

Melanie Kovacs ist überzeugt, dass «Code Literacy» nicht nur zur Allgemeinbildung gehört, sondern Mitarbeiter und Firmen nachhaltig erfolgreicher macht. Mit Master21 hat die Zürcherin das erste Coding Bootcamp der Schweiz lanciert. Mit dem Ziel, Business-Leuten ein besseres Verständnis von «Tech» zu vermitteln.

PROGRAMMIEREN LERNEN – JETZT!

Text MELANIE KOVACS Illustration FABIAN WIDMER

Ich will mir in einem Media Markt eine neue Kaffeemaschine kaufen. Auf Google Maps suche ich die nächste Filiale. Ich gehe zu meinem Auto, öffne mit dem Funkschlüssel die Tür, starte das GPS und fahre los. An der roten Ampel warte ich, bis sie grün wird. Im Geschäft angekommen, nehme ich den Lift in den 3. Stock, wo die Kaffeemaschinen stehen. Ich suche mir eine aus, gehe zur Kasse und scanne den Barcode am Self Checkout.

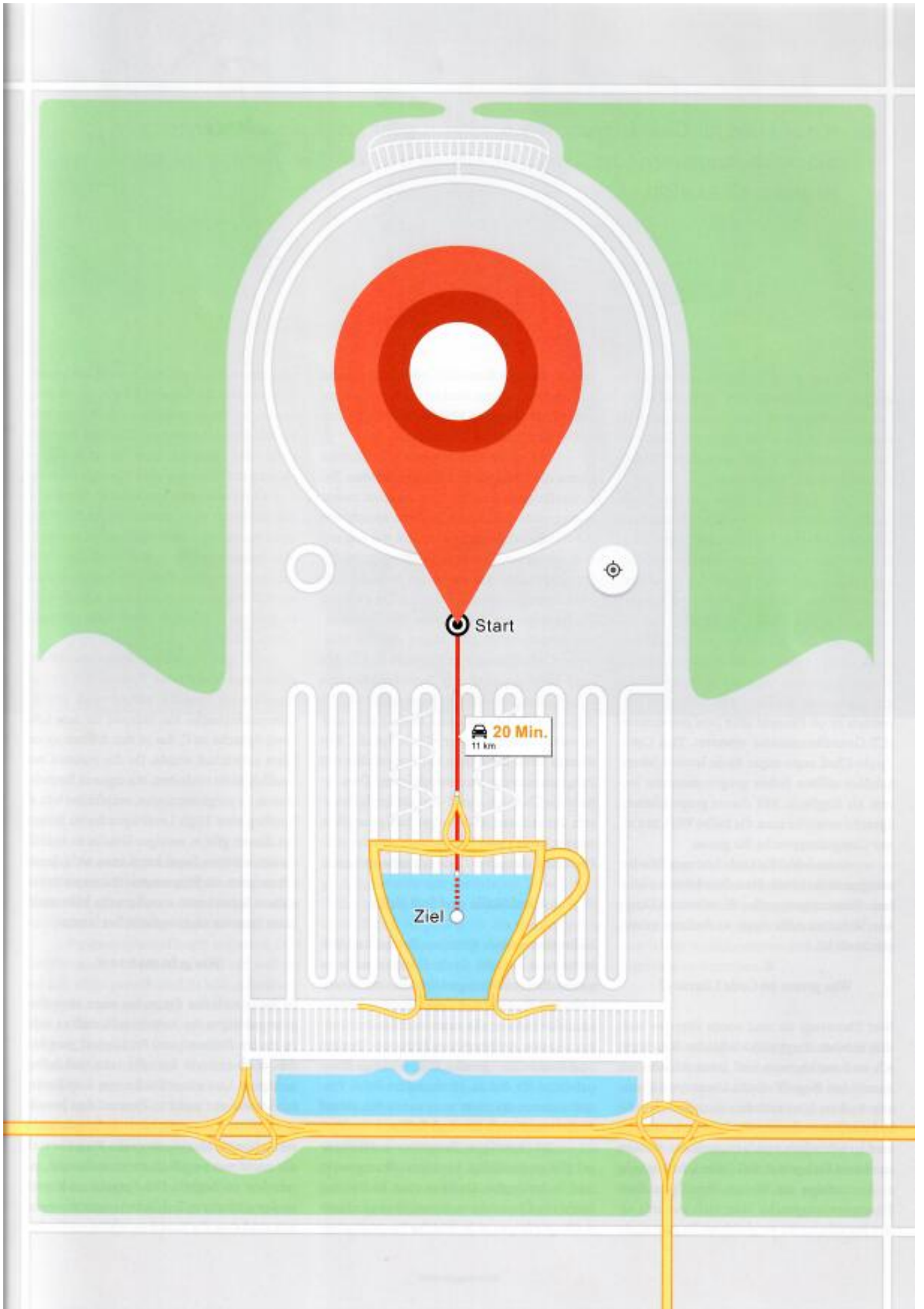
In einer knappen halben Stunde habe ich Google, einen Funkschlüssel, das GPS-System, Ampeln, einen Lift, eine Kaffeemaschine und den Self Checkout benutzt. All diese Dinge wurden von Menschen programmiert und funktionieren mit Code.

