

## OPENS CAD – PROGRAMMIERUNG

### 3D- Gestaltung mit OpenSCAD

Die Original-Website lautet: <https://openscad.org/>

„Spickzettel“ / Cheat Sheet – die Befehlsübersicht:

<https://openscad.org/cheatsheet/index.html>

Dieser ist mit allen möglichen Befehlen bereits verlinkt!

Weitere Infos sind auf <https://archive.org/search?query=openscad> zu finden. Hier werden einzelne Artikel mit Erklärungen aufgezeigt.

OpenSCAD ist eine kostenlose Software zum Erstellen von fundierten 3D-CAD-Objekten.

Die Software ist für Linux/UNIX, MS Windows und Mac OS X verfügbar. Unter „<https://openscad.org/downloads.html>“ wird für alle genannten Systeme das Programm zur Verfügung gestellt.

Unter anderem ist unter „<https://openscad.org/documentation.html>“ eine englischsprachige Dokumentation vorhanden.

OpenSCAD besitzt eine eigene integrierte Programmiersprache, die aus textbasierten Befehlen besteht. Damit werden die Grundlagen von 3D-Formen gezeichnet, die als Bausteine für alle Entwürfe dienen.

Diese Programmiersprache ist speziell für 3D entwickelt und leicht zu erlernen. Die Grundlage ist eine beschreibende Sprache, die auf natürliche Art mehr bietet, Geometrie zu beschreiben, als auf die traditionelle Art und Weise möglich wäre.

#### **Warum OpenSCAD verwenden?**

OpenSCAD ist ein Open-Source-Programm, das kostenlos heruntergeladen werden kann. Es ist eine der am weitesten verbreiteten 3D-Design-Softwareanwendungen in der Maker-Community, weshalb auch viele Online-Ressourcen dafür verfügbar sind.

Stattdessen wird durch textbasierten Code das Design definiert, wie das Verschieben verschiedener Teile, fortwährende Änderungsmöglichkeit in jedem einzelnen Schritt des Designprozesses, Teilen von Formen des eigenen Designs mit anderen Usern. OpenSCAD ermöglicht ähnliche Dinge wie andere High-End-Tools; es ist jedoch schneller zu erlernen, sowie einfacher zu bedienen.

OpenSCAD erschien 2009 von Marius Kintel und Claire Wolf, heute 2022 nur noch von Marius Kintel gehostet.

Soweit zum Original.

Meine Anleitungen stammen zum größten Teil aus dem Internet, von der Originalseite von OpenSCAD, aus englischen Büchern (Bücherei) und sonst noch woher. Jedes Mal zuerst grob übersetzen, ausprobieren, nachschlagen im Web, und so fort.

Da meine 3D-Drucker-Legende bis 2017 zum Anet A8 reicht (damals noch selbst zusammengebaut), über den Anycubic Mega S (kam 2020) ging, landete ich im September 2025 bei Elegoo Centauri Carbon.

