OpenSCAD – Der Schlüsselanhänger

Hier werdet ihr die Grundlagen von OpenSCAD lernen, um damit euer erstes 3D-Objekt zu erstellen. Dazu werdet ihr lernen, Kugeln, Zylinder, Quader und Text…

- zu erstellen,
- zu verschieben,
- und zu kombinieren.

[1]

Während des gesamten Moduls geben euch die Arbeitsblätter Hinweise zur Umsetzung. Achtet dabei einfach auf die folgenden Symbole, die…

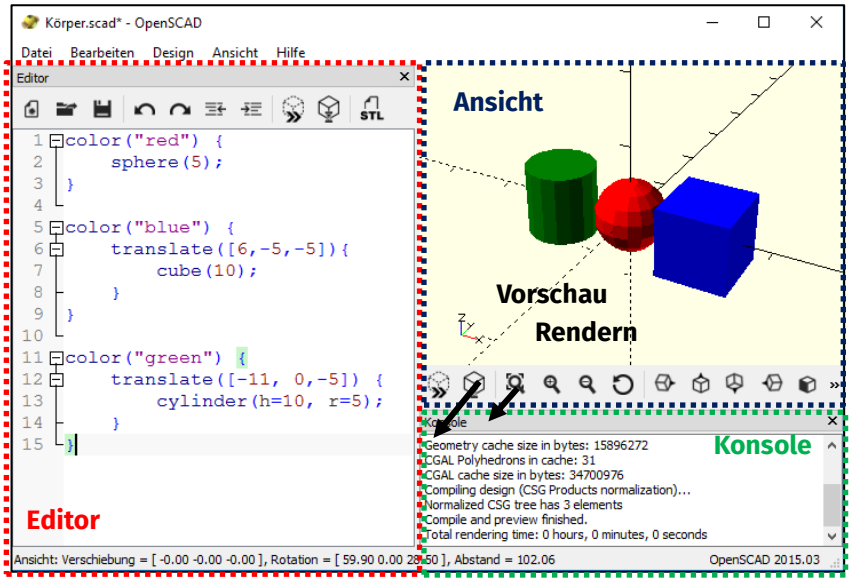
- euer Arbeiten strukturieren und Teilziele aufzeigen,
- euch Hilfen geben, Wichtiges, Schwieriges, etc. kennzeichnen und
- die Arbeitsaufträge und Aktionen beinhalten.

Die Software: OpenSCAD

Mit OpenSCAD lassen sich 3D-Objekte programmieren. Man kann praktisch alles aus 3 verschiedenen Körpern zusammensetzen: Kugeln, Zylindern und Würfeln.

Im **Editor** gebt ihr an, wie euer Objekt aussehen soll.

In der **Ansicht** könnt ihr sehen, wie euer Objekt aussieht. Dazu könnt ihr **Rendern (F6)** drücken, damit euer Objekt komplett berechnet wird, oder auf **Vorschau (F5)** für eine Schnellansicht.



[2]

In der **Konsole** seht ihr Fehlermeldungen, falls OpenSCAD nicht versteht, was ihr wollt, oder ihr irgendwo einen Fehler gemacht habt.

Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

Rendern dauert bei komplizierteren Objekten sehr lange. Darum gibt es eine Vorschaufunktion, die schnell ein Objekt erzeugt, aber dafür fehlerhaft sein kann.

Auf den folgenden Seiten lernt ihr, Körper...

- zu erstellen und
- zu verschieben,

um die Grundfläche des Schlüsselanhängers zu erstellen.

[3]

Der erste Körper: Zylinder

Zylinder haben einen Radius und eine Höhe, die beide in dem Programm angegeben werden:

```
cylinder(h= <Höhe>, r=<Radius>);
```

Wenn etwas in spitzen Klammern steht **<Beispiel>**, müsst ihr das durch eigene Werte ersetzen. Die eckigen Klammern müsst ihr **NICHT** in den Code schreiben!

Denkt dabei auch an das Semikolon **;**.

Beispiel auf dem Arbeitsblatt:

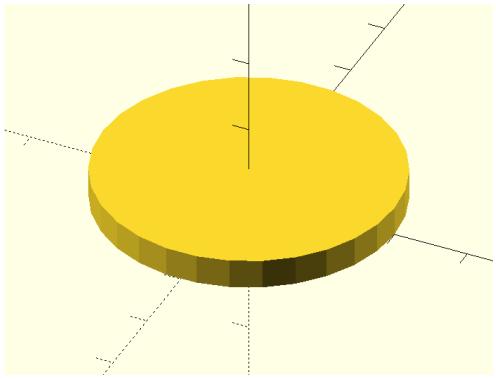
```
cylinder(h= <Höhe>, r=<Radius>);
```

Euer Code:

```
cylinder(h= 5, r=2);
```

```
//Rumpf, runder Zylinder

cylinder(r=20,h=4);
```



[4]

Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

Erstellt einen Zylinder, für den Rumpf mit Höhe 4 und Radius 20 eures Schlüsselanhängers.

