



Hans-Georg
Schumann

2. Auflage



Small Basic

FÜR **KIDS**



Hinweis des Verlages zum Urheberrecht und Digitalen Rechtemanagement (DRM)

Der Verlag räumt Ihnen mit dem Kauf des ebooks das Recht ein, die Inhalte im Rahmen des geltenden Urheberrechts zu nutzen. Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und Einspeicherung und Verarbeitung in elektronischen Systemen.

Der Verlag schützt seine ebooks vor Missbrauch des Urheberrechts durch ein digitales Rechtemanagement. Bei Kauf im Webshop des Verlages werden die ebooks mit einem nicht sichtbaren digitalen Wasserzeichen individuell pro Nutzer signiert.

Bei Kauf in anderen ebook-Webshops erfolgt die Signatur durch die Shopbetreiber. Angaben zu diesem DRM finden Sie auf den Seiten der jeweiligen Anbieter.

Für Julia, Janne, Katrin und Daniel

Hans-Georg Schumann



Small Basic für Kids

2. Auflage



mitp

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-95845-323-4

2. Auflage 2016

www.mitp.de

E-Mail: mitp-verlag@sigloch.de

Telefon: +49 7953 / 7189 - 079

Telefax: +49 7953 / 7189 - 082

© 2016 mitp-Verlags GmbH & Co. KG

Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Lektorat: Katja Völpe

Sprachkorrektorat: Petra Heubach-Erdmann

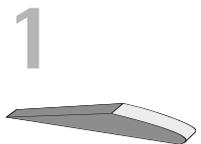
Covergestaltung: Christian Kalkert

Satz: Ill-Satz, Husby, www.drei-satz.de

Inhalt



Vorwort	11
Was heißt denn Programmieren?	11
Was ist eine Entwicklungsumgebung?	12
In welcher Sprache wird programmiert?	12
Aller Anfang ist schwer	14
Einleitung	15
Wie arbeite ich mit diesem Buch?	15
Was brauchst du für dieses Buch?	17
Wie gut kennst du deinen PC?	17
Das erste Programm	19
Small Basic holen und installieren	19
Small Basic starten	23
Tasten suchen und finden	25
Ein Programm eingeben und starten	27
Eine Frage verdient eine Antwort	34
Small Basic beenden	36
Zusammenfassung	37
Ein paar Fragen	38
... aber nur eine Aufgabe	38



2



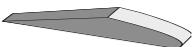
Bedingungen und Kontrollen	39
Ein Programm wieder öffnen	40
Wenn ... dann	41
Die If-Struktur	44
If und Else	46
Und sonst?	48
Ein bisschen Grundrechnen	50
Zuweisung und Ergebnis	52
Zusammenfassung	54
Eine Frage	55
... und zwei Aufgaben	55

3



Vergleichen und wiederholen	57
Zensurenbild	58
Und und Oder, oder?	59
Ein kleines Ratespiel	62
Die While-Struktur	64
Dein PC zählt mit	66
Noch mehr Spielkomfort	69
Goto gibt's auch noch	70
Zusammenfassung	72
Ein paar Fragen	73
... und ein paar Aufgaben	73

4



Geld-Spielereien	75
Spiel mit dem Glück	75
Die For-Struktur	78
Auf dem Weg zum Millionär	80
Wie hoch ist dein Startkapital?	82
Schleifenvielfalt	83
Macht Lotto reich?	85
Textbearbeitung	87
Zusammenfassung	88
Ein paar Fragen	89
... und eine Reihe Aufgaben	89

Grafik mit Draw und Fill	91
Text oder Grafik	92
Linie, Rechteck und Kreis	94
Jetzt wird's bunt	98
Eckig oder rund?	100
Bunte Mischung	102
Text geht auch	104
Zusammenfassung	105
Ein paar Fragen	106
... und zwei Aufgaben	106
Pixel und Turtle	107
Punkte und Felder	107
Von Kreisen und Wellen	111
Farbteppiche	114
Die Sache mit der Schildkröte	117
Zusammenfassung	121
Keine Frage	122
... aber eine Menge Aufgaben	122
Objekte und Ereignisse	125
Shapes	126
Eine kleine Ballerei	130
Bewegung auf Tastendruck	133
Mausklicks	135
Zusammenfassung	137
Ein paar Fragen	139
... und ein paar Aufgaben	139
Tasten- und Maussteuerung	141
Einsatz der Pfeiltasten	141
MouseMove oder MouseClick?	144
Ball oder Käfer?	147
Käfer folgt Maus	150

5



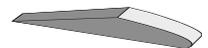
6



7



8



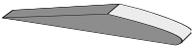
Maus trifft Käfer	153
Zusammenfassung	157
Ein paar Fragen	157
... und ein paar Aufgaben	158

9



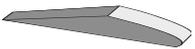
Prozeduren	159
Teilstücke	159
Die Spielschleife	162
Alles zusammen	164
Die Sache mit dem Timer	166
Schlag den Ball	167
Zusammenfassung	172
Ein paar Fragen	173
... und ein paar Aufgaben	173

