

VZ200 Basic Cheat Sheet

REM <kommentar>	ein Kommentar	REM *** START ***
PRINT [<var1>;<...>;<varN>]	Ausgabe mit Zeilenumbruch	PRINT "HALLO";\$N
PRINT <...>;	Ausgabe ohne Zeilenumbruch	PRINT "SAG MAL WAS";
PRINT @<n>,<...>	setzt den Cursor an die angegebenen Position, $n=x+32*y$	PRINT @128,"ZEILE 5"
PRINT TAB(<n>);<...>	setzt den Cursor weiter zu angegebenen Position innerhalb der Doppelzeile	PRINT TAB(3);"3";TAB(20);"20"
PRINT USING <format>;<var1>,<...>,<varN>	Formatierte Ausgabe	PRINT USING „##.## / ##.##“;1.23, 0.75
INPUT <var1>[,<...>,<varN>]	Eingabe von einer oder mehrerer Werte	INPUT A\$, X, Y
NEW	Programmspeicher löschen	NEW
RUN [<n>]	Programm (ab Zeile n) starten	RUN 30
[LET] <var>=<wert>	Wert einer Variablen zuweisen	LET X=7 B\$="PETER"
LIST [[n]-[m]]	Programm (Zeilen m bis n) auflisten	LIST -100
GOTO <n>	zu Zeile n springen	GOTO 20100
STOP	das Programm unterbrechen	STOP
CONT	das Programm fortführen	CONT
END	das Programm beenden (kein CONT mehr möglich)	END
CLEAR [<n>]	dem Programm n Bytes Speicher reservieren (Default: 50 bytes)	CLEAR 5000
ABS(<n>)	liefert den positiven Wert einer Zahl	X = ABS(-12.33):REM 12.33
SGN(<n>)	liefert das Vorzeichen einer Zahl (-1, 0 oder 1)	X = SGN(-12.33):REM -1
SQR(<n>)	liefert die Wurzel einer Zahl	E = SQR(25):REM 5
LOG(<n>)	liefert den natürlichen Logarithmus einer Zahl	L = LOG(4.44):REM 1.49065
EXP(<n>)	liefert den Wert e^n	E = EXP(1):REM 2.71828
INT(<n>)	liefert den ganzzahligen Anteil einer Zahl	I = INT(1.9334):REM 1
RND(<n>)	liefert eine Zufallszahl zwischen 1 und n, oder eine rationale Zahl zwischen 0 und 1 für n=0	IF RND(2)=1 THEN PRINT "TOP" ELSE PRINT "FLOPP"
SIN(<n>)	liefert den Sinuswert einer Zahl	W=30:Y = SIN(W/180*3.1415):REM .499987
COS(<n>)	liefert den Cosinuswert einer Zahl	X = COS(W/180*3.1415):REM .866033
TAN(<n>)	liefert den Tangenswert einer Zahl	T = TAN(W/180*3.1415):REM .57733
ATN(<n>)	liefert den Arcus-Tangenswert einer Zahl	W = ATN(X/Y)/3.1415*180:REM 30
LEN(<string>)	liefert die Länge eines Strings	PRINT LEN("HALLO")
STR\$(<n>)	wandelt eine Zahl in einen String um	PRINT STR\$(1277):REM "1277"
VAL(<string>)	wandelt einen String in eine Zahl um	PRINT VAL("1277"):REM 1277
LEFT\$(<string>,<n>)	liefert die ersten n Zeichen eines Strings	L\$ = LEFT\$("NINA",2):REM NI
RIGHT\$(<string>,<n>)	liefert die n letzten Zeichen eines Strings	R\$ = RIGHT\$("NINA",2):REM NA
MID\$(<string>,<m>,<n>)	liefert n Zeichen ab Position m eines Strings	M\$ = MID\$("NINA",2,2):REM IN
ASC(<string>)	liefert den Ascii-Wert des ersten Zeichens eines Strings	PRINT ASC("A"):REM 65
CHR\$(<n>)	wandelt einen Ascii-Wert in einen String um	PRINT CHR\$(65):REM A
INKEY\$	liest ein Zeichen von der Tastatur, ohne zu warten	10 IF INKEY\$="" THEN 10
IF <bedingung> THEN <befehl-true> ELSE <befehl-false>	Bedingte Ausführung, je nach Ergebnis der Bedingung	IF A>=2 THEN PRINT "MEHRERE ERBSEN" ELSE PRINT "EINE ERBSE"

