

Transkript: „Coding als Sprache der Zukunft“

Einleitung

Müssen wir eine Programmiersprache beherrschen, um die digitale Welt mitgestalten zu können? Und was ist eigentlich ein Algorithmus? Wo begegnet uns Coding im Alltag? Das und noch viel mehr schauen wir uns jetzt mal genauer an.

Eintauchen

Die Digitalisierung unserer Gesellschaft schreitet zweifellos zügig voran und Codes durchdringen mittlerweile vielerlei Lebensbereiche und sie prägen sie maßgeblich. Im Alltag machen wir selbstverständlich Gebrauch von Dingen, die erst durch Programmieren möglich wurden. Sei es das Buchen einer Zugfahrkarte, das Schicken von Nachrichten an Freund*innen, das Fotografieren per Smartphone. Die Liste ist endlos.

Eine Welt ohne Code ist heutzutage undenkbar geworden. Für diejenigen, die auch in Zukunft verantwortungsbewusst und aktiv an dieser Welt teilhaben möchten, ist ein Verständnis für die Zusammenhänge und Hintergründe von großer Bedeutung. Aus diesem Grund wird von Vertreter*innen aus Wirtschaft und Politik zunehmend gefordert, dass Programmieren in Schulen als eine Art zweite Fremdsprache eingeführt werden sollte.

Verstehen

Coding heißt so viel wie Programmieren. Um noch ein bisschen besser zu verstehen, warum Coding so wichtig ist, möchte ich mit Anna Seidel von DIGITAL@School sprechen, einer ehrenamtlichen Initiative von Mitarbeitenden der Deutschen Telekom AG. Sie engagiert sich für die digitale Bildung sowie der Förderung von MINT-Fächern bei Kindern und zeigt, dass Programmieren keine Fremdsprache ist und durchaus Spaß machen kann.

Interview

Teachtoday: Hallo Anna, schön, dass du heute Zeit für uns hast.

Anna Seidel: Hallo, danke für die Einladung. Ich freue mich, dabei zu sein.

Teachtoday: Sag mal, was ist denn eigentlich DIGITAL@School und was macht ihr genau?

Anna Seidel: DIGITAL@School ist eine Community in erster Linie, wo die Mitarbeitenden der ganze Telekom sich ehrenamtlich engagieren und Kindern, Jugendlichen, aber auch Eltern, Erwachsenen alles rund Digitalisierung beibringen. Und zwar auf Augenhöhe. Und da, wo die Hilfe gebraucht wird, sollte man Fragen haben zum wie fange ich mit Coding an oder wie ist das mit dem ersten Handy für mein Kind? Ist alles beantwortet die Community oder können die Ehrenamtlichen beantworten in komplett eigener Initiative und komplett selbstorganisiert.



Teachtoday: Okay, und nimm uns mal mit, warum sind denn grundlegende Programmierkenntnisse in der heutigen Gesellschaft überhaupt so wichtig?

Anna Seidel: Programmieren liegt im Grunde fast überall im Alltag, auch uns sozusagen zu Füßen. Wenn wir an Staubroboter denken, aber auch wenn wir an Zähneputzen denken. Das alles hat mit Programmieren oder mit logischen Abläufen zu tun. Und das ist in den immer weiter entwickelnden digitalisierten Welt mega, mega wichtig, diese Grundlagen zu verstehen. Warum ist es so wie es ist und nicht anders? Und ob das selbstständig ist oder eben nicht? Und wer hinterher agiert und warum Google oder andere Anbieter solche Sachen machen, wie sie machen. Ich rede jetzt sehr kryptisch. Ich mache das ganz konkret.

Teachtoday: Ja, gerne.

Anna Seidel: Wenn wir in die Schulen gehen, ich nehme gerne als Beispiel eine Grundschule, ist es selbstverständlich, dass in der Grundschule die Basics zum sicheren Straßenverkehr beigebracht werden. Und aus unserer Sicht ist es genauso wichtig, diese Basics zum sicheren Umgang mit digitalen Medien mindestens genauso beigebracht werden, dass entweder jemand/ jemand von der Schule sich damit beschäftigt oder jemand von extern damit beschäftigt oder nach Hilfe gefragt wird, wie ein Verkehrslotter an der Straße steht immer früh vor dem Schulbeginn. So sind wir von der Community unterwegs und bieten genau diese Hilfe. Wir sind keine Polizisten, aber wir geben den Kindern immer wieder den Hinweis Schau mal, das ist der Zebrastreifen und liebe Autos, haltet mal an, da laufen jetzt die Kinder. Und das ist unser Wunsch und unser Ziel, dass die Kinder und auch Erwachsene bewusst mit den digitalen Medien umgehen.