10



Sound, Clock und Timer	175
Treffer mit Sound	175
Spiel mit Timing	177
Zeit und Datum	181
Jump and Duck	182
Der passende Schluss	188
Zusammenfassung	189
Ein paar Fragen	190
... und ein paar Aufgaben	190

11



Figuren in Bewegung	191
Endlich ein Bild	191
Die Figur in Bewegung	194
Tasten-Wanderung	197
Spielfeld mit und ohne Grenzen	199
Figur-Zoom	202
Hintergründe	204
Zusammenfassung	209
Zwei Fragen	209
... und ein paar Aufgaben	209

Komponenten und Datenfelder 211

- Buttons zum Klicken 212
- Eine Box für den Text 214
- Fragen und Antworten 216
- Richtig oder falsch? 220
- Datentransfer 223
- Zusammenfassung 225
- Ein paar Fragen 226
- ... und ein paar Aufgaben 226

Boom – Noch ein Spiel? 227

- Kanone und Kugel 227
- Der erste Schuss 231
- Mehr Komfort mit Buttons 234
- Auf dem Weg zum Ziel 237
- Alles am Stück 241
- Zusammenfassung 245
- Keine Fragen 246
- ... aber zwei Aufgaben 246

Anhang A 247

- Für Eltern 247
- ... und für Lehrer 248

Anhang B 251

- Kleiner Fehlervermeidungs-Check 251

Anhang C 255

- Der Wortschatz von Small Basic 255

Stichwortverzeichnis 269

12



13



A



B



C





Vorwort

Es sind schon wahre Wunderwerke, die Computer! Man trifft diese Dinge überall, sie können allerhand Sachen, man kann damit Texte schreiben, Bilder malen und spielen.

In Wirklichkeit aber sind sie stohdumm. Denn sie machen nur das, was ihnen ein Programm sagt. Ist das ein Grafik-Programm, dann lassen sich mit dem Computer Bilder bearbeiten. Ist das ein Actionspiel, dann lässt sich der Computer mit einem Joystick oder den Tasten steuern und bewegt irgendwelche Objekte über den Bildschirm.

Programme werden von Leuten erstellt, die sich in einer Programmiersprache auskennen, sich Nächte um die Ohren schlagen und nur in der Nähe ihres Computers schlafen. So sagt man. Aber das muss ja nicht stimmen.

Was heißt denn Programmieren?

Wenn du aufschreibst, was ein Computer tun soll, nennt man das Programmieren. Das Tolle daran ist, dass du selbst bestimmen kannst, was getan werden soll. Lässt du dein Programm laufen, macht der Computer die Dinge, die du ausgeheckt hast. Natürlich wird er dann nicht dein Zimmer aufräumen und dir auch keine Pizza servieren. Aber kannst du erst mal programmieren, kannst du den Computer sozusagen nach deiner Pfeife tanzen lassen.

Allerdings passiert es gerade beim Programmieren, dass der Computer nicht so will, wie du es gerne hättest. Meistens ist das ein Fehler im Programm. Der Fehler kann aber auch irgendwo anders im Computer oder im Betriebssystem liegen. Das Dumme bei Fehlern ist, dass sie sich gern so gut verstecken, dass die Suche danach schon manchen Programmierer zur Verzweiflung gebracht hat.

Vielleicht hast du nun trotzdem Lust bekommen. Dann brauchst du ja nur noch eine passende Entwicklungsumgebung und schon kann's losgehen.

Was ist eine Entwicklungsumgebung?

Um ein Programm zu erstellen, musst du erst mal etwas eintippen. Das ist wie bei einem Brief oder einer Geschichte, die man schreibt. Das Textprogramm kann hier sehr einfach sein, weil es ja nicht auf eine besondere Schrift oder Darstellung ankommen muss. So etwas wird Editor genannt.

Ist der Programmtext eingetippt, kann ihn der Computer nicht einfach lesen und ausführen. Jetzt muss alles so übersetzt werden, dass der PC versteht, was du von ihm willst. Weil er aber eine ganz andere Sprache spricht als du, muss ein Dolmetscher her.

Du programmierst in einer Sprache, die du verstehst, und der Dolmetscher übersetzt es so, dass es dem Computer verständlich wird. So etwas heißt dann Compiler (ausgesprochen: Kompailer).

Schließlich müssen Programme überarbeitet, verbessert, wieder getestet und weiter entwickelt werden. Dazu gibt es noch einige zusätzliche Hilfen. Daraus wird dann ein ganzes System, die Entwicklungsumgebung.

In welcher Sprache wird programmiert?

Leider kannst du nicht so programmieren, wie dir der Schnabel gewachsen ist. Eine Programmiersprache muss so aufgebaut sein, dass möglichst viele Menschen in möglichst vielen Ländern einheitlich damit umgehen können.

