GANZ EINFACH

Programmieren für Kinder

Mit Bausteinen und Spielen erste eigene Programme schreiben



Danilo Sieren

G A N Z E I N F A C H

Band 58

Programmieren für Kinder

Mit Bausteinen und Spielen erste eigene Programme schreiben

Danilo Sieren

GANZ EINFACH

Die Reihe GANZ EINFACH erklärt Technik und Alltagsthemen so, wie es eine gute Freundin oder ein guter Freund am Küchentisch tun würde: geduldig, ehrlich und mit vielen Beispielen. Jeder Band steht für sich, und wo ein anderer Band ein Thema vertieft, steht ein Hinweis im Text.

Impressum und Hinweise zur Entstehung stehen auf der letzten Seite dieses Buches.

Vorwort

Es beginnt oft mit einer Frage am Abendbrottisch. »Wie macht man eigentlich ein Spiel?« Oder: »Kann ich der Katze im Tablet beibringen, dass sie tanzt?« Oder, etwas ungeduldiger: »Warum macht der Computer nicht, was ich will?« Hinter diesen Fragen steckt der Wunsch, Technik nicht nur zu benutzen, sondern selbst zu gestalten. Genau darum geht es beim Programmieren.

Viele Eltern, Großeltern und Lehrkräfte fühlen sich bei diesem Thema unsicher. Sie haben selbst nie programmiert und fürchten, nicht helfen zu können. Die gute Nachricht: Du musst kein Informatiker sein, um ein Kind beim Programmierenlernen zu begleiten. Die Werkzeuge für Kinder sind so gestaltet, dass Erwachsene und Kinder gemeinsam einsteigen können. Bunte Bausteine ersetzen komplizierte Befehle, und die meisten Programme laufen kostenlos im Browser oder auf dem Tablet.

Dieses Buch zeigt dir den ganzen Weg: wie Kinder ohne Computer erste Grundideen des Programmierens erleben, wie sie mit ScratchJr und Scratch eigene Geschichten und Spiele bauen, wie kleine Platinen wie der Calliope mini Programme in die echte Welt bringen und wie ältere Kinder den Schritt zu Python wagen. Du lernst, welche Angebote sicher sind, wie du Konten und Daten schützt und wie du ein Kind motivierst, wenn es frustriert vor einem Fehler sitzt. Und wir sprechen darüber, was künstliche Intelligenz für das Programmierenlernen bedeutet.

Ich verspreche dir nicht, dass jedes Kind Programmierer wird. Das ist auch nicht das Ziel. Das Ziel ist, dass Kinder verstehen, dass hinter jeder App, jedem Spiel und jedem Roboter Menschen stehen, die Anweisungen geschrieben haben. Und dass sie selbst solche Anweisungen schreiben können. Diese Erfahrung verändert, wie man auf Technik blickt.

Durch das ganze Buch begleitet dich Familie Neumann aus Witten. Jana ist 38 und arbeitet als Pflegefachkraft im Schichtdienst. Sie hat nie programmiert und hält sich selbst für »nicht technisch«. Ihre Tochter Lotta, 8, liebt Geschichten, Tiere und ihre Katze Mücke. Ihr Sohn Emil, 11, spielt am liebsten Minecraft und will unbedingt ein eigenes Spiel bauen. Und Janas Vater Harald, 71, hat dreißig Jahre lang Programme für eine Versicherung in Dortmund geschrieben, auf Großrechnern, in einer Sprache namens COBOL. Er hält bunte

Bausteine zunächst für Spielerei. Gemeinsam entdecken die vier, was Programmieren heute für Kinder bedeuten kann.

Du brauchst kein technisches Vorwissen. Wenn du einen Browser öffnen und eine App installieren kannst, bringst du alles mit. Den Rest lernt ihr gemeinsam.

Setz dich neben ein Kind. Wir fangen an.

Danilo Sieren, Dortmund, im November 2026

So arbeitest du mit diesem Buch

Dieses Buch richtet sich an Erwachsene, die Kinder beim Programmierenlernen begleiten: Eltern, Großeltern, Lehrkräfte, Leiterinnen von Arbeitsgemeinschaften. Ältere Kinder ab etwa zwölf Jahren können viele Kapitel auch selbst lesen.