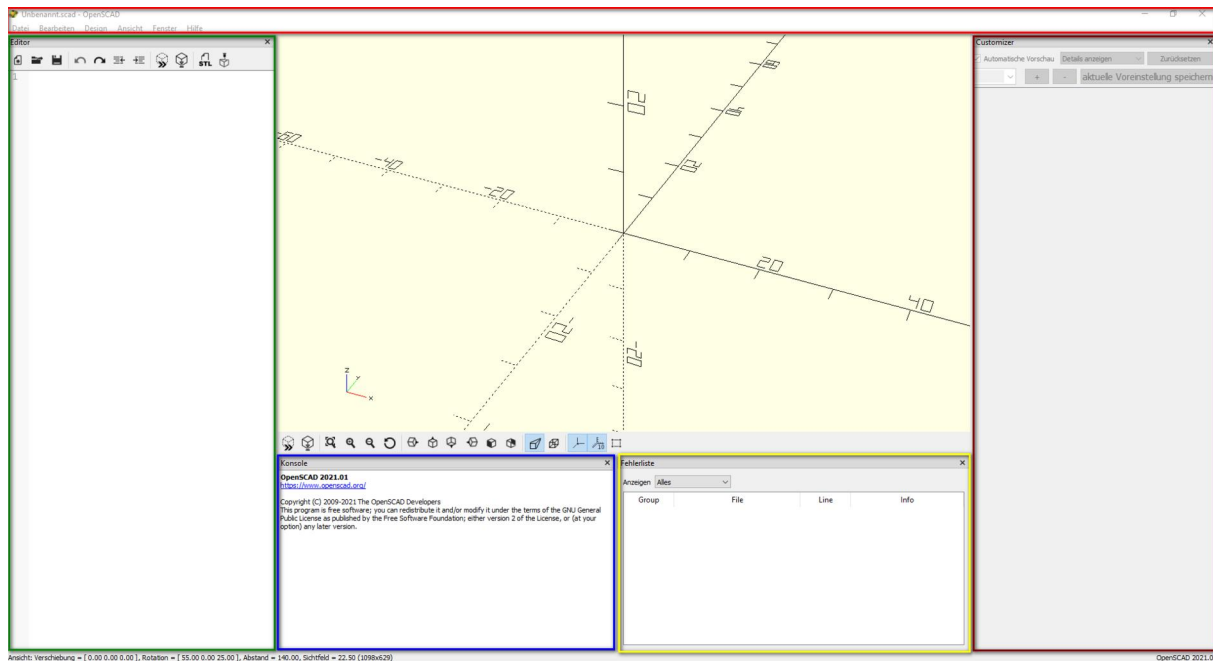
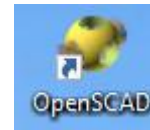
Seit Beginn pfriemelte ich mich durch die englischen Anleitungen von OpenSCAD. Nein, wirklich, es funktionierte nicht immer auf Anhieb. Plötzlich tauchte die Idee auf, die positiven Arbeiten auf Deutsch festzuhalten. So entstand im Laufe der Monate ein gefüllter Ordner und mein Wissen wuchs beständig. Nein, lediglich die wichtigsten Artikel habe ich ausgegraben und übersetzt. Es wird nie fertig werden. Ich denke nur an die library von BOSL2 ...

Lesen Sie sich seitenweise ein. Für fast alle Skripte ist ein Beispiel vorhanden. Anklicken und ohne zu Tippen erscheint es auf dem von Ihnen installierten OpenSCAD.

Über eine Rückmeldung freue ich mich. Fragen einfach ohne Zögern stellen, auch wenn sie noch so penibel erscheinen. Manchmal ist nur ein kleiner Anstoß zur zündenden Idee notwendig.

Viel Spaß mail [at] brixelweb.de

Nach dem Download und der Installation wird das Programm über dessen Symbol gestartet.



Ganz oben quer liegt die **Hauptmenuzeile** in rot. Darunter links ist der **Texteditor** in Grün für die Eingabe.

Unten in der Mitte ist die **Konsole** (links, blau) und die **Fehlerliste** (rechts, gelb) zu finden. Das rechte längliche Feld ist der **Customizer** (braun).

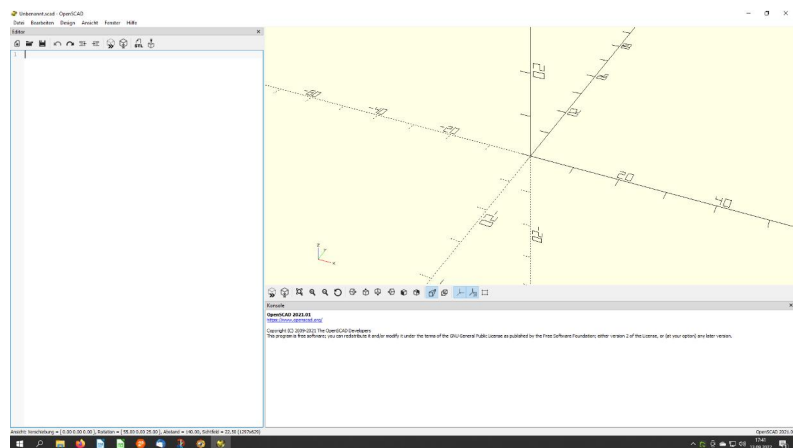
Im Zentrum selbst ist das **Vorschaufenster** zu sehen.