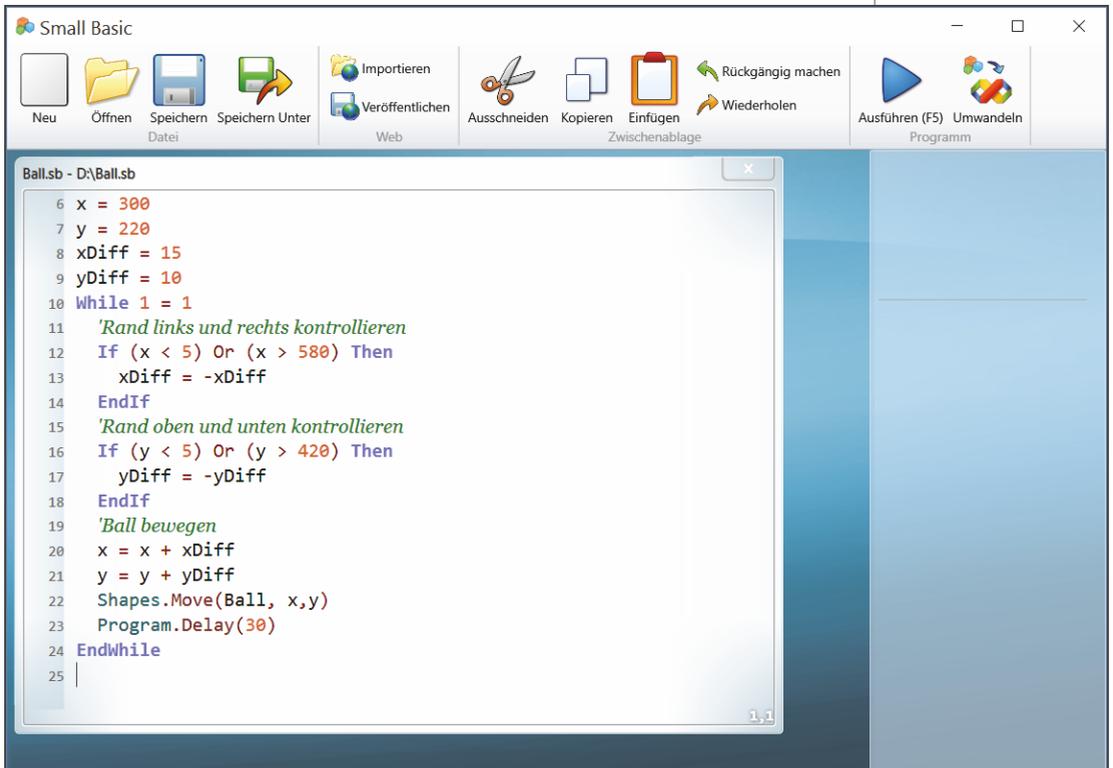
Weil in der ganzen Welt Leute zu finden sind, die wenigstens ein paar Brocken Englisch können, besteht auch fast jede Programmiersprache aus englischen Wörtern. Es gab auch immer mal Versuche, z.B. in

In welcher Sprache wird programmiert?

Deutsch zu programmieren, aber meistens klingen die Wörter dort so künstlich, dass man lieber wieder aufs Englische zurückgreift.

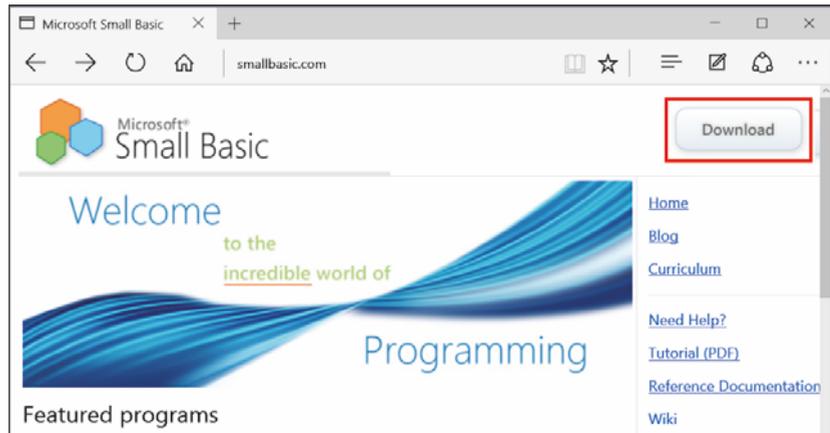
In diesem Buch hast du es mit der Programmiersprache Basic zu tun. Sie ist weit verbreitet und leicht zu erlernen. Daneben gibt es noch einige andere Sprachen wie z.B. Java, C++/C# und Delphi (und auch dazu passende Kids-Bücher).

In den verfügbaren Basic-Systemen habe ich nach einem gesucht, das sich möglichst einfach handhaben lässt. Auch kann es nicht schaden, wenn sich damit zumindest einfache Spiele ohne allzu großen Aufwand programmieren lassen.



Da begegnete mir in Small Basic eine kleine, aber feine Entwicklungsumgebung, die das Wesentliche zu bieten hat, was man so zum Programmieren gebrauchen kann. Und weil sie auch noch Spaß machen kann, lag es nahe, sich mit diesem Basic-Dialekt näher zu beschäftigen.

Dieses Programmiersystem wird kostenlos von Microsoft zur Verfügung gestellt, die neueste Version kannst du dir unter <http://smallbasic.com> herunterladen.