Die ersten Kapitel erklären, warum Programmieren für Kinder wertvoll ist, was ein Programm überhaupt ist und wie man die Grundideen ganz ohne Computer erlebt. Danach geht es um den sicheren Rahmen: Geräte, Konten, Datenschutz und Bildschirmzeit. Im Mittelteil lernst du mit ScratchJr und Scratch die wichtigsten Bausteine kennen, von Bewegungen über Schleifen und Bedingungen bis zu Variablen, und baust mit dem Kind erste Spiele und Geschichten. Die zweite Hälfte führt zu kleinen Computern wie dem Calliope mini, zu Robotern, zu Minecraft Education, zu Python und zu künstlicher Intelligenz. Zum Schluss geht es um Lerngruppen, Wettbewerbe und die Rolle der Erwachsenen.

Wenn das Kind schon mit Scratch vertraut ist, kannst du mit Kapitel 16 beginnen. Lies aber unbedingt Kapitel 5 über Sicherheit und Datenschutz.

Die Bausteine jedes Kapitels

Jedes Kapitel beginnt mit einer Erklärung in ganz normaler Sprache. Fachbegriffe erkläre ich beim ersten Auftreten, und im Glossar in Anhang A findest du alle noch einmal.

In der **Werkstatt** setzt ihr das Gelernte Schritt für Schritt um. Die Schritte sind nummeriert und so beschrieben, dass du sie mit dem Kind direkt am Rechner oder Tablet nachvollziehen kannst.

Die Kästen mit der Überschrift **Vertiefung** gehen tiefer in die Technik. Beim ersten Lesen kannst du sie überspringen. **Achtung**-Kästen warnen vor Fallen, die Daten, Geld oder Nerven kosten können. **Tipp**-Kästen enthalten Abkürzungen aus der Praxis. In den **Lernprojekt**-Kästen begleitest du Familie Neumann.

Am Ende jedes Kapitels beantworte ich **Häufige Fragen** und fasse in **Kurz gesagt** das Wichtigste zusammen.

Wie Bausteine im Buch aussehen

Programme aus Bausteinen lassen sich auf Papier schlecht abbilden. In diesem Buch schreibe ich sie deshalb als Text in einem grauen Kasten. Jede Zeile steht für einen Baustein, eingerückte Zeilen gehören in den Baustein darüber, etwa in eine Schleife:

```
Wenn die grüne Flagge angeklickt wird
wiederhole 10-mal
    gehe 10er-Schritt
    drehe dich um 15 Grad
```

In Scratch sehen diese Bausteine farbig aus und rasten ineinander ein. Die Namen der Bausteine entsprechen der deutschen Oberfläche.

Programme und Versionen

In diesem Buch nenne ich Programme, Geräte und Angebote bekannter Anbieter als Beispiele. Das ist keine Werbung, sondern soll dir die Orientierung erleichtern. Fast alle vorgestellten Programme sind kostenlos. Dieses Buch beschreibt den Stand vom November 2026. An einer neuen Version von Scratch wird gearbeitet. Menüs können bei dir anders aussehen, die Grundideen bleiben gleich.

> **ACHTUNG** **Datenschutz zuerst**
>
> Viele Angebote für Kinder verlangen ein Konto. Lies Kapitel 5, bevor du ein Kind irgendwo anmeldest. Für die ersten Schritte braucht ihr kein Konto: Scratch, ScratchJr und viele andere Werkzeuge funktionieren auch ohne Anmeldung.

Inhalt

Vorwort....3
So arbeitest du mit diesem Buch....5
Warum Kinder programmieren lernen....9
Was ein Programm eigentlich ist....16
Programmieren ohne Computer....22
Der richtige Einstieg für jedes Alter....28
Ein sicherer Rahmen: Geräte, Konten, Datenschutz....34
ScratchJr für die Kleinsten....40
Scratch kennenlernen....46
Figuren bewegen: die Bühne als Koordinatensystem....53
Schleifen: wenn etwas immer wieder passiert....59
Bedingungen: wenn, dann, sonst....65
Variablen: Punkte, Leben und Zeit....71
Ereignisse und Nachrichten....77
Das erste eigene Spiel....83
Geschichten, Animationen und Quiz....89
Musik und Klang programmieren....95
Fehler finden: Debuggen mit Geduld....101
Die Scratch-Gemeinschaft....107
Calliope mini und micro:bit....113
Sensoren und kleine Erfindungen....120
Roboter programmieren....126
Minecraft und Spielewelten....132
Der Schritt zur Textsprache: Python....138
Zeichnen mit der Python-Schildkröte....145
Kleine Python-Projekte....152
Künstliche Intelligenz verstehen und nutzen....159
Gemeinsam lernen: Kurse, Clubs und Wettbewerbe....165
Erwachsene als Begleiter....171
Wie es weitergehen kann....177
Glossar....183
Bausteine auf einen Blick....187
Übersichten....189
Musik und Zahlen....191
Vorlagen....193
Checklisten....195
Fragen zum Selbsttest....197
Problemlöser....199

