

# INHALT DER QBASIC PROGRAMMIER HILFE

PDF-Konvertierung von Thomas Antoni --- [www.qbasic.de](http://www.qbasic.de) --- Hottest QBasic Stuff on Erarth

■	<a href="#">Erste Schritte</a>
■	<a href="#">Quick Basic starten</a>
■	<a href="#">Die Benutzeroberflaeche</a>
■	<a href="#">Das erste Programm</a>
■	<a href="#">Jetzt wird's farbig</a>
■	<a href="#">Variablen</a>
■	<a href="#">Der SYSTEM Befehl</a>
■	<a href="#">Der INPUT Befehl</a>
■	<a href="#">Der IF - THEN Befehl</a>
■	<a href="#">Der SCREEN Befehl</a>
■	<a href="#">Der PSET Befehl</a>
■	<a href="#">Der LINE Befehl</a>
■	<a href="#">Der CIRCLE Befehl</a>
■	<a href="#">Der PAINT Befehl</a>
■	<a href="#">Der SLEEP Befehl</a>
■	<a href="#">Der BEEP Befehl</a>
■	<a href="#">Der PLAY Befehl</a>
■	<a href="#">Der RANDOMIZE TIMER(Zufallszahlengenerator)</a>
■	<a href="#">Der GOSUB Befehl</a>

## Anleitung : Erste Schritte

---

Hier lernst du, wie man programmiert. Vom kleinen "Hallo-wie-geht-es-dir-Programm" bis zum komplexen Raumschiff-Baller-Spiel. Ich moechte dich darauf hinweisen, das Quick Basic ein ziemlich altes und einfaches Programm zur Erstellung von anderen Programmen ist. Und noch was: Es macht am meisten Spaß!!!

## Quick Basic starten

---

Oeffne das Verzeichnis, in dem sich bei dir Quick Basic befindet. Kopiere die darin enthaltenen Daten auf eine Leere Diskette. Das Programm muß sich auf einer Diskette befinden, da es sonst nicht laufen will. Oeffne das Diskettenlaufwerk. Klicke doppelt auf die Datei Qbasic.exe. Du wirst nun nach einem "Parameter" gefragt. Gib' einfach nichts ein und klicke auf O.K. Dann kann es passieren, dass eine Frage, die irgendwas mit dem Arbeitsverzeichnis zu tun hat erscheint, drücken sie "Ja". Jetzt sagst du : Drin! Das war ja einfach!

Tip: Sollte die Frage erscheinen : BlaBlaBlaGeblubberGeblubber---Im MS-Dos Modus starten Ja/Nein kann ich dir nicht weiterhelfen. Wenn ich zum Beispiel "Ja" druecke dann geht es. Wer weiss, vielleicht stuerzt bei dir der Computer ab..... Aber wenn es einmal geht, dann geht es immer ! Also : AUF EIGENE GEFAHR. Das sollte dich aber nicht abschrecken "Ja" zu druecken! Ich schreibe das hier nur weil ich keine Schuld haben will wenn mal was passiert ! Um diese Meldung (und die Gefahr) zu umgehen, starte den MS-Dos Modus und gib folgendes ein:

A:

Qbasic.exe

Natuerlich muss sich Qbasic schon auf der Diskette befinden!

## Die Benutzeroberflaeche

---

Du siehst, nachdem du QuickBasic gestartet hast, ein großes, rechteckiges blaues Ding. Das ist das Bearbeitungs-Fenster. Oben siehst du, wie in Windows auch, die Menuepunkte. Im Bearbeitungs-Fenster schreibst du den sogenannten "Programmcode" rein. Er sagt dem Computer, was er tun soll. Es ist zwar alles ziemlich blau und sieht kompliziert aus, ist es aber nicht !

## Das erste Programm

---

Starte QuickBasic, klicke auf den Menuepunkt "Datei" und waehle "Neu". Kapiert? Erst Datei, dann "Neu". In unserem ersten Programm wollen wir einfach "Hallo" auf den Bildschirm schreiben. Dazu muessen wir dem Computer erst mal klarmachen, das er "Hallo" schreiben soll. Um das zu erreichen, musst du in das Bearbeitungs Fenster den sogenannten Befehl PRINT schreiben. Also : Schreibe PRINT. Freizeichen. Dahinter kommt ein suesses, kleines Anfuhrungszeichen. Hinter das Anfuhrungszeichen kommt ein "Hallo" oder "Hi" oder was weiss ich. Jetzt wieder ein Anfuhrungszeichen. Das mueßte dann so aussehen :

```
PRINT "Hallo"
```