Aller Anfang ist schwer

Ich würde lügen, wenn ich behaupte, dass Programmieren einfach ist. Immerhin musst du eine neue Sprache lernen, die nicht nur aus ein paar englischen Wörtern besteht. So manches wirst du erst nach einiger Übung verstehen. Nicht selten kann es vorkommen, dass du die Lust verlierst, weil einfach gar nichts klappen will. Das Programm tut etwas ganz anderes, du kannst den Fehler nicht finden, und du fragst dich: Wozu soll ich eigentlich programmieren lernen, wo es doch schon genug Programme gibt?

Trotzdem meine ich: Es ist nicht nur einen Versuch wert, es lohnt sich, es immer wieder zu versuchen. Hast du erst mal »Blut geleckt«, dann lässt es dich so schnell nicht wieder los ...



Einleitung

Wie arbeite ich mit diesem Buch?

Grundsätzlich besteht dieses Buch aus viel, viel Text und einer Menge Abbildungen dazwischen. Natürlich habe ich mich bemüht, alles so zuzubereiten, dass daraus lauter gut verdauliche Happen werden. Damit das Ganze noch genießbarer wird, gibt es zusätzlich noch einige Symbole, die ich dir hier gern vorstellen möchte:

Arbeitsschritte

- Wenn du dieses Zeichen siehst, heißt das: Es gibt etwas zu tun. Damit kommen wir beim Programmieren Schritt für Schritt einem neuen Ziel immer näher.

Grundsätzlich lernt man besser, wenn man einen Programmtext selbst eintippt oder ändert. Aber nicht immer hat man große Lust dazu. Weil du alle Programme im Buch auch auf der Verlags-Website findest, ist hinter einem Programmierschritt auch der jeweilige Dateiname (z.B. GRAFIK1.SB) angegeben. Wenn du also den Programmtext nicht abtippen willst, kannst du stattdessen die zugehörige Datei laden. Folge einfach diesem Link:

<http://www.mitp.de/322>

Fragen und Aufgaben

Am Ende eines Kapitels wirst du jeweils einige Fragen und Aufgaben entdecken. Diese Übungen sind nicht immer ganz einfach, aber sie helfen dir, noch besser zu programmieren. Lösungen zu den Fragen und Aufgaben findest du ebenfalls dort auf der Website des Verlages, wo auch die Beispielprogramme liegen. Du kannst sie dir im Editor von Windows oder auch in deinem Textverarbeitungsprogramm anschauen. Oder du lässt sie dir ausdrucken und hast sie dann schwarz auf weiß, um sie neben den PC zu legen.

Wenn eine aktuelle Aufgabe die Änderung eines Projekts betrifft, steht direkt dahinter in Klammern mit einem Pfeil (→) versehen der Name der Datei, in der das geänderte Programmprojekt zu finden ist (→ FIGUR1.SB).

Notfälle



Manchmal hast du eine falsche Taste gedrückt oder etwas vergessen. Oder es wird gerade ganz knifflig. Dann fragst du dich, was du nun tun sollst. Bei diesem Symbol findest du eine Lösungsmöglichkeit. Notfalls kannst du aber auch ganz hinten im Anhang B nachschauen, wo eine Reihe von häufigen Programmierpannen aufgeführt ist.

Wichtige Stellen im Buch



Hin und wieder findest du ein solch dickes Ausrufezeichen im Buch. Dann ist das eine Stelle, an der etwas besonders Wichtiges steht.



Wenn es um eine ausführlichere Erläuterung geht, tritt Buffi in Erscheinung und schnuppert in seiner Kiste mit Tipps & Tricks.

Was brauchst du für dieses Buch?

Betriebssystem

Das Betriebssystem sorgt dafür, dass der Betrieb deines Computers möglichst reibungslos ablaufen kann. Es übernimmt u.a. die Verwaltung des Arbeitsspeichers und der Geräte, die mit dem Computer verbunden sind: von Monitor und Tastatur bis zu Festplatten und anderen Laufwerken.

Fast jeder Computer arbeitet heute mit dem Betriebssystem Windows. Das gibt es in mehreren Versionen. Small Basic funktioniert unter allen Windows-Versionen bis 10.

Speichermedien

Auf jeden Fall ist ein USB-Stick sinnvoll, auch wenn du deine Programme auf der Festplatte unterbringen willst. Denn auf diesem Speicher-Stick sind deine Projekte zusätzlich sicher aufgehoben.

Gegebenenfalls bitte deine Eltern oder Lehrer um Hilfe: Sie sollen den Anhang A lesen. Dann können sie dir bei den ersten Schritten besser helfen.

Wie gut kennst du deinen PC?

Du musst dich mit deinem PC nicht perfekt auskennen, um mit Small Basic zu programmieren. Es ist aber gut zu wissen, wie man Small Basic startet und beendet. Das erfährst du gleich im ersten Kapitel.

Wenn du noch Schwierigkeiten mit dem PC hast, ist es besser, sich erst mal ein grundlegendes Buch darüber anzuschauen, beispielsweise *Computer für Kids*.

Kennst du dich aber schon gut mit dem Computer und mit Windows aus, dann lass uns beginnen!





1

Das erste Programm

Hier geht es gleich ans »Eingemachte«. Nachdem wir Small Basic installiert und gestartet haben, machen wir einen kleinen Spaziergang durch unsere Programmierumgebung, und dabei entsteht dann auch dein erstes Programm.