Schritt für Schritt: Das Sammelspiel in Scratch 201
Wichtige Adressen 202
Englische Begriffe 203
Nachwort 205
Quellen und weiterführende Links 206
Ausführliches Inhaltsverzeichnis 208
Stichwortverzeichnis 222

KAPITEL 1

Warum Kinder programmieren lernen

Kinder wachsen mit Tablets, Smartphones und Spielkonsolen auf. Sie wischen, tippen und spielen mit großer Selbstverständlichkeit. Wie diese Geräte funktionieren, bleibt ihnen aber meist verborgen. Programmieren öffnet diese verschlossene Tür. Dieses Kapitel zeigt, was Kinder dabei lernen, was es nicht leisten kann und warum der Einstieg leichter ist, als viele Erwachsene denken.

Vom Benutzen zum Gestalten

Ein Kind, das ein Spiel spielt, folgt den Regeln, die jemand anderes festgelegt hat. Ein Kind, das ein Spiel programmiert, legt die Regeln selbst fest. Es entscheidet, wie schnell die Figur läuft, was passiert, wenn sie einen Stern berührt, und wann das Spiel zu Ende ist. Dieser Wechsel vom Benutzen zum Gestalten ist der eigentliche Kern des Programmierenlernens.

Wer einmal erlebt hat, dass eine Figur auf dem Bildschirm genau das tut, was man ihr aufgetragen hat, sieht Technik anders. Apps und Spiele erscheinen nicht mehr als Zauberei, sondern als etwas, das Menschen gemacht haben und das man verstehen, kritisieren und verbessern kann.

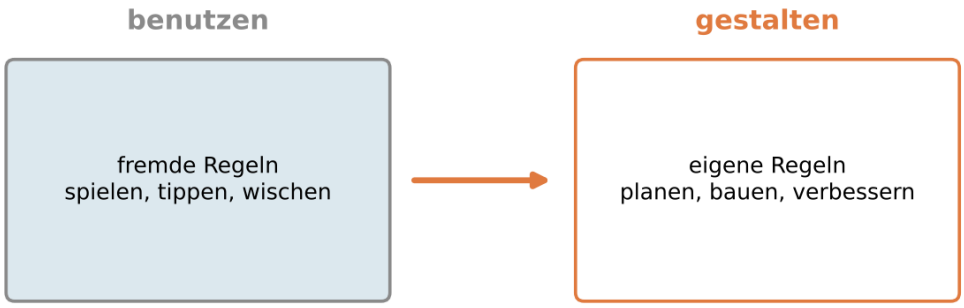
Was Kinder dabei lernen

Programmieren schult Fähigkeiten, die weit über den Computer hinausreichen:

- **Probleme zerlegen:** Ein großes Vorhaben wie »Ich will ein Spiel bauen« wird in kleine Schritte aufgeteilt.
- **Genau denken:** Ein Computer versteht keine ungefähren Anweisungen. Kinder lernen, präzise zu formulieren.
- **Muster erkennen:** Wiederholungen werden zu Schleifen, Entscheidungen zu Bedingungen.
- **Ausprobieren und verbessern:** Ein Programm funktioniert selten auf Anhieb. Kinder lernen, Fehler als Teil der Arbeit zu sehen.
- **Geduld und Ausdauer:** Ein Fehler, der nach zwanzig Minuten gefunden ist, macht stolz.

- **Ausdruck und Kreativität:** Geschichten, Musik, Kunst und Spiele entstehen aus eigenen Ideen.
- **Zusammenarbeit:** Projekte werden geteilt, verändert und gemeinsam verbessert.

Fachleute fassen viele dieser Fähigkeiten unter dem Begriff **informatisches Denken** zusammen, im Englischen Computational Thinking. Gemeint ist die Fähigkeit, Probleme so zu beschreiben, dass ein Mensch oder eine Maschine sie Schritt für Schritt lösen kann.



Vom Benutzen zum Gestalten: Wer programmiert, legt die Regeln selbst fest, statt nur fremden Regeln zu folgen.