Damit sagen wir dem Computer : Schreibe bitte mal Hallo auf den Bildschirm. Starte das Programm, indem du auf AUSFUEHREN und dann auf START klickst. Wie du siehst, steht jetzt "Hallo" auf dem Bildschirm. Fuehre das Programm weitere Male aus. Du wirst sehen, dass, jedes mal, wenn du das Programm startest ein "Hallo" dazukommt. Das liegt daran, das dieses "Hallo" nicht wieder geloescht wird. So kommt bei jedem Start ein "Hallo" dazu. Um das zu verhindern, muessen wir dem Computer sagen, das er den Bildschirm bitte mal saeuern soll. Das tun wir, indem wir einfach eine Zeile ueber dem PRINT-Befehl, CLS hinschreiben. Das sollte dann so aussehen :

```
CLS
```

```
PRINT "Hallo"
```

Fuehre das Programm erneut aus, und siehe da ! Jetzt steht nur noch ein mal "Hallo" auf dem Bildschirm. Auch nach mehrfachen Starten, steht immer nur ein "Hallo" auf dem Bildschirm. Speichere dein Programm ab, indem du erst auf Datei klickst, dann auf Speichern. Speichere das Programm unter Hallo1.bas ab.

Du hast jetzt folgende Aufgaben zu bewaeltigen :

Schreibe ein Programm, in dem der Computer "Hallo ! Ich bin ein einfaches Programm !" auf den Bildschirm schreibt. Speichere ab unter DeinName1.bas .

## Jetzt wird`s farbig !

---

Ist es dir zu langweilig mit nur einer Farbe ? Kann ich verstehen ! Um dir zu zeigen, das es auch anders geht, oeffne erst mal Hallo1.bas . So. Das mit den Farben ist ganz einfach ! Was heist Farbe auf Englisch? Richtig, COLOR ! Damit haben wir schon einen neuen Befehl! Gib mal COLOR zwischen CLS und PRINT ein. Freizeichen. Jetzt gib hinter COLOR eine Zahl von 0 bis 15 ein. Das ganze mueßte dann so aussehen:

```
CLS
```

```
COLOR deinezahl
```

```
PRINT "Hallo"
```

Damit sagst du dem Computer : Benutze fuer Schrift und Grafik die Farbe, die ich dir mit COLOR nenne ! Starte das Programm! Du wirst sehen, das die Schrift jetzt eine Farbe hat! Speichere ab unter Hallo2.bas, indem du auf DATEI/SPEICHERN UNTER klickst. Hier eine Aufschluesselung der Farben :

Erst die Farbe, dann die Zahl

Schwarz 0

Dunkelblau 1

Dunkelgruen 2

Cyan 3

Rot 4

Lila 5

Orange 6

Hellgrau 7  
 Dunkelgrau 8  
 Hellblau 9  
 Hellgruen 10  
 Tuerkies 11  
 Hellrot 12  
 Rosa 13  
 Gelb 14  
 Weiß 15

Die Aufgaben, die du jetzt zu bewaeltigen hast :

Schreibe ein Programm, das dem Computer sagt : "Tolles Programm". Farbe : Hellgruen. Vergiß nicht, den Bildschirm am Anfang des Programms zu saeuubern ! Speichere ab unter Deinname2.bas !

## Variablen

Fuer alle, die nicht wissen, was Variablen sind: Variablen sind so etwas aehnliches wie Platzhalter. Bei der Programmierung sind Variablen das wichtigste, was es gibt. Man gibt ihnen Namen, wie zum Beispiel "Geld" und weist ihnen Werte zu, wie zum Beispiel 250. Bei der Programmierung rechnet man mit Variablen, man laeßt sich den Wert einer Variable auf den Bildschirm schreiben oder legt einen neuen wert fuer eine Variable fest. Ein Beispiel: Mache ein neues Programm, indem du auf DATEI/NEU klickst. Unsere Variable heit "Geld". Erstmal weisen wir der Variable Geld einen Wert zu. Sagen wir mal 5. Dazu schreibe mal ganz an den Anfang des Programms: geld! = 5. Jetzt hat die Variable Geld den Wert 5. Das Ausrufungszeichen ist da fuer da, das der Computer wei, was das fuer ein Variablentyp ist.