Teachtoday: Du hast es gerade schon so ein bisschen angerissen. Welche Vorteile siehst du denn darin, das Programmieren jetzt zu erlernen, selbst wenn man vielleicht nicht unbedingt eine Karriere in der Softwareentwicklung anstrebt? Also welche Kompetenzen sind da wichtig?

Anna Seidel: Das sind ganz, ganz grundlegende Kompetenzen. Ich werde auch nicht zum Doktor der Mathe, wenn ich Mathematik Grundlagen lerne automatisch. Also ich muss nicht ein Doktor Physik werden, um die grundlegenden physischen Gesetze zu verstehen. Und genauso sollte es auch mit Programmieren sein. Es fördert Kreativität, es fördert logisches Denken und macht das Leben einfacher, wenn man an Automatisierungsprozesse denkt. Das kann sehr, sehr viel Arbeit abnehmen, wenn man Grundsätze von Programmieren kennenlernt, so wie Kopfrechnen. Man nimmt ja auch nicht immer Taschenrechner, weil man dann für einkaufen geht, sondern Kopfrechnen, grundsätzliches Verstehen hilft uns einfach im Alltag. Und mittlerweile, also in der jetzigen Zeit, gehört Programmieren nun mal wie Rechnen, Lesen, Schreiben dazu als Grundkompetenz.

Teachtoday: Du hast es gerade schon gesagt. Also uns begegnet Coding im Alltag ganz oft. Kannst du uns vielleicht noch ein Beispiel nennen? Wo finden wir denn Programmieren noch in so ganz alltäglichen Dingen? Also wie kann man das vielleicht vergleichen oder so?

Anna Seidel: Das ist eine schöne Frage, denn gleiche Frage stellen wir den kleinen Kindern auch. Wo findest du denn solche Abläufe in deinem Alltag? Und die Kinder kommen dann auch selber auf die Idee „Ach so, das ist ja etwas, was sich wiederholt“. Und dieses Wiederholen der Abläufe wie Ich stehe jeden Tag früh auf und gehe ins Bad und putze meine Zähne. Dann gibt es Frühstück und dann gehe ich zur Schule. Und das sind im Grunde genommen die Abläufe, die schon rein programmiert sind in unser Gehirn, denn wir denken darüber gar nicht nach. Wir machen das einfach, ganz automatisiert. Und die Kinder sprudeln von den Beispielen aus dem Alltag, wo sie diese automatische Abläufe, wenn einmal die Regel klar ist, denen dann folgen. Oft ist das im Schulalltag sehr viel Wiedererkennung.

Teachtoday: Das heißt, ihr arbeitet sehr auf spielerische Art und Weise, dann auch mit den Kindern, so ein grundlegendes Verständnis erst mal zu bekommen. Super. Jetzt ist ja künstliche Intelligenz gerade in aller



Munde. Wie ist denn deine Meinung? Wird die KI das Programmieren ablösen? Und braucht man denn im Zeitalter von KI überhaupt noch Programmierer*innen?

Anna Seidel: Unbedingt. Unbedingt. KI verändert mit Sicherheit die ganze Welt des Programmierens und der Entwickler, aber löst die auf gar keinen Fall auf. Warum? Weil die KI ist nicht kreativ. Die menschlichen Aspekte und vor allem die Kreativität ist etwas, was die KI nicht ersetzen kann. Und das ist, wenn man das versteht, wie die KI gebaut ist, dass die KI immer in der Vergangenheit aufbaut mit Datensätzen, die damit gefüttert wird, das ist immer die Vergangenheit, aber nie die Zukunft. Und die Zukunft ist nur an Menschen gebunden. Die KI wird mit Sicherheit das revolutionieren und effizienter gestalten. Das heißt, die Wiederholungsvorgänge auflösen und vieles automatisieren, was oft mit Menschen mit menschlicher Arbeit sehr viele Stunden kostet. Das wird mit Sicherheit anders sein in der Zukunft, aber auch jetzt schon. Aber das wird nie ersetzen. Aber auch Ethik und Verantwortung übernehmen wird die KI nicht können und sollte auch nicht machen, sondern das liegt an den Menschen, welche Daten verarbeitet werden dürfen und welche nicht. Wie ist die Qualität der Software, des Produkts, die mit Programmieren entwickelt worden ist, wird auch nicht die KI abnehmen können, sondern nur die Menschen. Ist es das, was erwartet ist? Hat es das Problem gelöst, was gestellt ist? Und das Problem Lösung und Problemstellung liegt auch an Menschen. Das kann KI nicht.