Was Programmieren nicht ist

Einige Missverständnisse halten sich hartnäckig:

- **Programmieren ist keine Mathematik.** Zahlen kommen vor, aber für den Einstieg reicht das Einmaleins. Viele Kinder, die in Mathematik Mühe haben, programmieren mit Begeisterung.
- **Programmieren ist nichts für Einzelgänger.** Die meisten Projekte entstehen im Austausch, und Kinder zeigen sich ihre Werke gern gegenseitig.
- **Programmieren heißt nicht mehr Bildschirmzeit ohne Sinn.** Wer programmiert, gestaltet aktiv. Das ist etwas anderes als stundenlanges Anschauen von Videos. Trotzdem braucht es Pausen, wie Kapitel 5 zeigt.
- **Programmieren ist nicht nur für Jungen.** Mädchen programmieren genauso gut und gern, wenn die Projekte sie ansprechen und niemand ihnen das Gegenteil einredet.

Programmieren und künstliche Intelligenz

Seit KI-Systeme selbst Programmcode schreiben können, fragen sich viele Eltern, ob Kinder überhaupt noch programmieren lernen müssen. Die Antwort ist ja, aus einem einfachen Grund: Wer verstehen will, was eine KI tut, warum sie Fehler macht und wie man sie sinnvoll einsetzt, braucht ein Grundverständnis davon, wie Programme funktionieren. Wer nie selbst eine Schleife gebaut hat, kann schlecht beurteilen, ob der Vorschlag einer KI richtig ist. Kapitel 25 geht ausführlich darauf ein.

Fähigkeit	beim Programmieren	im Alltag
Probleme zerlegen	Spiel in Figuren und Regeln aufteilen	Hausaufgaben planen
genau formulieren	Befehle in richtiger Reihenfolge	Wegbeschreibung geben
Muster erkennen	Wiederholungen als Schleife	Rechenwege verkürzen
Fehler suchen	Programm prüfen	Fahrrad reparieren
durchhalten	nach Fehlern weitermachen	Instrument üben

VERTIEFUNG **Woher die Idee kommt**

Die Idee, dass Kinder programmieren lernen sollten, ist älter als der Heimcomputer. In den 1960er-Jahren entwickelte der Mathematiker Seymour Papert am Massachusetts Institute of Technology die Programmiersprache **Logo**, mit der Kinder eine kleine Schildkröte über den Boden oder den Bildschirm steuerten. Papert war überzeugt, dass Kinder am besten lernen, wenn sie selbst etwas erschaffen. Diese Haltung prägt bis heute viele Angebote. Scratch, das bekannteste Programm für Kinder, entstand 2007 in derselben Forschungsgruppe und trägt Paperts Gedanken weiter.

LERNPROJEKT **Die Frage am Abendbrottisch**

Bei Familie Neumann in Witten gibt es Nudeln, und Emil redet nur von Minecraft. Ein Freund hat ihm erzählt, dass er ein eigenes Spiel programmiert hat. »Wie geht das?«, fragt Emil. Jana zuckt mit den Schultern. Sie arbeitet im Nachtdienst auf einer Station, kennt sich mit Infusionspumpen aus, aber nicht mit Programmen. Lotta will wissen, ob man Mücke, der Katze, im Tablet das Tanzen beibringen kann. Am Sonntag erzählt Jana ihrem Vater davon. Harald hat dreißig Jahre lang Programme für eine Versicherung geschrieben. Er lacht. »Spiele? Wir haben Beiträge berechnet.«

> Dann wird er nachdenklich. »Aber zeig mal, was die Kinder da heute machen.«

Programmieren in Schule und Lehrplan

In Deutschland hat sich in den letzten Jahren viel bewegt. Immer mehr Bundesländer führen Informatik als Pflichtfach in der Sekundarstufe I ein, oft ab der fünften oder siebten Klasse. In einigen Ländern gibt es Programmierprojekte bereits im Sachunterricht der Grundschule, etwa mit dem Calliope mini, der eigens dafür entwickelt wurde. Die Kultusministerkonferenz hat Kompetenzen für die Bildung in der digitalen Welt beschrieben, zu denen auch das Verständnis von Algorithmen gehört.

Für Eltern heißt das: Viele Kinder kommen in der Schule mit Programmieren in Berührung, aber oft nur für wenige Stunden. Wer zu Hause Interesse weckt und begleitet, gibt dem Kind einen Vorsprung und vor allem die Erfahrung, dass Programmieren Spaß macht und nicht nur eine Schulaufgabe ist.

Was Kinder über sich selbst lernen

Neben technischen Fähigkeiten lernen Kinder beim Programmieren viel über sich selbst. Sie erleben, dass sie an einer Aufgabe dranbleiben können, auch wenn sie schwer ist. Sie erfahren, dass Fehler nicht peinlich sind, sondern Hinweise. Sie entdecken, dass sie eine Idee von Anfang bis Ende umsetzen können. Gerade Kinder, die in der Schule wenig Erfolgserlebnisse haben, blühen manchmal auf, wenn sie merken, dass ihr Spiel funktioniert und andere es spielen wollen.