Wo wir auch sind: Der Code ist auch da. Nicht nur in der IT, sondern auch im Handel, bei der Mobilität und überall sonst. Die meisten Menschen haben jedoch keine Ahnung, wie die Programme, die sie tagtäglich verwenden, funktionieren. Sie können keinen Code lesen – geschweige denn schreiben.

Grund eins für Code Literacy: Wir sind umgeben von Code. Deshalb macht es Sinn, verstehen zu wollen, wie die Maschinen um uns herum funktionieren, wie sie uns beeinflussen und was möglich ist. Allein mit unserem persönlichen Supercomputer, dem Smartphone, verbringen wir schliesslich sehr viel Zeit. In der Arbeitswelt ist die Digitalisierung in aller Munde. Es reicht jedoch nicht, Prozesse einfach nur zu digitalisieren, wir müssen die Menschen mitnehmen. «Know-how ist der zentrale Erfolgsfaktor in der digi-

talen Transformation», sagt Patrick Warnking, Verwaltungsrat bei der Initiative Digitaliswitzerland und Landesmanager Google Schweiz.

Laut der «Digital IQ Studie 2017» von PWC mit über 2000 befragten Business- und Technologie-Verantwortlichen ist die fehlende Kollaboration zwischen Business und IT häufig der Grund für blockierte Transformationsanstrengungen. Projekt- und Produktleiter verstehen IT-Spezialisten schlicht nicht. Dieses Unverständnis führt nicht nur zu Verzögerungen, sondern kostet auch ziemlich viel Geld. Laut einer Studie der Universität Oxford, die über 1400 IT-Projekte im Gesamtwert von 240 Milliarden Dollar untersuchte, scheitern 40 Prozent von ihnen. Sie kosten nicht nur doppelt so viel wie budgetiert und verschlingen zwei Drittel mal so viel Zeit, sondern sind obendrein komplett nutzlos. —



■ «Grund drei für Code Literacy: Wir benötigen mehr Leute, die selber kreieren können. Konsumieren oder IT-relevante Dinge wie Websites oder Apps in Auftrag zu geben, reicht nicht.»

Grund zwei für Code Literacy: Die digitale Transformation erfordert Zusammenarbeit zwischen Business und IT. Businessleute, die selber etwas von Code verstehen, sind besser gerüstet für die Zusammenarbeit mit IT-Spezialisten.

