

C Programmierung – Kurzreferenz der wichtigsten C-Befehle – Für Einsteiger		https://www.evamariakiss.de/tutorial/c-programming	
→ Variablen und Konstanten	→ Eingabe (scanf bzw. scanf_s)	→ Ausgabe (printf)	→ Verzweigungen (if-else)
<pre>int a = 0, b = 10; float x = 0.0; double y = 0.0; // Zeichen char z = 'A'; // Zeichenkette char *text = "Hallo"; // Konstante const double PI = 3.14;</pre>	<pre>int a = 0; double x = 0; char z = ''; printf("Eingabe a: "); scanf("%d", &a); printf("Eingabe x: "); scanf("%lf", &x); printf("Eingabe z: "); scanf("%c", &z);</pre>	<pre>int a = 0; double x = 0; char z = ''; printf("a = %d, x = %.2lf, z = %c\n", a, x, z); char *text = "Ausgabe"; printf("%s: \n", text);</pre>	<pre>float betrag = 1000, zinss = 0; if (betrag > 50000) zinssatz = 1.0; else if (betrag > 10000) zinssatz = 0.5; else if (betrag > 0) zinssatz = 0.2; else zinssatz = -0.2;</pre>
→ Fallunterscheidung (switch-case)	→ Schleifen (while, for)	→ Funktionen ohne Rückgabewert	→ Funktionen mit Rückgabewert
<pre>int wahl = 0; printf("Zahl eingeben: "); scanf("%d",&wahl); switch(wahl){ case 1: printf("1. Wahl\n");break; case 2: printf("2. Wahl\n");break; case 3: printf("3. Wahl\n");break; default: printf("Fehler!\n"); }</pre>	<pre>int n = 0; while (1){ // Endlosschleife printf("%d, ", n); n++; if (n > 100) break; // mit Abbruch } double sum = 0.0; int i = 1; while (i <= 5) { // While sum += i; i += 1; } printf("Summe: %.2f\n", sum); double sum = 0.0; for(int i=1;i<=5;i++) // For sum += i; printf("Summe: %.2f\n", sum);</pre>	<pre>#include <stdio.h> // (1) Funktionsprototyp void trennzeile(); int main(void){ // (2a) Funktionsaufruf trennzeile(); printf("Hallo!\n"); // (2b) Funktionsaufruf trennzeile(); } // (3) Funktionsdefinition void trennzeile(void){ printf("*****\n"); }</pre>	<pre>#include <stdio.h> double mittel(int a, int b); int main(void){ int x = 2, y = 4; double m = mittel(x, y); printf("Mittelw = %lf\n", m); printf("Mittelw aus 50 und 100 = %lf\n", mittel(50, 100)); } double mittel(int a, int b) { double m = (a + b) / 2; return m; }</pre>
→ Arrays 1D	→ Arrays 2D	→ Adressen und Zeiger	→ Strukturen
<pre>// Array deklarieren // Max. 100 Elemente // Mit 0 initialisiert double x[100] = {0}; // Einige Werte zuweisen x[0] = 20; x[99] = 1.5; // Andere Werte zuweisen for (int i=0;i<10;i++) x[i] = 10*i + 0.5; // Array ausgeben for (int i=0;i<10;i++) printf("%.2lf, ", x[i]);</pre>	<pre>#include <stdio.h> #define M 10 // Anz. Zeilen #define N 10 // Anz. Spalten int main(void) { double a[M][N]; srand(1); // Zufallsgen. init. for(int i=0;i<M;i++){// Zeil for(int j=0;j<N;j++){// Spal // Zufallszahl < 100 a[i][j] = rand()%100; printf("%5.2f ",a[i][j]); } printf("\n"); } }</pre>	<pre>int a = 10; int *z = NULL; z = &a; // z zeigt auf Adresse a // Wert von a wird über z ... *z = 20; // ... auf 20 gesetzt printf("Wert von a = %d\n",*z); // p mit NULL initialisieren double *p = NULL; double list[] = {1.2, 2.3, 3.4}; // p zeigt auf list[0] p = &list[0]; p++; // Zeiger p inkrementieren // p zeigt jetzt auf list[1] printf("%.2lf\n",*p);</pre>	<pre>#include <stdio.h> struct Student { char* name; int matnr; }; typedef struct Student STUDENT; int main(void) { STUDENT st; st.name = "Max Muster"; st.matnr = 12345; printf("%s %d\n", st.name, st.matnr); }</pre>