Eltern berichten oft, dass Kinder durch das Programmieren geduldiger werden und lernen, Probleme in Schritte zu zerlegen, auch bei Hausaufgaben oder beim Aufräumen. Ob dieser Effekt bei jedem Kind eintritt, ist offen. Schaden kann die Erfahrung aber nicht.

Ein Blick in die Zukunft der Kinder

Niemand weiß, welche Berufe es in zwanzig Jahren geben wird. Sicher ist aber, dass Computer, Programme und künstliche Intelligenz in fast allen Bereichen eine Rolle spielen werden: in der Pflege, im Handwerk, in der Landwirtschaft, in der Kunst. Wer versteht, wie diese Systeme grundsätzlich funktionieren, kann sie besser nutzen, kritischer hinterfragen und mitgestalten. Programmieren ist des-

halb weniger Berufsvorbereitung als Allgemeinbildung, ähnlich wie Lesen von Karten oder Grundwissen in Biologie.

Ebenso wichtig ist eine andere Erfahrung: dass man Dinge selbst herausfinden kann. Kinder, die erleben, dass sie mit Geduld und Ausprobieren ein Problem lösen, nehmen diese Haltung in viele Lebensbereiche mit.

Was Eltern häufig befürchten

Viele Eltern haben Bedenken, bevor sie ihr Kind programmieren lassen. Einige der häufigsten und was dazu zu sagen ist:

»Mein Kind sitzt schon genug vor dem Bildschirm.« Das ist ein berechtigter Einwand. Programmieren muss nicht zusätzliche Bildschirmzeit bedeuten, es kann bestehende Zeit ersetzen, etwa Videos oder Spiele. Außerdem gibt es viele Übungen ohne Bildschirm, wie Kapitel 3 zeigt.

»Ich kann nicht helfen, weil ich es selbst nicht kann.« Du musst nicht helfen können, um zu begleiten. Kinder brauchen vor allem Interesse, Zeit und jemanden, der zuhört. Die technischen Fragen löst ihr gemeinsam oder mit den vielen Hilfen im Internet und in diesem Buch.

»Das ist doch nur Spielerei.« Spielen ist die Art, wie Kinder lernen. Wer ein Spiel programmiert, übt dieselben Denkweisen wie ein Profi, nur mit kindgerechten Werkzeugen.

»Mein Kind wird dadurch zum Stubenhocker.« Programmieren ist eine Beschäftigung unter vielen. Mit klaren Regeln und viel Abwechslung bleibt genug Zeit für Bewegung, Freunde und Natur. Viele Projekte, etwa mit Sensoren oder Robotern, führen sogar nach draußen.

Ein erster Schritt in fünf Minuten

Wer sofort loslegen möchte, braucht weder Konto noch Installation. Öffne im Browser die Seite von Scratch, klicke auf Entwickeln und lass das Kind die Katze mit der Maus über die Bühne ziehen. Dann ziehen beide gemeinsam den Baustein »gehe 10er-Schritt« in die Mitte und klicken darauf. Die Katze bewegt sich. Noch ein Klick, noch ein Schritt. Jetzt die Zahl auf 50 ändern. Diese fünf Minuten genügen oft, um Neugier zu wecken. Der Rest dieses Buches zeigt, wie es weitergeht.

Werkstatt: Ein Gespräch über Technik

1. Setz dich mit dem Kind zusammen und sucht drei Geräte im Haushalt, in denen ein Programm steckt, etwa Waschmaschine, Fernseher, Ampel.
2. Überlegt bei jedem Gerät, welche Anweisungen es befolgt: Was passiert, wenn man eine Taste drückt?
3. Frag das Kind, welches Spiel oder welche App es gern selbst bauen würde.
4. Schreibt die Idee auf ein Blatt Papier, mit Zeichnung.
5. Hängt das Blatt an einen gut sichtbaren Platz. Es ist das erste Ziel.

Häufige Fragen

Ab welchem Alter kann ein Kind programmieren? Erste Ideen können schon Kindergartenkinder spielerisch erleben. Mit Bausteinen am Tablet geht es ab etwa fünf Jahren, mit Scratch ab etwa acht. Kapitel 4 geht darauf ein.