! ist Zahlenfolge

\$ ist Buchstabenkette

Nun lassen wir uns mal auf den Bildschirm schreiben, was fuer einen Wert die Variable hat. Dazu schreibe direkt unter den Befehl geld! = 5 , PRINT "Du hast "; geld! ;" DM". "Was soll denn das ???", denkst du dir jetzt wahrscheinlich. Das Semicolon (;) unterbricht den Schriftzug "Du hast DM", um zwischen dem "hast" und dem "DM" den Wert der Variable geld! zu schreiben. Wenn du nur den Wert der Variable geld! schreiben willst, dann wuerde das so aussehen :

```
PRINT geld!
```

Es geht aber auch noch anders :

```
PRINT "Der Wert der Variable geld! ist ";geld!
```

```
PRINT geld! ;" DM hast du"
```

Bis jetzt sollte dein Programm so aussehen :

```
geld! = 5
```

```
PRINT "Du hast "; geld! ; " DM"
```

Jetzt wollen wir mal ein bisschen rechnen! Schreib doch mal unter das PRINT , geld! = geld! - 3. Das sagt dem Computer, der neue Wert der Variable geld! ist das Ergebnis der Aufgabe geld! - 3. Darunter schreiben sie PRINT "Du hast noch "; geld! ;" DM" . Dein Programm muelte jetzt so aussehen :

```
geld! = 5
```

```
PRINT "Du hast ";geld!;" DM"
```

```
geld! = geld! - 3
```

```
PRINT "Du hast noch ";geld!;" DM"
```

Starte das Programm. Na? Hat dein Computer richtig gerechnet ? Ich hoffe doch! Aber was haben wir noch vergessen ? Richtig, das CLS am Anfang des Programms! Ergaenze es und speichere unter Var1.bas. Ach, uebrigens : Variablen lassen sich ueberall einsetzen !

Du hast noch folgende aufgaben zu bewaeltigen:

Schreibe ein Programm , das irgend etwas mit Variablen ausrechnet und in der Farbe Hellrot auf den Bildschirm schreibt, und Speichere es unter Deinname3.bas ab.

## Der SYSTEM Befehl

Hast du dich schon mal gefragt, wie man ein Programm schlagartig beendet? Nein ? Das ist fuer spaetere , komplexere Spiele seeeeeehr wichtig. Das einzige , was du tun musst, ist : Setze an die Stelle, an der du beenden willst ein Befehl namens SYSTEM. Das wars ! Mehr steckt nicht dahinter !

Und hier eine aufgabe :

oeffne ALLE bisherigen Programme, und setze den Befehl SYSTEM an die entsprechende stelle.

## Der INPUT Befehl

INPUT. Was heißt das wohl? INPUT sagt dem Computer : Lass den Benutzer mal was eingeben und speichere es unter einer Variable. Ach übrigens: Der Index im Menü Hilfe ist eine große Hilfe :-) Mache ein neues Programm. Als erstes musst du was eingeben? CLS! Wir wollen doch einen leeren Bildschirm für unser Programm! Danach PRINT "Wie geht es dir?". Und jetzt kommt der große Moment... Wir geben den Befehl INPUT ein. Freizeichen. Jetzt müssen wir eine Variable eingeben. Sagen wir mal eine Buchstabenkette. Die Variable heißt wiegehts. Wie war doch gleich die Endung fuer eine Buchstabenkette? Schau bei Variablen nach, wenn du sie nicht weisst. Nagut, nagut die Endung heisst \$ (auch als "String" bezeichnet). Das ganze muesste jetzt so aussehen:

```
CLS
```

```
PRINT "Wie geht es dir?"
```

```
INPUT wiegehts$
```

So. Was heißt das jetzt eigentlich? Bei CLS wird der Bildschirm gereinigt. Bei PRINT wird "Wie geht es dir?" auf den Bildschirm geschrieben. Bei INPUT soll man was eingeben, das dann in der Variable wiegehts\$ gespeichert wird. Wenn man "Scheisse" eingibt, hat die Variable den Wert "Scheisse". Die letzte Zeile: Sag dem Computer, das er auf den Bildschirm schreiben soll: "Dir geht es also was-man-eingegeben-hat !". O.K. hier die Aufloesung. Gib PRINT "Dir geht es also ";wiegehts\$ ein. Das Programm muesste dann so aussehen :

```
CLS
```

```
PRINT "Wie geht es dir?"
```

```
INPUT wiegehts$
```

```
PRINT "Dir geht es also "; wiegehts$
```

Führe das Programm aus.

Hier einige Aufgaben:

Schreibe ein Programm, das dich nach mehreren Sachen fragt und dich immer wiederholt. Speichere unter Deinname4.bas ab.