In diesem Kapitel lernst du

- ⊙ wie man Small Basic installiert
- ⊙ wie man Small Basic startet
- ⊙ wo welche besonderen Tasten zu finden sind
- ⊙ wie man ein Programm eingibt und startet
- ⊙ Methoden für die Eingabe und Ausgabe von Text kennen
- ⊙ wie man Small Basic beendet

Small Basic holen und installieren

Die Installation von Small Basic sollte für dich auch als Anfänger kein Problem sein. Das Programm, das wir dazu benötigen, heißt SMALL-BASIC.MSI.

1



Zuerst musst du diese Datei herunterladen. Das geht am besten direkt über die passende Seite von Microsoft. Aktuell ist diese Adresse:

<http://smallbasic.com>

Hier findest du gleich rechts oben die Schaltfläche für den Download.

Ein anderer Weg führt über den Link

<http://www.microsoft.com/downloads/de-de>

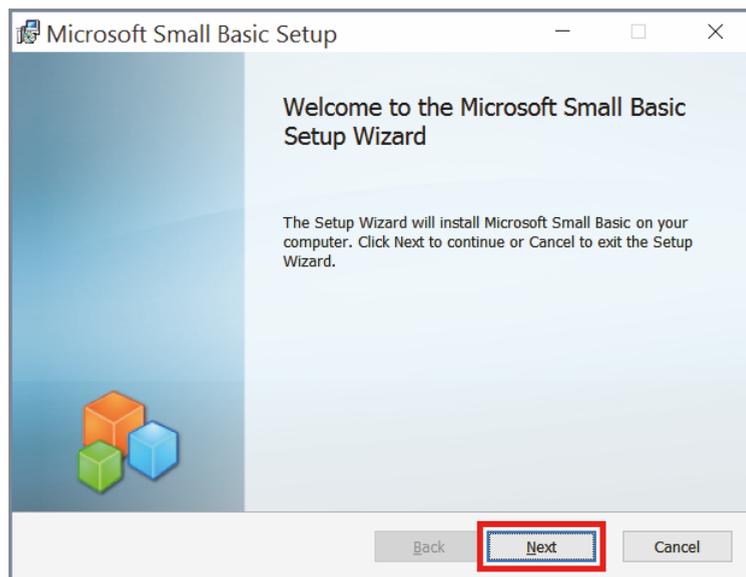
(Hier kannst du als Suchwort »Small Basic« eingeben.)

Ansonsten findest du über eine der Suchmaschinen weitere Stellen mit Möglichkeiten zum Download von Small Basic. (Legst du Wert auf deutsche Menüeinträge, dann achte darauf, dass du die deutsche Version erwischst.)

- Und nun doppelklicke mit der Maus in dem Ordner, in den du das Programm heruntergeladen hast, auf das entsprechende Symbol.

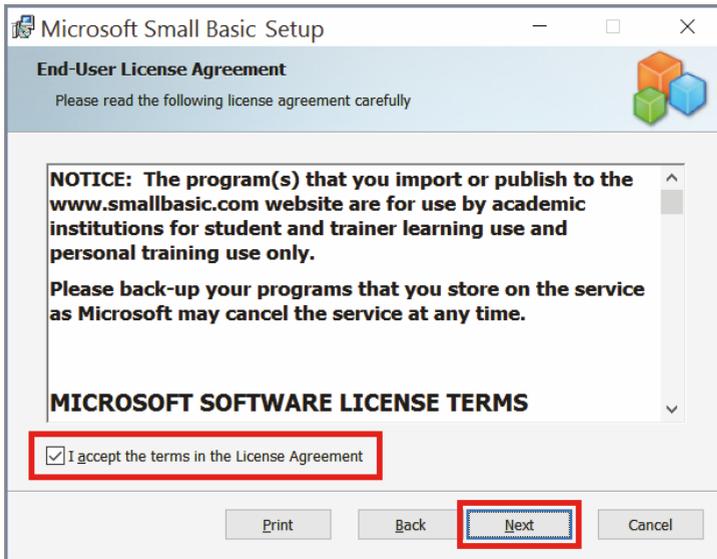


- Einen Moment später begrüßt dich ein Dialogfeld. Dort klickst du einfach auf NEXT.

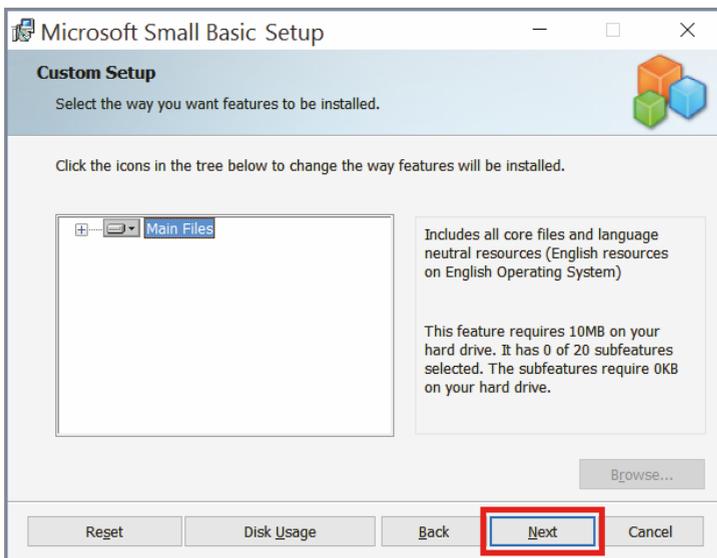


Im nächsten Fenster findest du die Lizenzbedingungen. Wenn du willst, kannst du sie dir durchlesen.