Für den Anfang wird die Fehlerliste und der Customizer nicht benötigt. Daher können diese über das jeweilige „X“ in der Titelleiste geschlossen werden. (Jedoch jederzeit über das Menu -> Fenster wieder erreichbar.)

Wird die Maus über die Rahmen von zwei Fenster geschoben, wird der Cursor zu einem Doppelbalken mit zwei Pfeilen. Damit lassen sich die Größen der einzelnen Fenster untereinander vergrößern und verkleinern.

001-04

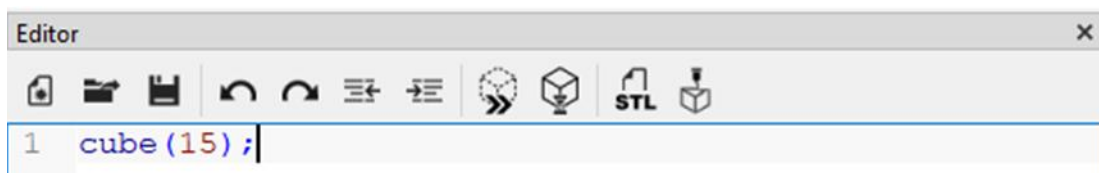
So werden die einzelnen Fenster nach Wunsch angepasst.



Im Texteditor werden die Skripte verfasst. Über die Taste **F5** wird es anschließend im Vorschaufenster angezeigt. Sollte ein Fehler im Script vorhanden sein, wird die Berechnung für die Vorschau abgebrochen und in der Konsole auf den Fehler darauf hingewiesen.

In OpenSCAD sind 2D-Objekte erstellbar, die ebenso in 3D extrudiert werden können. Alle 2D- und 3D-Objekte werden über den Texteditor als Script eingegeben.

Oben links im Texteditor wird folgendes eingegeben: `cube(15);`



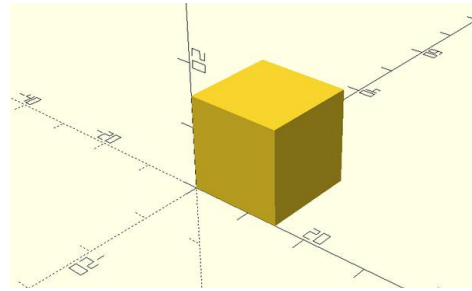
In der Symbolleiste des Editors sind zu finden:

	Neue Datei ( Strg + N ), Datei öffnen ( Strg + O ), Speichern ( Strg + S )
	Rückgängig (Strg + Z ) und Wiederholen ( Strg + Shift + Z )
	Einzug vermindern ( Strg + Shift + I ) und Einzug erhöhen ( Strg + I )
	Vorschau starten ( F5 ), Rendern ( F6 )
	STL-Datei erzeugen ( F7 ) und 3D-Druck ( F8 )

Das ist praktisch, denn nach der Eingabe im Texteditor kann dort gleich auf das Symbol zur Anzeige der Vorschau geklickt werden.

Oder einfach die Taste **F5** drücken.

Beides führt zum gleichen Ergebnis:  
Es wird ein Würfel mit der Kantenlänge  
von 15 angezeigt.



Auch unter dem Vorschauenfenster sind eine große Menge von Symbolen zu finden.

Die ersten zwei sind bereits bekannt:  
Vorschau (F5) und Rendern (F6).



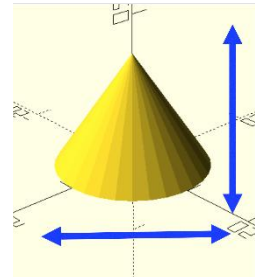
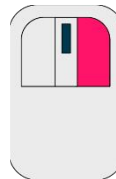
	<p>Jetzt folgen ein paar Symbole zur Ansicht: die Lupe mit den vier Ecken ist das Wichtigste: dies stellt das Objekt format-füllend dar.</p> <p>Vergrößern und verkleinern dürften auch klar sein. Mit dem Kreis-Pfeil-Zeichen wird die Anzeige zurückgesetzt.</p>
	<p>Ansicht von: rechts, oben unten und links, sowie von vorn und von hinten.</p>
	<p>Perspektivische (bereits blau markiert = aktiv) und orthogonale Ansicht.</p>
	<p>Koordinaten-Achsen anzeigen, Bemaßung anzeigen, Kanten anzeigen.</p> <p>Koordinaten-Achsen mit Bemaßung sind blau markiert = aktiv</p>

Die drei blau markierten Symbole sollten für ein angenehmes Arbeiten voreingestellt sein.