## Die IF - THEN Anweisung

IF - THEN heisst für den Computer in Programmiersprache genau das, was es für uns auf deutsch auch heisst. Das Ganze funktioniert z.B. so: WENN die Variable zumbeispiel\$ den Wert "Hallo" hat, DANN schreibe "Hallo". Ende des Anweisungsblocks. Das sieht in Programmiersprache so aus:

```
IF zumbeispiel$ = "Hallo" THEN
```

```
PRINT "Hallo"
```

```
END IF
```

Schreiben wir mal ein Programm dazu. Dieses Programm antwortet auf die Frage "Wie geht es dir?". Also los:

```
CLS
```

```
PRINT "Wie geht es dir"
```

```
INPUT antwort$
```

```
IF antwort$ = "gut" THEN
```

```
PRINT "Das freut mich!"
```

```
END IF
```

```
IF antwort$ = "schlecht" THEN
```

```
PRINT "Das tut mir leid!"
```

```
END IF
```

```
SYSTEM
```

Na? Fertig? Starte das Programm! Das wars schon ! Jetzt hast du alles noetige wissen, um ein primitives Spiel wie Starfurz nach zu programmieren ! Speichere unter ifthen1.bas

Hier ein paar Aufgaben:

Schreibe das Programm weiter, indem der Computer auf mehrere Antworten reagiert, und speichere unter Deinname5.bas

## Die SCREEN Anweisung

---

WENN DU GRAFIKEN BENUTZEN WILLST IST DIESER BEFEHL UNUMGAENGLICH !!!!!!!! Mit diesem Befehl stellst du den Bildschirm - Modus ein. So geht's :

SCREEN 12

Die erste Zahl ist der Bildschirm-Modus. Hier sind alle MIR bekannten Modi aufgelistet:

1 - Schrift Modus (Standart)  
12 - Grafik Modus 16 Farben  
13 - Grafik Modus 256 Farben

Und denk dran vor JEDEM Programm, in dem Grafiken vorkommen den SCREEN - Befehl zu verwenden !!!

## Die PSET Zeichen-Anweisung

---

Hast du Starfurz schon nachprogrammiert ? Ja ? Dann weißt du, dass es ohne Grafik ziemlich langweilig ist. PSET schafft Abhilfe. Es sagt dem Computer : Male einen Pixel(Bildpunkt). Mache ein neues Programm. Als erstes kommt der Bildschirm-Leeren-Befehl (Jetzt musst du selber wissen, wie der heißt !!!). Dann kommt was wohl ??? PSET ! So wird's gemacht :

SCREEN 12  
CLS  
PSET (50,50),4

Die erste Zahl in der Klammer ist die X-Koordinate des Pixels (Das heißt : Soviele Pixel weit rechts wird gezeichnet...). Die zweite Zahl in der Klammer ist die Y-Koordinate des Pixels(Soviele Pixel weit unten...). Die erste Zahl hinter der Klammer ist die Farbe (Zahl 4 = Rot oder siehe der COLOR Befehl). Schreibe jetzt ein eigenes Programm, in dem irgendwo auf dem Bildschirm ein Pixel in wasweissichfüreiner Farbe.

Deine Aufgabe: Ein kleines Muster malen lassen.

## Die LINE Zeichen-Anweisung

---

Wenn du die Aufgabe erfüllt hast(Wehe dir, wenn nicht!), dann weißt du, wie anstrengend es ist, jeden einzelnen Pixel einzeln zu setzen. Dann mal' doch ne' Linie. So gehts:

SCREEN 12  
CLS  
COLOR was-weiss-ich  
LINE (50,50)-(60,60)

Die ersten beiden Zahlen sind die Start-Koordinaten der Linie (X vor Y, wie immer). Die Zahlen hinter dem Bindestrich sind die End-Koordinaten der Linie. Die Farbe wird vom COLOR-Befehl festgelegt. So einfach geht das !

Deine Aufgabe : Lasse ein kleines Muster mit dem LINE - Befehl malen

## Die CIRCLE Zeichen-Anweisung

---

So wird ein Kreis gemalt :

```
SCREEN 12
CLS
COLOR was-weiss-ich
CIRCLE (50,50), 5
```

Die ersten beiden Zahlen sind die Koordinaten des Mittelpunktes des Kreises (X vor Y) und die Zahl danach ist die Größe des Kreises vom Mittelpunkt aus (in Pixeln)

Deine Aufgabe : Muster Malen

## Die PAINT Zeichen-Anweisung

---

Damit wird ausgemalt. So geht's :

```
SCREEN 12
CLS
PAINT (50,50),4,8
```

Die ersten beiden Zahlen sind Koordinaten für den Füllpunkt (X vor Y). Die dritte ist die Füll-Farbe, und die vierte ist die Farbe, bis zu der gefüllt wird. (Die Wand sozusagen...)