- Auf jeden Fall musst du am Ende zustimmen, also »I accept the terms in the Licence Agreement« anklicken, damit davor ein Häkchen steht. Dann klicke auch hier auf die Schaltfläche NEXT.

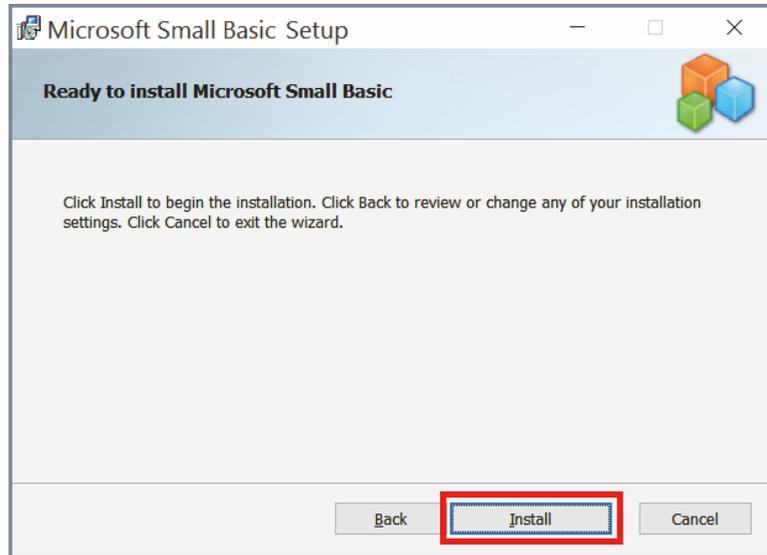


- Auch wenn es nun so aussieht, als gäbe es hier eine Wahl: Klicke einfach nur auf NEXT.

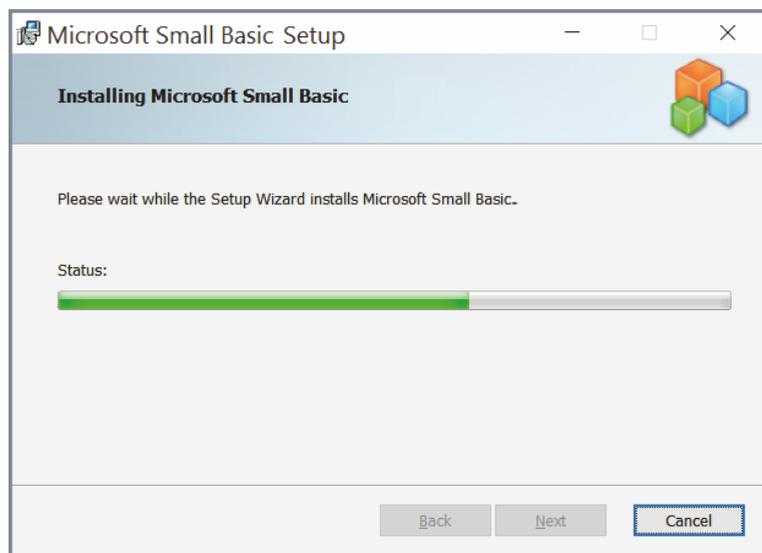


1

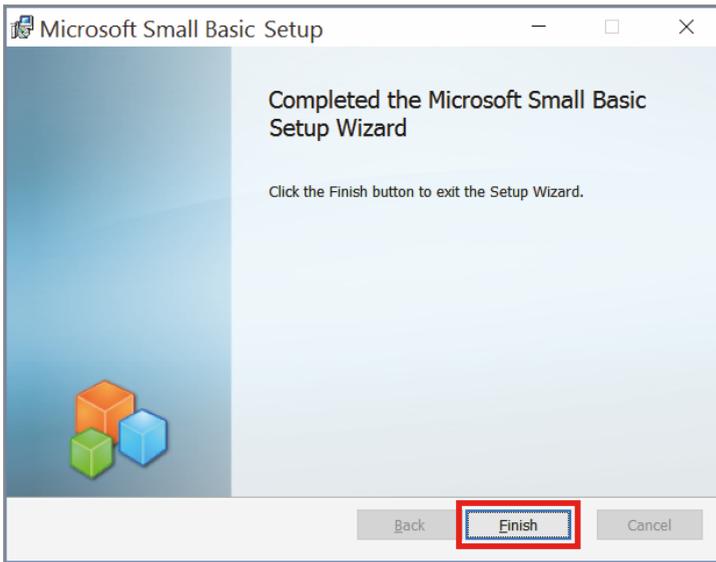
- Und auch im folgenden Fenster geht es weiter mit einem einfachen Mausklick, diesmal auf **INSTALL**.



