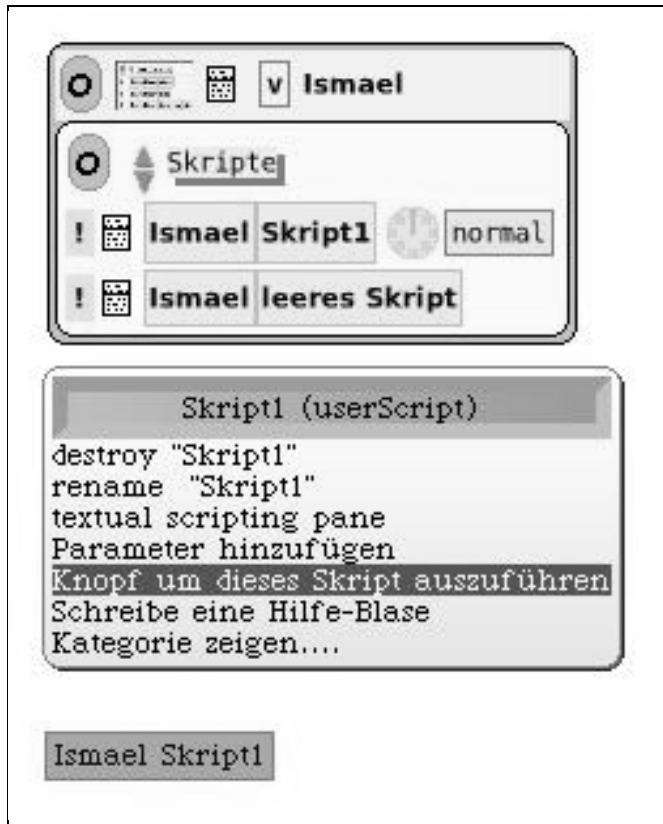
### Beispiel 1: Wissenstest im Fach Religion

Auf dem Bildschirm erscheint eine Frage mit drei möglichen Antworten (Bild 1). Die Benutzer sollen eine Antwort (durch Anklicken) auswählen, worauf eine Bestätigung oder Fehlermeldung folgt.

Wir ziehen acht Text-Objekte (ohne Rand) auf die Arbeitsfläche und ändern die ersten fünf gemäß Bild 1. Die letzten drei Texte dienen als Antworten; jeder Antwort muss nun ein Knopf zugeordnet werden, bei dessen Betätigung dann diese Antwort erscheint. Das erste Antwort-Objekt nennen wir *Ismael* und öffnen seinen Betrachter (Alt+Maus-Linksklick, türkisfarbenes Augensymbol). Dort finden wir eine Kachel mit der Aufschrift „Ismaels Buchstaben ← Text“: Diese fassen wir am Zuweisungspfeil und ziehen sie auf die Arbeitsfläche, worauf sich ein Skripteditor für das Objekt *Ismael* öffnet (Bild 2, oben).

Auf die gleiche Weise öffnen wir je einen Skript-Editor für die anderen beiden Antwort-Objekte namens *Isaak* und *Josef*. Nun können wir das Ismael-Skript durch die Kacheln *Isaak verstecke dich* und *Josef verstecke dich* ergänzen. Klicken wir im Kopf des Skripts auf das kleine Quadrat (an dritter Stelle von links), so verschwinden die Kacheln und der Programmtext erscheint (Bild 2, unten). Wie ist er zu interpretieren?

Das Wort *self* bezieht sich auf das „Objekt selbst“, hier also auf das Textobjekt *Ismael*. Die erste Zeile fordert das Textobjekt auf, eine gewisse Buchstabenfolge (in Hochkommas eingeschlossen) zu „setzen“; *setCharacters*: ist eine sogenannte *Methode* (Operation) mit zwei Argumenten. Die nächsten Zeilen bestehen aus Methoden-Aufrufen, die jeweils durch einen Punkt voneinander getrennt sind. Die Methoden *show* (= zeigen) und *hide* (=



**Bild 3: Menü und Aktivierungsknopf eines Skripts (Beispiel 1).**

verbergen) sind einstellig; das Argument (der Objekt-Name *self*, *Isaak*, *Josef*) steht jeweils links.

