

Shader Language Cheatsheet

[Signed Distance Funktionen](#)

[Website für Upload](#)

Programmiergrundbausteine

Name	Beschreibung
if(bool) {1} else {2}	Falls bool wahr ist, wird der Code in {1} ausgeführt, ansonsten der in {2}
for(i=0;i<x;i++){1}	Der Code in {1} wird x-mal ausgeführt

Nützliche Konstanten

Datentyp	Name	Beschreibung	Tipp
vec2	resolution	Die Auflösung des Bildschirms, x = Breite, y = Höhe	
vec2	uv	normalisierte Bildschirmauflösung: (0,0) = linke untere Ecke, (1,1) rechte obere Ecke	
float	time	Anzahl der gerenderten Frames nach starten des Shaders	Nützlich mit sin(), cos() um einen Loop zu erstellen
vec4	iMouse	Position der Maus, falls geklickt	Hier wird meistens nur iMouse.xy gebraucht

Nützliche mathematische Funktionen

Name	Beschreibung	Tipp
sin()	die Sinus Funktion	Nützlich um Loops zu erstellen
cos()	die Cosinus Funktion	Nützlich um Loops zu erstellen
abs()	gibt den Betrag eines Wertes aus	Um
min(a,b)	gibt den geringeren der beiden Werte zurück	Für Übergänge
max(a,b)	gibt den größeren der beiden Werte zurück	Für Übergänge

Nützliche Shader Language Funktionen

Name	Beschreibung
float length(vec)	gibt die Länge eines Vektors zurück
float clamp(x, min, max)	Begrenzt den Wert von von x zwischen min und max und gibt ihn zurück
float step(edge, x)	Wenn der Wert von x größer als der Wert von edge ist, wird 1 zurückgegeben, ansonsten 0
vec2 center(vec2 uv)	Erweitert das Koordinatensystem auf -1 bis 1 und setzt den Ursprung (0,0) in die Mitte
float smoothstep(edge0, edge1, x)	wie step, nur wird zwischen den beiden Kanten ein sauberer Übergang, wo der Rückgabewert zwischen 0 und 1 liegt, zurückgegeben
float fract(x)	gibt immer nur den Dezimalpunkt von x an, so dass der Rückgabewert immer zwischen 0 und 1 ist