Teachtoday: Für Einsteiger*innen kann das Programmieren jetzt ja oft relativ einschüchternd wirken, sage ich mal. So viele Zahlen und Zeichen. Welchen Tipp hast du da für die Person, das Ganze mal auszuprobieren, selbst wenn sie noch unsicher sind, wie sie vorgehen sollten?

Anna Seidel: Ich denke, dass es mit den Programmiersprachen, die für Einsteiger sehr gut geeignet sind, die auch frei im Web verfügbar sind, die Einsteiger-Kurse auf diversen Lernplattformen, die sind kostenlos und da einfach ausprobieren und nachmachen, ist eine sehr gute Methode, die wir oft bei ganz kleinen Kindern anwenden, bevor sie diese Grundbefehle der Programmierung kennenlernen, ist es wie in Mathematik. Man legt auch Stäbchen zu Stäbchen zusammen und versteht, okay, da sind jetzt zwei Stäbchen. So ist es in Programmiersprache. Man kann die in Baukastensystemen wie Legosteinen zusammenbauen, zusammenstecken und dann entsteht ein Haus. Und genau so fängt man an, oft nicht mit Text basierten Programmiersprachen, sondern mit sehr viel Farben, sehr viel Bildern und mit sehr viel Haptik auch arbeiten wir gerne. Und da lernen auch manche Lehrkräfte, dass ja, man muss eben nicht direkt zum Piloten werden, wenn man im Straßenverkehr teilnehmen möchte. Es reicht auch, wenn ich zu Fuß oder mit Fahrrad fahren kann, aber die Verkehrsregeln kenne.

Teachtoday: Das heißt, man steigert sich nach und nach und fängt erst mal klein an und wie du auch schon meinstest, auch eher spielerisch mit vielen Bildern, vielen Formen und wagt sich dann immer weiter vor. Gut, ja, damit verabschiede ich mich auch schon bei dir, Anna. Vielen, vielen Dank für die vielen spannenden Impulse, die du uns gegeben hast und alles Gute für dich.

Anna Seidel: Danke schön.

Teachtoday: Ich nehme heute mit, dass Coding in unserem alltäglichen Leben allgegenwärtig ist und auch in nahezu allen Lebensbereichen eine immer größere Rolle spielen wird. Programmieren ist nicht nur für angehende Softwareentwickler*innen relevant, sondern fördert auch eine breite Palette von Kompetenzen, darunter Abstraktes und kritisches Denken, Teamarbeit und die kreative Problemlösung. Eine ganz wichtige Botschaft ist daher – Anna hat es auch eben gesagt, haben Sie keine Angst vorm Coding. Einfach mal ausprobieren und in kleinen Schritten starten. Sie werden sehen, wie viel Spaß Programmieren machen kann.



Entdecken

Coding kommt nahezu überall zum Einsatz. Aber können wir alle Programmieren lernen? Ja - das behaupten zumindest Prominente wie Bill Gates oder Mark Zuckerberg. Die Initiative „Hour of Code“ macht genau das möglich. Vorgegeben werden lediglich eine Aufgabe und Programmblöcke. Das Ganze ist sehr niedrigschwellig und es können wirklich alle einsteigen. So sollen vor allem Vorurteile und Ängste abgebaut werden.

Es lohnt sich auch ein Blick auf die Seite „Code.org“, einer gemeinnützigen Organisation, die sich dafür einsetzt, dass Schüler*innen an Schulen die Möglichkeit haben, Informatik als Teil ihrer Grundbildung zu erlernen. Die Organisation arbeitet daran, den Zugang zur Informatik an Schulen zu erweitern und besonders Mädchen und Schüler*innen aus unterrepräsentierten Gruppen stärker einzubeziehen. Auch wenn Programmierer*innen gute Chancen auf dem Arbeitsmarkt haben, nicht alle müssen programmieren können. Viel wichtiger als das Programmieren selbst ist die Fähigkeit, digitale Prozesse und Algorithmen zu verstehen und sich in die digitale Welt einbringen zu können.

Handeln

Beim Programmieren verfassen wir Algorithmen. Diese dienen ähnlich wie Schritt für Schritt Anleitung oder Handlungspläne dazu, uns bei der Lösung von Problemen oder der Bewältigung von Aufgaben zu unterstützen. Das Konzept des Programmierens hat Ähnlichkeiten zu vielen alltäglichen Tätigkeiten, wie zum Beispiel dem Backen. Beim Backen nutzen wir ebenfalls einen Plan, das Rezept, festzulegen, in welcher Abfolge bestimmte Schritte durchgeführt werden müssen.