Muss ich selbst programmieren können? Nein. Du lernst mit dem Kind. Oft ist es sogar gut, wenn das Kind dir etwas erklären kann.

Braucht man einen teuren Computer? Nein. Ein gewöhnlicher Laptop, ein Tablet oder ein älterer Rechner reicht für fast alle Angebote in diesem Buch.

Ist das nicht nur ein Trend? Programmieren ist seit Jahren Teil der Lehrpläne vieler Länder. In Deutschland führen immer mehr Bundesländer Informatik als Pflichtfach ein. Grundkenntnisse werden so selbstverständlich wie Lesen von Karten oder Rechnen mit Prozenten.

Lernen Kinder mit Programmieren besser rechnen? Programmieren übt logisches Denken und Zahlenverständnis, etwa bei Koordinaten und Winkeln. Ein Ersatz für den Mathematikunterricht ist es nicht, aber eine schöne Ergänzung.

Was, wenn ich selbst keine Lust auf Technik habe? Dann bring die Neugier für das Kind mit, nicht für die Technik. Es reicht, dabei zu sein und Fragen zu stellen.

Ist Programmieren nur etwas für kluge Kinder? Nein. Mit passenden Werkzeugen kann jedes Kind programmieren. Interesse und Geduld sind wichtiger als Schulnoten.

KURZ GESAGT **Das Wichtigste aus Kapitel 1**

Programmieren macht aus Kindern Gestalter statt bloßer Benutzer. Sie lernen, Probleme zu zerlegen, genau zu denken, Fehler zu finden und durchzuhalten. Programmieren ist keine reine Mathematik und nichts nur für Jungen. Auch im Zeitalter der KI bleibt das Grundverständnis wichtig. Erwachsene müssen keine Fachleute sein, um Kinder zu begleiten.

KAPITEL 2

Was ein Programm eigentlich ist

Bevor ein Kind am Bildschirm Bausteine zusammensetzt, lohnt ein Blick auf die Grundidee. Was ist ein Programm, was ist ein Algorithmus, und warum muss man einem Computer alles so genau sagen? Dieses Kapitel erklärt die wichtigsten Begriffe so, dass du sie einem Kind erklären kannst.

Anweisungen in der richtigen Reihenfolge

Ein **Programm** ist eine Folge von Anweisungen, die ein Computer ausführt. Ein **Algorithmus** ist die Beschreibung eines Lösungswegs, Schritt für Schritt. Jedes Rezept ist ein Algorithmus: erst die Butter schmelzen, dann den Zucker unterrühren, dann die Eier dazugeben. Wer die Reihenfolge vertauscht, bekommt einen anderen Kuchen oder gar keinen.

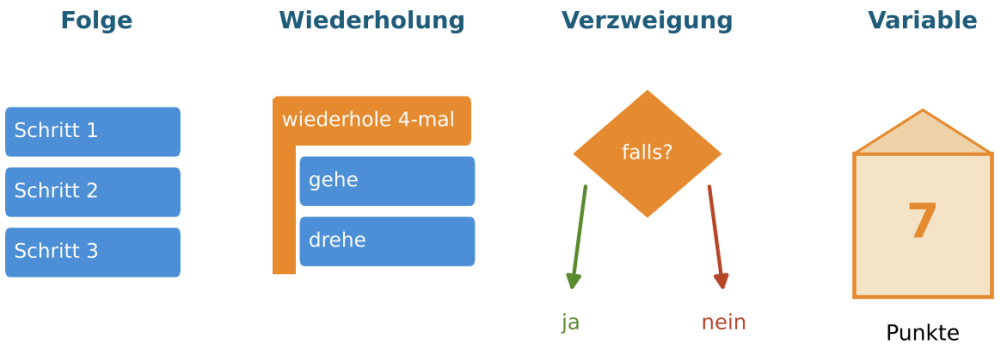
Ein Computer ist dabei ein besonders gehorsamer, aber auch besonders begriffsstutziger Helfer. Er tut genau das, was in den Anweisungen steht, nicht mehr und nicht weniger. Er weiß nicht, was gemeint war. Wenn im Rezept steht »Eier dazugeben«, gibt er vielleicht die Eier samt Schale hinein, weil niemand gesagt hat, dass man sie aufschlagen muss.

Vier Grundbausteine

Fast alle Programme bestehen aus wenigen Grundbausteinen, die sich immer wieder kombinieren:

- **Folge:** Anweisungen werden nacheinander ausgeführt.
- **Wiederholung:** Eine oder mehrere Anweisungen werden mehrfach ausgeführt. Man spricht von einer **Schleife**.
- **Verzweigung:** Je nach Bedingung passiert das eine oder das andere. Man spricht von einer **Bedingung** oder einer Wenn-dann-Anweisung.
- **Variable:** Ein benannter Speicherplatz, in dem sich das Programm etwas merkt, etwa den Punktestand.