Im nächsten Dialogfeld wird nun der Fortschritt der Installation angezeigt.



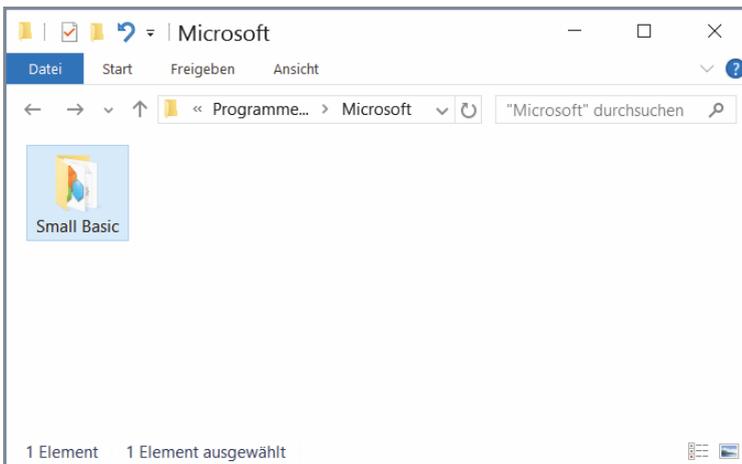
- Zum Schluss erscheint ein letztes Dialogfeld und mit einem Klick auf **FINISH** bringst du die ganze Installation zu Ende.



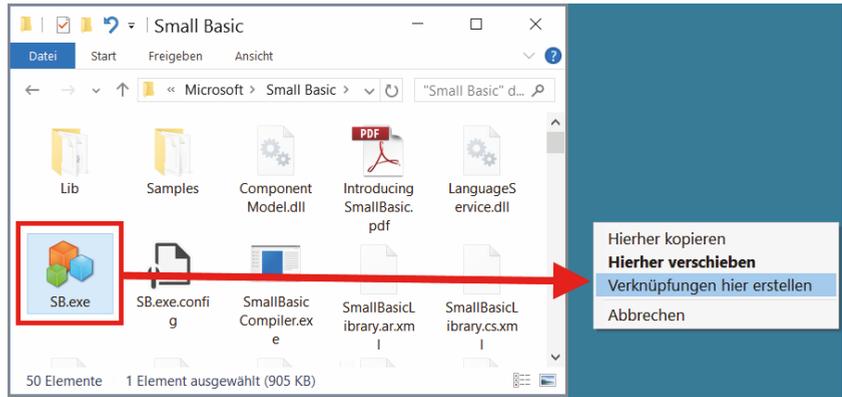
Small Basic starten

Und wie kriegt man das neue Programm jetzt zum Laufen? Eine Möglichkeit, Small Basic zu starten, ist diese:

- Öffne den Ordner, in den Small Basic installiert wurde, er heißt: `C:\PROGRAMME\MICROSOFT\SMALL BASIC`.



- Hier suchst du unter den vielen Symbolen eines derjenigen heraus, die aussehen wie eine gekippte Lila-8, und zwar das mit dem Namen SB.EXE.

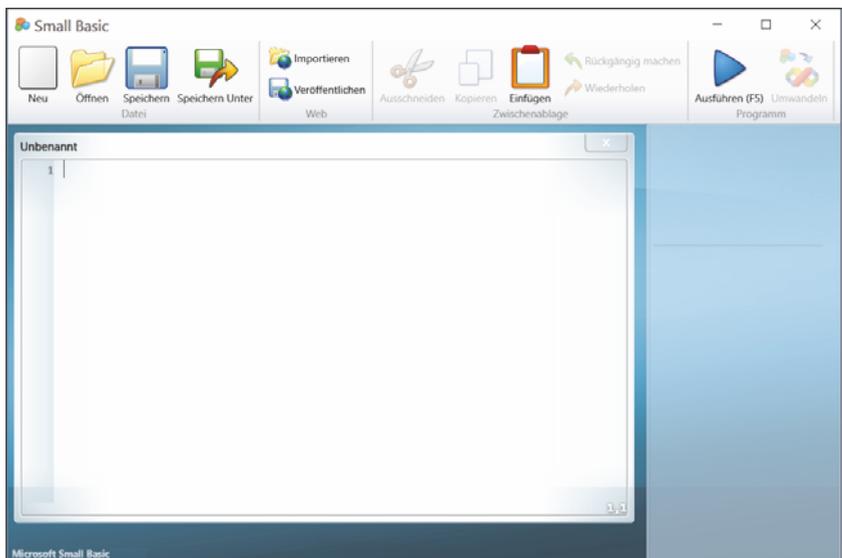


➤ Nun kannst du das Programm mit einem Doppelklick auf das Symbol starten:



Du kannst den Eintrag mit gedrückter rechter Maustaste auch direkt auf den Desktop ziehen. Im Kontextmenü wählst du VERKNÜPFUNG. Dann hast du dort ein Symbol für den direkten Start von Small Basic. (Das Symbol solltest du in SMALL BASIC umbenennen.)