Lassen Sie uns nun einmal das Konzept des Programmierens genauer betrachten, indem wir ein einfaches Waffelrezept in einer speziellen Sprache darstellen. Das nennen wir kodieren. Stellen Sie sich vor, ein Waffelrezept ist vergleichbar mit einer speziellen Sprache für uns und unser Waffeleisen. Diese Sprache enthält besondere Anweisungen. Genau wie Menschen miteinander sprechen, kommunizieren wir durch den Code mit dem Waffeleisen. Wenn wir das Rezept lesen und verstehen, sind wir in der Lage, köstliche Waffeln zu zubereiten.

Lassen Sie uns die ähnlichen Bausteine von Rezepten und Code mal zusammen anschauen. Da haben wir zum einen die Befehle. Ein Rezept besteht aus vielen kleinen Anweisungen, den sogenannten Befehlen. Genauso gibt es im Code Befehle, die dem Computer mitteilen, was er ausführen soll. Wenn wir jetzt hier mal auf unser Rezept schauen, dann sind das hier oben die Angaben, die ich benötige, die Waffeln herzustellen.

Dann haben wir die Variablen. Im Rezept verwenden wir beispielsweise die Angabe 250 Gramm Mehl. Dabei stellen 250 Gramm eine Variable dar. Sie kann verschiedene Werte haben, abhängig davon, wie viele Waffeln sie zubereiten möchten. In ähnlicher Weise verwenden wir im Code Variablen, unterschiedliche Werte zu speichern. Und auch hier wieder im Rezept sehen Sie hier die Mengenangaben zu den verschiedenen Zutaten. Daten, also die Variablen im Rezept.

Dann haben wir als dritten Punkt die Funktion. Betrachten Sie eine Funktion als einen Schritt in einem Rezept. Wenn Sie Teig mischen, entspricht das einem Schritt im Rezept. Ebenso gibt es im Programmieren Funktionen, die festlegen, was der Computer tun soll. Ich zeig Ihnen das hier auch noch einmal in unserem Rezept. Da haben wir hier die erste Funktion, das Mischen des Teiges und die zweite Funktion, das Backen im Waffeleisen.



Dann haben wir als letzten Punkt die Schleifen und Bedingung. Manchmal rühren wir den Teig so lange, bis er gleichmäßig ist. Das entspricht dann einer Schleife. Eine Schleife ist also das Wiederholen eines Vorgangs so lange, bis ein vorgeschriebener Zustand oder eine Bedingung eintritt. Wie zum Beispiel hier im Rezept. Wir sagen „Wiederhole, „Drehe Teigrührer in Schüssel so lange, bis der Teig cremig ist. Das ist dann eine Bedingung. Das finden Sie hier. Im Code nutzen wir vergleichbare Schleifen und Bedingungen, den Computer zu steuern.

Wenn wir diese Konzepte vergleichen, wird deutlich, dass Programmieren und alltägliche Tätigkeiten viele Gemeinsamkeiten aufweisen. Durch das Verständnis dieser Gemeinsamkeiten können wir sowohl Code als auch Rezepte besser verstehen und anwenden. Probieren Sie doch einmal ein Waffelrezept aus. Sie werden feststellen, wie ähnlich die Herangehensweise ist.

Reflektieren

Halten wir nun zum Schluss noch einmal fest:

1. Coding und Programmieren sind schon längst Teil unseres Alltags. Technische Geräte, aber auch viele Berufsfelder nutzen Programme, um Produkte zu entwerfen oder Arbeitsschritte zu organisieren.
2. Coding fördert analytisches Denken und die Fähigkeit, komplexe Probleme in kleinere, lösbare Teile zu zerlegen. Diese Denkweise kann nicht nur in der Programmierung, sondern auch im täglichen Leben und bei der Bewältigung von Herausforderungen hilfreich sein.
3. Alle können Programmieren lernen. Einfache Programme zu schreiben ist leichter als viele denken.

In vielen Berufsfeldern sind Fähigkeiten wie Coding notwendig. Jedoch ist es genauso wichtig, auch kreativ zu denken und digitale Prozesse und Algorithmen zu verstehen. Um an einer digitalisierten Welt kompetent teilhaben zu können, braucht es einen Blick dafür, wie digitale Technologien funktionieren. Das Erkunden, Ausprobieren und Verstehen von Programmier-Codes ist ein erster wichtiger Schritt in diese Richtung.

Abmoderation

Hier endet nun Ihre Reise durch die Welt des Codings. In diesem Modul haben Sie vieles über Coding erfahren, von Programmiersprachen über Anwendungsbereiche, hin zu Zukunftsaussichten. Seien Sie also neugierig und probieren Sie es unbedingt einmal selbst aus. Tschüss und bis zum nächsten Mal.