Wir brauchen nicht nur Manager, die Programmcodes lesen können, sondern auch mehr Fachkräfte aus dem Bereich der Informations- und Kommunikationstechnologie (ICT). Laut einer Studie des Berufsverbands ICT-Berufsbildung fehlen der Schweiz bis 2024 25 000 dieser Fachkräfte, fast die Hälfte im Bereich Softwareentwicklung. Doch nicht nur ICT-Fachkräfte brauchen digitale Fähigkeiten. Gemäss EurActiv-Studie, ein führendes Medium zur Europapolitik, werden in 90 Prozent aller Jobs mindestens ICT-Grundkenntnisse erwartet. Tim Cook, Apple-Chef, sagte sogar Ende letzten Jahres, Schüler sollten lieber programmieren lernen als Englisch. Mit dieser gesprochenen Sprache erreiche man die halbe Welt, mit einer Computersprache die ganze.

Grund drei für Code Literacy: Wir benötigen mehr Leute, die selber kreieren können. Konsumieren oder IT-relevante Dinge wie Websites oder Apps in Auftrag geben, reicht nicht.

Was genau ist Code Literacy?

Wer überzeugt ist und «code literate» werden möchte, fragt sich vielleicht: Was lerne ich und wo beginne ich? Dazu müssen wir zuerst den Begriff «Code Literacy» definieren und in Niveaustufen einteilen. Literacy bedeutet nicht nur die Fähigkeit zu lesen und zu schreiben, sondern auch das Wissen in einem Fachgebiet. Bei Code Literacy geht es demzufolge um Wissen über Code. Eine Programmiersprache lässt sich gut mit einer Fremdsprache vergleichen: Es geht nicht

darum, dass alle flüssend Code sprechen und zu Programmierern werden – sich einigermaßen verständigen zu können, reicht fürs Erste. Das lernt man im Grundkurs.

Bei einer Fremdsprache lernt man zuerst die Grammatik und den Satzbau. Bei einer Programmiersprache kommt zuerst «Computational thinking» dran, sprich: wie ein Computer funktioniert und wie ich mit ihm «spreche», damit er mich versteht. Bei Fremdsprachen unterscheidet man die Ausbildungsstufen von A1 bis C2. Da es beim Programmieren noch keine standardisierten Niveaustufen gibt, teilen wir bei Master21 «Code Literacy» in folgende drei Stufen ein: Level 1 schafft ein Grundverständnis von «Computational thinking» und hilft, die digitale Welt zu verstehen. Level 2 hilft mitzureden – sogenannter «Tech Speak». Mit diesem Level kann man auf Augenhöhe mit Programmierern kommunizieren. Dies erhöht die Chancen, ein IT-Projekt im definiertem Zeitrahmen und Budget fertigzustellen, massiv. Level 3 sind Programmierer oder ICT-Fachleute, die selber Code schreiben können.

Hard Skills und Soft Skills

Nebst verbesserter Kommunikation und Kollaboration verhilft Code Literacy auch zu einem Problemlösungs-Mindset und Kreativität. Die Vorteile von Code Literacy könnte man also mit vier «K» zusammenfassen: Kommunikation, Kollaboration, kritisches Denken und Kreativität – gemäss WEF wichtige Kompetenzen für das 21. Jahrhundert. Beim Programmieren trainiert man nebst den «Hard Skills» somit auch die «Soft Skills».

Im heutigen Programmierschub gibt es unzählige Sprachen, Frameworks und Technologien. Da kann man als Neuling leicht die Übersicht verlieren. Verunsicherte Anfänger fragen sich: Welche Programmier-

sprache ist die beste, welche soll ich lernen? Eine eindeutige Antwort gibt es nicht: Fragt man zehn Programmierer, erhält man zehn verschiedene Antworten. Ob man Portugiesisch oder Spanisch lernt, hängt davon ab, ob man auf Spanisch oder Portugiesisch mit dem Computer sprechen möchte. Denn letztlich übersetzt der Computer alles in Nullen und Einsen, sogenannten Maschinencode. Einige Programmiersprachen wurden geschrieben, um Betriebssysteme zu bauen, andere, um iOS-Apps zu programmieren, und wieder andere, um möglichst schnell Webapplikationen zu erstellen.