VZ200 Basic Cheat Sheet

= < > <= >= <>	Vergleichsoperatoren	
<bool> AND OR NOT <bool>	logische Und/Oder/Nicht-Verknüpfung	20 IF A=2 AND B=3 THEN 40
<n> AND OR NOT <m>	bitweise Und/Oder/Nicht-Verknüpfung	E = 3 OR 6:REM 7
FOR <var>=<n> TO <m> [STEP <s>]	Zählt var von n bis m (mit Schrittweite s)	FOR I=10 TO 1 STEP -1:PRINT I:NEXT
NEXT [<var>]	springe zum nächsten Schleifendurchlauf (der Indexvariable)	NEXT I
GOSUB <n>	Aufruf des Unterprogramms ab Zeile n	GOSUB 12345
RETURN	Unterprogramm beenden	RETUEN
DIM <var>(<m>[,<n>[,<o>]])	Arrayvariablen deklarieren (1, 2 oder 3-dimensional)	DIM R\$(2,5)
DATA <n1>[,<n2>,<...>]	Daten bereitstellen	DATA 1,2,"PILLE",3,4,"PALLE"
READ <var1>[,<var2>,<...>]	Daten in Variablen einlesen	READ A,B,M\$
RESTORE	Beim Lesen von Daten wieder von vorne beginnen	RESTORE
PEEK(<adresse>)	liefert Wert einer Speicherstelle	B = PEEK(28672)
POKE <adresse>,<wert>	schreibt einen Wert in den Speicher	POKE 28672,1:REM EIN "A" LINKS OBEN AM SCREEN
CLOAD ["<filename>"]	lädt ein Programm von Cassette	CLOAD "SUPERSPIEL"
CRUN ["<filename>"]	führt ein Programm von Cassette aus	CRUN "MEGASPOILER"
CSAVE ["<filename>"]	Schreibt ein Programm auf Cassette	CSAVE "MEINS"
VERIFY ["<filename>"]	Vergleicht auf Cassette gespeichertes Programm mit dem aktuellen Programm im Speicher	VERIFY "MEINS"
PRINT# "<filename>",<wert1>[,<wert2><...>]	Schreibt Werte auf Cassette	PRINT# "DATEN",1,2,"PILLE"
INPUT# "<filename>",<var1>[,<var2><...>]	liest Werte von Cassette	INPUT# "DATEN",A,B,M\$
MODE(<n>)	setzt den Grafikmodus (0 = Text, 1 = Hires)	MODE(1)
SET(<x>,<y>)	setzt einen Grafikpunkt	SET(10,10)
RESET(<x>,<y>)	löscht einen Grafikpunkt	RESET(9,10)
POINT(<x>,<y>)	liefert die Farbe eines Grafikpunkts	60 IF POINT(9,10)=0 THEN 100
COLOR [<vordergrund>][,<palette>]	Setzt Vordergrundfarbe (1-8) und / oder die Farbpalette (0-1)	COLOR 1,1
SOUND <note>,<länge>	Spielt einen Ton ab note: 0 = Pause; 1 = A2; 31 = D#5 länge: 1-4 = n/8; 5 = ¾; 7 = 1 ½; 6-9 = 1-3	SOUND 13,7:SOUND 0,4:SOUND 15,7
INP(<n>)	liest einen Wert von Port n	B=INP(123)
OUT(<n>,<wert>)	schreibt einen Wert in Port n	OUT(123,4)
USR(<n>)	Ruft eine benutzerdefinierte Maschinenroutine auf. Startadresse: 788e (30862) Wert n:7921 (31009)	POKE 30862,L0:REM LOWBYTE DES PRG POKE 30863,HI:REM HIGHBYTE DES PRG E=USR(1234)
LLIST, LPRINT	wie LIST und PRINT, aber auf Druckerport	LLIST LPRINT "HALLO DRUCKER"
COPY	macht einen Screenshot auf dem Druckerport	COPY