Dazu kommen **Ereignisse**, die ein Programm starten oder unterbrechen, etwa ein Mausklick oder eine gedrückte Taste.



Die vier Grundbausteine: Folge, Wiederholung, Verzweigung und Variable. Fast jedes Programm lässt sich daraus zusammensetzen.

Ein Beispiel aus dem Alltag

Das Zähneputzen lässt sich als Programm beschreiben:

```
Zahnbürste nehmen
Zahnpasta auftragen
wiederhole für jede Seite
    putze 30 Sekunden
wenn noch Zahnpasta im Mund ist
    ausspülen
Zahnbürste ausspülen
Zahnbürste in den Becher stellen
```

Hier stecken Folge, Wiederholung und Bedingung. Kinder erkennen solche Muster schnell, wenn man sie darauf aufmerksam macht.

Programmiersprachen

Ein Computer versteht im Innersten nur Nullen und Einsen. Damit Menschen ihm Anweisungen geben können, gibt es **Programmiersprachen**. Es gibt Hunderte davon, etwa Python, JavaScript, Java oder C. Sie unterscheiden sich in ihren Regeln, der **Syntax**, aber die Grundbausteine sind überall gleich.

Für Kinder gibt es **blockbasierte Sprachen**. Statt Befehle zu tippen, ziehen Kinder farbige Bausteine aus einer Liste und stecken sie zusammen. Bausteine, die nicht zusammenpassen, rasten nicht ein. So entstehen keine Tippfehler, und das Kind kann sich ganz auf die Idee konzentrieren.

Art	Beispiel	Vorteil	Nachteil
Bausteine	ScratchJr, Scratch, MakeCode	keine Tippfehler, sehr anschaulich	bei großen Programmen unübersichtlich

Art	Beispiel	Vorteil	Nachteil
Text	Python, JavaScript	mächtig, im Beruf verbreitet	Tippfehler, mehr Regeln
Mischform	MakeCode mit Wechsel zu Python	sanfter Übergang	zwei Ansichten

TIPP **Begriffe nicht erzwingen**

Kinder müssen die Fachbegriffe nicht sofort kennen. Viel wichtiger ist, dass sie die Idee verstehen. Ein Kind, das sagt »Ich will, dass das zehnmal passiert«, hat die Schleife verstanden, auch wenn es das Wort noch nicht benutzt. Die Begriffe kommen von allein, wenn sie im Alltag immer wieder fallen.

VERTIEFUNG **Warum der Computer so genau sein muss**

Menschen verstehen auch ungenaue Anweisungen, weil sie den Zusammenhang kennen und mitdenken. Ein Computer hat diesen Zusammenhang nicht. Er kann nur ausführen, was eindeutig festgelegt ist. Deshalb muss eine Programmiersprache so gebaut sein, dass jede Anweisung genau eine Bedeutung hat. Das ist der Grund, warum ein fehlendes Komma oder ein vertauschter Buchstabe in einer Textsprache ein Programm zum Absturz bringen kann. Blockbasierte Sprachen nehmen Kindern diese Last ab, weil die Form der Bausteine nur sinnvolle Kombinationen erlaubt.

LERNPROJEKT **Das Brotprogramm**

Harald kommt zum Frühstück und bringt eine Idee mit. Er legt eine Scheibe Brot, ein Messer und ein Glas Nutella auf den Tisch und sagt zu Emil: »Du bist der Programmierer. Ich bin der Computer. Sag mir, wie ich ein Nutellabrot mache.« Emil sagt: »Mach Nutella aufs Brot.« Harald nimmt das geschlossene Glas und legt es auf das Brot. Lotta kreischt vor Lachen. Emil versucht es noch einmal: »Glas aufmachen.« Harald dreht am Deckel, aber in die falsche Richtung. Nach zehn Minuten und vielen Lachern steht ein Programm aus vierzehn Schritten auf einem Zettel. Am Ende hat Harald ein Nutellabrot gemacht, und Emil weiß, warum Computer so genaue Anweisungen brauchen. Jana schreibt sich das Spiel für ihre Station auf. Die Auszubildenden könnten es auch gebrauchen.