Deine Aufgabe : Muster malen :-)

## Die SLEEP Anweisung

---

Mit der Anweisung SLEEP sagt man dem Computer er soll doch mal warten. Das sieht dann ungefaehr so aus:

```
---Anweisungen---
SLEEP 1
---Anweisungen---
```

Die erste Zahl hinter SLEEP gibt die Anzahl der Sekunden an, die der Computer warten soll. Steht dort keine Zahl, wird so lange gewartet, bis eine Taste gedrückt wird. Dann würde das Ganze ungefaehr so aussehen :

```
---Anweisungen---
SLEEP
---Anweisungen---
```

Das geht doch eigentlich ganz einfach oder ?

Hier eine Aufgabe : Schreibe ein Programm mit SLEEP

## Die BEEP Anweisung

---

BEEP sendet ein Tonsignal an den im Computer integrierten Lautsprecher. Beispiel :

```
---Anweisungen---  
BEEP  
---Anweisungen---
```

VORSICHT - Das Tonsignal hoert sich bei jedem Computer anders an !

Hier eine Aufgabe :

Schreibe ein Programm mit BEEP und teste es (wenn möglich) auf mehreren Computern !

## Die PLAY Anweisung

---

PLAY sendet ein Tonsignal an den im Computer integrierten Lautsprecher. Beispiel :

```
---Anweisungen---  
PLAY "c"  
---Anweisungen---
```

Wie du siehst, steht hinter dem PLAY in Anführungszeichen ein c. Für alle, die im Musikunterricht nicht aufgepasst haben : C ist eine Note. Dieser Befehl sagt dem Computer also, er soll die Note C spielen. Die einzelnen Noten schreibt man einfach so hintereinander, wie hier:

```
---Anweisungen---  
PLAY "cdefgab"  
---Anweisungen---
```

Die Tonleiter ist hier allerdings etwas anders :

C  
D  
E  
F  
G  
A  
B

Wenn du die Oktave aendern willst, dann gib ein großes O und dahinter die Zahl der Oktave(0 - 6 Standard = 4). Also so :

```
---Anweisungen---  
PLAY "O3"  
---Anweisungen---
```

Diese Anweisung sagt dem Computer, dass er in Oktave 3 wechseln soll. Wenn wir aber die Tonlaenge bestimmen wollen, müssen wir ein großes L und dahinter eine Zahl von 1 - 64 (1 = Ganze Note 2 = Halbe Note 4 = Viertel Note ...) Das koennte dann so aussehen:

```
---Anweisungen---  
PLAY "L8"  
---Anweisungen---
```

Bei Pausen ist es dasselbe, ausser dass man statt dem L ein P einsetzt. Weiteres kann man unter Hilfe/Index/Play (Musik) erfahren. Hier mal ein Musikstück :

```
---Anweisungen---  
PLAY "L16O4P16cdefgabO5P16cdefgabO6P16cdefgab"
```

## Der RANDOMIZE TIMER

---

Dieser Zufallszahlengenerator ist einfach zu gebrauchen gibt aber keine 'echten' Zufallszahlen. Zugegeben weiß ich selbst nicht, wie man 'echte' Zufallszahlen hinkriegt. Aber erst einmal hierzu. Als erstes muß man ihn aktivieren. Dazu gibt man am Anfang des Programms folgendes ein :

```
---Programm Anfang---  
RANDOMIZE TIMER  
---Anweisungen---
```

Jetzt ist er aktiviert. Und wenn wir eine Zufallszahl von ihm wollen, geben wir das ein:

```
---Anweisungen---  
Variable% = INT(RND * Zahl1 + Zahl2)  
---Anweisungen---
```

Das heisst : Variable% ist ja klar... ; Das INT rundet auf eine gerade Zahl (sonst gibt's Kommas); Das RND ist der eigentliche Befehl um die Zufallszahl zu erhalten. Zahl1 ist die maximale Groesse der Zahl ; Zahl2 ist die minimale Groesse der Zahl. Und das war's .

## Der GOSUB Befehl

---

GOSUB kann man ganz einfach erklären:

GOSUB ist so eine art Wegweiser, der dem Programm sagt, wo es weitermachen soll. Hier ein Beispiel:

start:

-Anweisungen-

GOSUB start

Das "GOSUB start" sagt dem Programm, wo es hingehen soll, in diesem Fall zu "start". Das "start:" sagt dem Programm, dass das Programm hierher kommen soll, wenn an anderer stelle ein GOSUB sagt, dass es zu "start" gehen soll. Hinter diesen "Zielort" kommt immer ein Doppelpunkt.