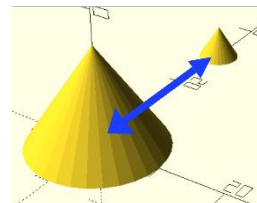
## Steuerung mit der Maus in der Vorschau

Im Vorschaufenster kann mit der Maus das Objekt verschoben, gezoomt und gedreht werden:

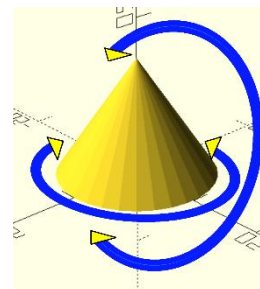
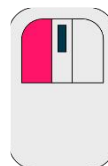
Mit der rechten Maustaste kann der Fensterausschnitt über dem Objekt verschoben werden: das Objekt (im Moment der Kegel) kann somit von einer zur anderen Ecke verschoben werden.



Das **Drehen vom Mausexplorer** vergrößert oder verkleinert das Objekt, ebenso bei gedrücktem Mausexplorer und der Auf- Ab-Bewegung der Maus.



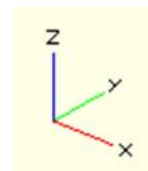
Die **linke Maustaste** bringt nun richtig Bewegung auf den Monitor: hiermit lässt sich das Objekt um alle Achsen drehen! Aber Vorsicht! **Schnell ist es auch aus dem Blickfeld verschwunden!**



Abhilfe schafft dann das Symbol

Es zentriert das Objekt wiederum auf dem Vorschaufenster.

Beim Drehen ist es sinnvoll das Symbol der Achsensymmetrie im Auge zu behalten – es hilft ungemein beim Orientieren!

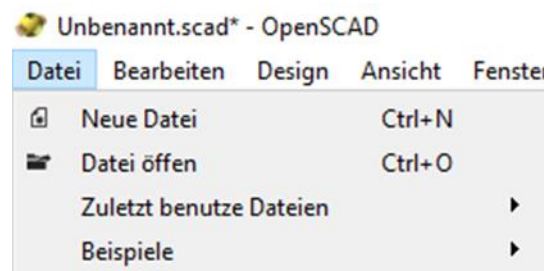


In der Hauptmenu-Zeile von OpenSCAD (direkt über dem Texteditor) unter „Fenster“ lassen sich alle einzelnen Fenster öffnen oder schließen.



Auch hier kann mittels „Datei“ eine Datei geöffnet oder gespeichert werden.

Sehr interessant finde ich auch die Funktion „Zuletzt benutzte Dateien“ als auch die Möglichkeit von „Beispiele“. Hier gibt es eine Fülle von Übungen.



Da OpenSCAD textbasiert arbeitet, entfällt das gewohnte Anklicken eines Piktogramms. Hier werden die Objekte als Script im Texteditor eingetragen, gleich mit den Größenangaben und dem Nullpunkt.

Die Standard-Objekte sind

Kugel	<code>sphere(Radius   Durchmesser)</code>
Würfel	<code>cube(Größe, Zentrum)</code>
Quader	<code>cube([Länge, Breite, Höhe], Zentrum)</code>
Zylinder	<code>cylinder(Höhe, Radius   Durchmesser, Zentrum)</code>
Kegel	<code>cylinder(Höhe, Radius1, Radius2 = 0, Zentrum)</code>
Kegelstumpf	<code>cylinder(Höhe, Radius1, Radius2, Zentrum)</code> oder <code>cylinder(Höhe, Durchmesser1, Durchmesser2, Zentrum)</code>

Ferner gibt es Polygon, Polyhedron und 2D-Objekte mit Extrusion.