Wir öffnen auch bei den Objekten *Isaak* und *Josef* das Programmfenster und ergänzen die Programme entsprechend. Nach jeder Änderung eines Programmtextes ist die Tastenfolge *Strg+S* (Speichern) zu drücken oder der grüne *Ok*-Haken anzuklicken.

Schließlich werden die drei Antwortknöpfe eingefügt. Wir finden den *Ismael*-Knopf im Betrachter (Kategorie *Skripte*) beim Anklicken des Menü-Symbols (rechts vom gelben Ausrufezeichen), Option: „Knopf, um dieses Skript auszuführen“ (Bild 3); er bekommt anschließend runde Ecken und eine andere Farbe – Entsprechendes gilt für die anderen beiden Knöpfe.

**Aufgabe 1.1:** Entwirf und programmiere einen eigenen Wissenstest deiner Wahl. (Schreibe bzw. male die Texte, Fragen, Antworten zuerst von Hand auf ein Blatt Papier.)

Was haben wir Neues gelernt?

- ▷ Jedes Objekt kann gewisse Tätigkeiten ausführen; wir nennen sie *Methoden* dieses Objekts.
- ▷ Die Methoden eines Objekts werden im Betrachter als Kacheln angezeigt und können von dort in ein Skript (Programm) eingebunden werden.

- ▷ Ein Objekt kann ein anderes Objekt „ansprechen“, indem es eine Methode dieses Objekts aufruft. Dies geschieht dadurch, dass eine Kachel aus dem Betrachter des anderen Objekts ins eigene Skript eingefügt wird.
- ▷ Der Kachelform eines Skripts entspricht eine Darstellungsform als Text, in dem die Methodenaufrufe als Wörter einer Programmiersprache (SMALLTALK) aufgeschrieben sind.
- ▷ Der Aufruf einer Methode wird auch als *Nachricht* an das Objekt bezeichnet, zu dem die Methode gehört. Je zwei Nachrichten werden durch einen Punkt getrennt.

*Bemerkung:* Beim Vergleich „Kachelform – Textform“ eines Skripts handelt es sich um die Kompetenz „Bedeutung und Darstellungsform einer Nachricht unterscheiden“ (AKBSI, 2008, S.24). Man könnte die Einsicht: „Beide Darstellungsformen tragen die gleiche Information“ als Merktex anfügen; aber dieser Umstand ist den Schülern sowieso klar: Es wäre die (unnötige) Betonung einer Selbstverständlichkeit.

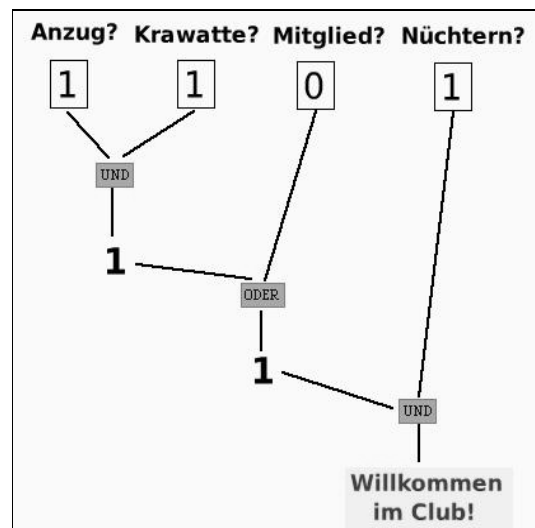
Beim nächsten Beispiel handelt es sich um einen Fall von *funktionaler Modellierung*: Ähnlich wie die arithmetischen Verknüpfungen ordnen auch logische Funktionen (hier: *und*, *oder*) je zwei (Wahrheits-)Werten einen dritten als Ergebnis zu.

## Beispiel 2: Clubzugang

Das Betreten des Golfclubs *Herzogstadt Celle* gestattet der als Türsteher engagierte Roboter nur Gästen, die entweder mit Anzug und Krawatte bekleidet sind oder eine Mitgliedskarte vorweisen können und nicht betrunken sind. Wie ist er zu programmieren?

Wir sehen vier Felder A, B, C, D vor, in die 1 (= wahr) oder 0 (= falsch) eingegeben werden kann:

- A = mit Anzug bekleidet,
- B = mit Krawatte bekleidet,
- C = hat Mitgliedskarte,
- D = ist nüchtern.



**Bild 4: Logikschaltung für den Clubzugang (Beispiel 2).**