Algorithmen im Alltag entdecken

Algorithmen begegnen uns ständig, ohne dass wir sie so nennen. Ein paar Beispiele, die Kinder kennen:

- **Kochrezepte:** Zutaten in einer Reihenfolge, Wiederholungen beim Rühren, Bedingungen wie »wenn der Teig zu fest ist, Milch dazugeben«.
- **Spielregeln:** »Wer eine Sechs würfelt, darf noch einmal.« Eine Bedingung mit Wiederholung.
- **Bastelanleitungen:** Schritt für Schritt mit Bildern.
- **Wegbeschreibungen:** »Geh bis zur Ampel, dann links, dann die zweite Straße rechts.«
- **Morgenroutine:** Aufstehen, anziehen, frühstücken, Zähne putzen, Ranzen packen.

Ein schönes Spiel ist, im Alltag gemeinsam Algorithmen zu suchen. Wer findet die meisten? Wer findet eine Schleife? Wer findet eine Bedingung? Kinder werden dabei schnell erfinderisch und entdecken Algorithmen in Ampeln, Waschmaschinen und dem Morgenkreis im Kindergarten.

Mehrdeutigkeit erkennen

Ein wichtiger Gedanke beim Programmieren ist, dass Anweisungen eindeutig sein müssen. Im Alltag ist vieles mehrdeutig: »Bring mal die Sachen rein« kann vieles bedeuten. Kinder haben Spaß daran, solche Mehrdeutigkeiten zu finden. Was genau heißt »ein bisschen Salz«? Was heißt »gleich«? Wann ist »bald«? Diese Übung schärft den Blick für die Genauigkeit, die ein Computer braucht, und nebenbei für die Sprache.

Daten und Programme

Neben Anweisungen besteht jedes Programm aus **Daten**: Zahlen, Texte, Bilder, Töne. Ein Spiel speichert Punkte, Namen und Positionen. Eine App speichert Fotos und Nachrichten. Programme verarbeiten Daten, indem sie sie lesen, verändern und ausgeben. Kinder verstehen diesen Zusammenhang gut am Beispiel eines Rezepts: Die Zutaten sind die Daten, die Arbeitsschritte das Programm, der fertige Kuchen das Ergebnis.

Dieser Gedanke hilft später, wenn Kinder über Datenschutz nachdenken: Welche Daten sammelt eine App über mich? Was macht sie damit? Wer Programme versteht, stellt solche Fragen ganz selbstverständlich.

Ein Programm Zeile für Zeile lesen

Bevor Kinder selbst programmieren, lohnt es sich, ein fremdes Programm laut vorzulesen und vorherzusagen, was passiert. Dieses **Vorhersagen** ist eine bewährte Methode im Informatikunterricht. Ein Beispiel:

```
setze Zahl auf 1
wiederhole 4-mal
    sage Zahl für 1 Sekunden
    setze Zahl auf Zahl mal 2
```

Was sagt die Figur? Kinder überlegen, schreiben ihre Vermutung auf und probieren es dann aus. Die Figur sagt 1, 2, 4 und 8. Wer richtig getippt hat, freut sich. Wer falsch lag, lernt, warum. Diese Übung schult das Verständnis dafür, wie ein Computer Schritt für Schritt vorgeht, und sie macht Spaß, weil sie ein kleines Rätsel ist.

Für Erwachsene ist sie ebenfalls lehrreich. Wer ein paar solcher Rätsel mit dem Kind löst, versteht schnell, wie Programme denken, und kann später besser mitreden.

Werkstatt: Ein Alltagsprogramm schreiben

1. Wähle mit dem Kind eine alltägliche Tätigkeit: Schuhe binden, Tisch decken, Pausenbrot schmieren.
2. Das Kind schreibt oder diktiert die Schritte so genau wie möglich.
3. Du spielst den Computer und führst die Schritte wörtlich aus, auch wenn es komisch wird.
4. Das Kind verbessert das Programm, wo du gescheitert bist.
5. Sucht gemeinsam nach Wiederholungen und Bedingungen in eurem Programm.
6. Tauscht die Rollen und lasst das Kind den Computer spielen.

Häufige Fragen

Was ist der Unterschied zwischen Programm und App? Eine App ist ein Programm für Smartphone oder Tablet. Jede App ist ein Programm, aber nicht jedes Programm ist eine App.

Was ist Code? Der Text oder die Bausteine, aus denen ein Programm besteht. Programmieren nennt man deshalb auch Coden.

Welche Programmiersprache ist die beste? Für Kinder die, mit der sie Freude haben. Für den Einstieg sind Bausteine ideal, später ist Python eine gute erste Textsprache.