Es gibt High-Level-Sprachen wie beispielsweise Ruby oder Python, die der gesprochenen Sprache näher sind als der Computersprache. Ein Beispiel für eine Low-Level-Sprache ist C, das in den frühen 1970er Jahre entwickelt wurde. Da die meisten vermutlich nicht vorhaben, ein eigenes Betriebssystem zu programmieren, empfehlen wir, als Einstieg eine High-Level-Sprache zu lernen. Bei diesen gibt es weniger Details zu spezifizieren, entsprechend kann man sich darauf fokussieren, die Programmierkonzepte zu verstehen. Sobald man eine Sprache beherrscht, kann man die nächste einfachere lernen.

Wie geht man vor?

Es stellt sich die Frage, wo man eine Programmiersprache lernen soll, online oder in einem Präsenzkurs. Prinzipiell geht beides, doch ersteres braucht sehr viel Selbstdisziplin: Laut einer Studie von Katy Jordan schliessen nur zirka 13 Prozent den jeweiligen Online-Kurs tatsächlich ab. Programmieren ist keine Magie, aber eine Welt für sich, die nicht ganz einfach zu verstehen ist, zumindest zu Beginn. Die Sprache zu lernen, ist der einfachere Teil. Schwieriger zu verstehen sind etwa die Zusammenhänge zwischen



Design, Frontend, Backend und Datenbanken. Zeit und Disziplin sind wichtig, um in diese Welt abzutauchen. Eine Kombination von Online und Präsenz kann sinnvoll sein.

Programmieranfänger verlassen ihre Komfortzone und geben schnell auf, weil sie auf sich allein gestellt sind. Es hilft, schrittweise begleitet zu werden, mit anderen im selben Boot zu sitzen und Experten um Rat fragen zu können. Erstmals im Kurs, verändert der Programmierunterricht das Denken: von «Ich verstehe das nicht, ich bin wohl zu dumm dafür» zu «Ich verstehe XY und ich werde auch Z noch lernen».

Am Anfang fragen sich Interessenten, was sie in den paar Tagen etwa eines Coding Bootcamps, wie es Master21 anbietet, lernen können. Neben der Erstellung persönlicher Blogs und Webseiten kann man

beispielsweise lernen, repetitive, langweilige Aufgaben zu automatisieren. So wie eine Teilnehmerin am Programmier-Event Rails Girls Zürich im Jahr 2016. Sie musste jeden Monat manuell unzählige Lohnauszüge herunterladen und umbenennen. An diesem Wochenende erstellte sie gemeinsam mit einem Coach ein Programm, das ihre Aufgabe mit 20 Zeilen Ruby Code automatisierte. Nach der Lösung dieses realen Problems war sie erst recht motiviert, programmieren zu lernen.

Um digitale Produkte und Dienste zu erstellen, reicht es nicht, bloss «code literate» zu sein. Erfolgreiche Apps und Webseiten werden von interdisziplinären Teams aus Strategen, Psychologen, UX-Designern, Frontend- und Backendentwicklern entwickelt. Dafür ist es hilfreich, etwas von User Experience (UX) zu verstehen. Apps und Websites

bieten nur dann einen Mehrwert, wenn sie ein wirkliches Nutzerbedürfnis stillen. Bevor man also einfach drauflos programmiert und die x-te nutzlose App erstellt, lohnt es sich, einen Schritt zurückzutreten und die Nutzerbedürfnisse zu verstehen. ☐

Melanie Kovacs ist Gründerin und Geschäftsführerin von Master21, dem ersten Anbieter von Coding Bootcamps und Code Weeks in der Schweiz. Master21 hat es sich zur Mission gemacht Professionals mit den nötigen Fähigkeiten, Tools und Mindsets auszustatten um erfolgreich in der digitalen Welt zu sein. Kürzlich wurde Melanie von Forbes auf die 30 under 30 Liste gesetzt.

Bildquelle: S.30 Master21