Nach einer kurzen Aufwärmphase macht sich das Fenster von Small Basic auf dem Desktop breit.



Ganz oben findest du eine Leiste mit Symbolen, die man mit der Maus anklicken kann. Das Wichtigste dazu fasse ich mal kurz zusammen:

- ❖ Über die DATEI-Gruppe lassen sich deine Programme öffnen und speichern.
- ❖ Du kannst im WEB veröffentlichte Programme importieren (und selber welche veröffentlichen).
- ❖ Zum Bearbeiten von Programmteilen dürfen natürlich die Werkzeuge für die ZWISCHENABLAGE nicht fehlen.
- ❖ Damit du testen kannst, ob der PC auch das tut, was du im PROGRAMM vorgegeben hast, musst du es ausführen. (Und falls du später einmal auf die Profi-Umgebung Visual Basic umsteigen willst, kannst du deine Small-Basic-Projekte dafür umwandeln.)



Die große linke Fläche unter der Menüleiste ist der **Editor**. Dort gibst du deinen Programmtext ein. Ein anderer Name für Programmtext ist **Quelltext** oder **Quellcode**.

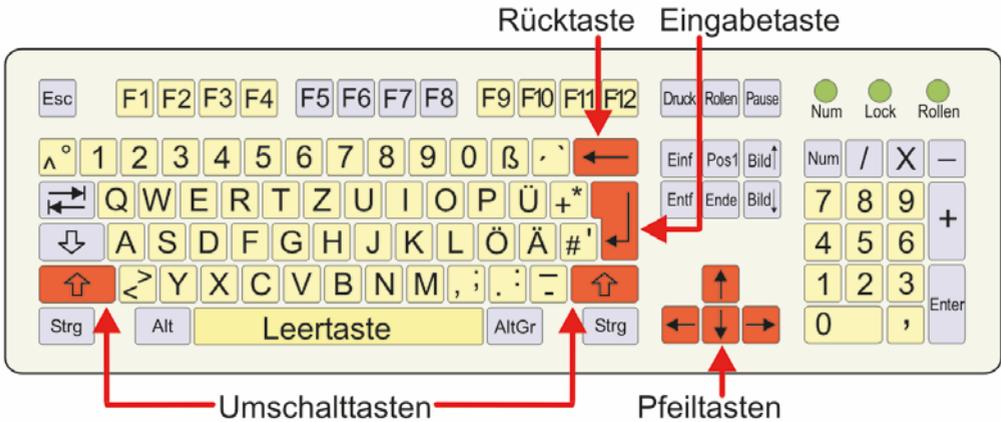
Tasten suchen und finden

Beim Eintippen eines Programmtextes wirst du feststellen, dass es bestimmte Tasten gibt, die man zwar wieder und wieder braucht, aber nicht immer gleich findet. Deshalb schauen wir uns jetzt mal gemeinsam die Tasten an, auf denen besondere Zeichen stehen oder die eine besondere Bedeutung haben.

Du kannst das Spiel dann selbst noch mal beliebig oft wiederholen. Das erspart dir einiges an Zeit, wenn du dich später voll auf ein Programm konzentrieren kannst und nicht ständig nach einer Taste suchen musst.

Als Erstes verschaffen wir uns noch mal einen Überblick über die Tasten zur Steuerung der Bearbeitung:

1



Die üblichen Anführungsstriche kriegst du mit `⌘` und `2` auf den Bildschirm. Die einfachen mit `⌘` und `#`.

Anführungsstriche



Immer wieder nötig sind in Small Basic Klammern, und zwar zwei Sorten. Die runden findest du mit `⌘` und `8` oder `9`. Und die eckigen mit `AltGr` und `8` oder `9`.

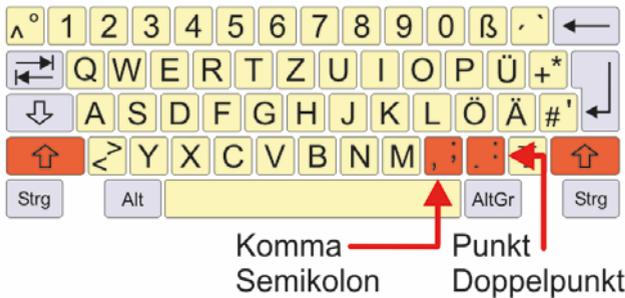
Klammern



Auch das Gleichheitszeichen taucht häufig in Programmen auf. Das erhältst du mit `⌘` und `0`.



Eine Menge Verwechslungen gibt es bei Punkt und Komma, Doppelpunkt und Semikolon – vor allem für Brillenträger, weil diese Zeichen so winzig sind. Da solltest du lieber zweimal hinschauen oder dir die Stellen genau merken, wo die Tasten sitzen.



Ein Programm eingeben und starten

Du möchtest jetzt endlich richtig loslegen? Genügend Platz für die Eingabe von Programmtext hast du ja.

Hier wartet der Editor schon auf ein neues Programm. Ansonsten genügt ein Klick auf NEU, um ein neues und leeres Editorfenster zu öffnen.



Eine weitere Möglichkeit ist die Tastenkombination `Strg N`.