Es bedeutet bei (**Radius | Durchmesser**), dass entweder der **Radius** oder der **Durchmesser** angegeben werden kann.

Die **Größe** beim Würfel bezieht sich auf dessen Kantenlänge.

(**Länge, Breite, Höhe**) sind die Angaben für die einzelnen Achsen:

Länge für die X-, Breite für die Y- und Höhe für Z- Achsen.

Es ist das jeweilige Vorzeichen (+/-) zu beachten.

001-08

**Nullpunkt:** das **Zentrum des Koordinatensystems**, es ist der Schnittpunkt aller drei Achsen. Als Standard gilt: center=false. Hierbei ist der Startpunkt des zu erstellenden Objektes genau im **Nullpunkt**. Da es als Standard festgelegt ist, kann es ebenso weggelassen werden.

Mit center=true liegt der **Mittelpunkt des Objektes** genau im Nullpunkt des Koordinatensystems.

Mittels **Radius1, Radius2 = 0** werden spitze Kegel erzeugt. Radius1 ist immer unten auf Null der Z-Achse (Höhe). Radius 2 ist oben, abhängig von der angegebenen Höhe. Da der obere Radius gleich Null ist, entsteht dadurch die Spitze.

**Radius1, Radius2:** damit werden bei einem Zylinder unterschiedliche Radien oder Durchmesser angegeben. So entsteht ein Kegelstumpf.

Wichtig: die **Kleinschreibung** bei OpenSCAD und jede Zeile mit einem zu erzeugenden Objekt wird mit Semikolon ( ; ) beendet.

Es sind nicht viele 3D-Objekte ... aber abwarten, was alles damit noch zusätzlich mit den Sonderfunktionen machbar ist!

Die schier unübersehbare Flut der Möglichkeiten überraschte auch mich!  
Denn aus 2D-Objekten über Extrusion ein paar 3D-Objekte zu erzeugen ist sehr spannend und interessant zugleich!

Bevor wir beginnen, muss die Schreibweise von OpenSCAD erklärt werden.

Die Befehlsübersicht (in der Hauptmenu-Zeile unter Hilfe aufzurufen) zeigt alle Befehle auf, die mit weiterführenden Erklärungen verlinkt sind.

Alles wird kleingeschrieben.

**( )** Hinter manchen Befehlen stehen die Werte in in runden Klammern ( und ).  
Beispiel: sphere(d = 20);

**([ ])** Bei anderen sind die eckigen Klammern **zusätzlich** notwendig ([ und ]). Nicht die runden außen vergessen. [ = AltGr + 8 und ] =AltGr + 9  
Beispiel: cube([9,14,13]); oder cube([13,12,8],false);



**{ }** Außerdem kommen geschweifte Klammern zum Einsatz: { und }  
{ = AlrGr + 7 und } = AltGr + 0

Ich habe mir beim Tippen angewöhnt, sofort die schließende Klammer zu setzen.

**//** Ferner setze ich gern den Doppel-Slash ein: /

```
// Kegelstumpf: oben breit, unten schmal  
cylinder(h=15, r1=5, r2=9);
```

Text wird somit eingefügt, um eine Beschreibung des Befehl zu erhalten. Es lassen sich später bestimmte Zeilen leichter auffinden.

**/\* \*/** Auch mehrere Textzeilen sind darzustellen:

```
/*  
Beispiel eines Kegelstumpfes  
Dateiname: Kegelstumpf  
*/  
  
// Kegelstumpf: oben breit, unten schmal  
cylinder(h=15, r1=5, r2=9);
```

Am Beginn steht ein /\* und am Ende ein \*/. Ohne Semikolon.

Zur Unterscheidung wird Text, den OpenSCAD bei der Ausführung überliest, in Grün angezeigt.

**;** Nach jedem Befehl muss als Abschluss ein Semikolon ; stehen, sonst ist der Befehl nicht beendet. (Deshalb hier in Rot)

Genug der Theorie. Am Schnellsten lernen wir an den Beispielen, die uns die Elemente erklären und am Monitor aufzeigen.