Hinweis: Die Längeneinheit in OpenSCAD ist mm. Eine 6 entspricht 6 mm.

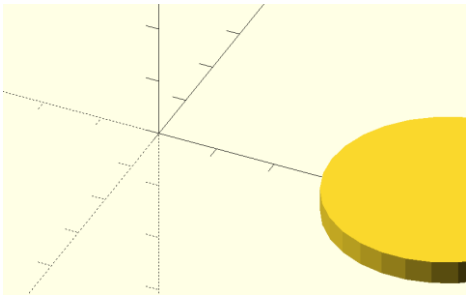
Platz für Neues

Damit ihr in der Mitte Platz habt, um weitere Körper zu erstellen, verschiebt ihr den Zylinder mit dem „Translate“-Befehl. Damit OpenSCAD weiß, um wie viel der Körper verschoben wird, wird für jede Achse ein Wert angegeben.

```
translate( [ <x-Wert>, <y-Wert>, <z-Wert> ] ) {
      <Körper>
}
```

```
//Rumpf, runder Zylinder

translate([50,0,0]){
    cylinder(r=20,h=4);
}
```



[5]

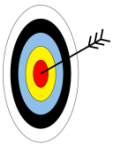
Ihr habt gerade eure erste **Funktion** kennengelernt. In OpenSCAD sind alle Funktionen gleich aufgebaut:

- In den runden Klammern werden Details für die Funktion angegeben (sog. **Parameter**).
- In den geschweiften Klammern („{“, „}“) wird angegeben, welche Körper verändert werden sollen.

Verschiebt den Rumpf-Zylinder um 50 mm zur Seite.

Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

Differenzen und Schnittmengen



Der Ring zum Aufhängen besteht aus einem Zylinder, aus dem ein anderer Zylinder herausgeschnitten wird.

[6]

Um den Ring zu konstruieren, braucht ihr zuerst einen Zylinder, in den ihr ein Loch stanzen könnt.

Erstellt einen zweiten Zylinder mit gleicher Höhe wie eure Grundfläche und einem Radius von mindestens 4 mm.

Verschiebt ihn an eine Kante eures Schlüsselanhängers.

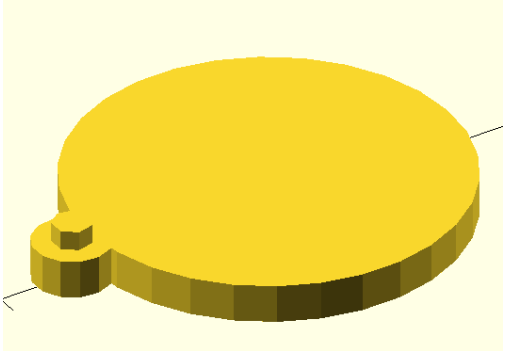
[7]

Nun benötigt ihr einen weiteren Zylinder für das Loch. Dabei sollte die Wandstärke eures fertigen Rings mindestens 2 mm betragen, damit alles am Ende stabil wird.

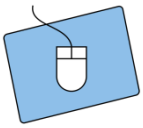
```
translate([50,0,0]) {
    cylinder(h=4, r=20);
}

translate([28,0,0]) {
    cylinder(h=4, r=4);
}

translate([28,0,0]) {
    cylinder(h=6, r=2);
}
```



[8]



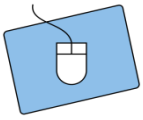
Erstellt einen weiteren Zylinder und verschiebt ihn an die Position, wo das Loch des Schlüsselrings sein soll.

Jetzt muss nur noch das Loch aus dem Schlüsselanhänger gestanzt werden. Dazu benötigt ihr die Funktion: `difference()` . Mit der Funktion `difference()` könnt ihr einen Körper aus einem anderen ausschneiden. Dabei werden aus dem ersten Körper alle weiteren ausgeschnitten:

Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

OpenSCAD – Der Schlüsselanhänger

```
difference() {
      <Körper>
      <auszuschneidender Körper 1>
      <auszuschneidender Körper 2>
      …
}
```



Benutzt `difference()`, um den Zylinder aus dem zusammengesetzten Körper auszuschneiden.

[9]

Weiter geht's auf der nächsten Seite…

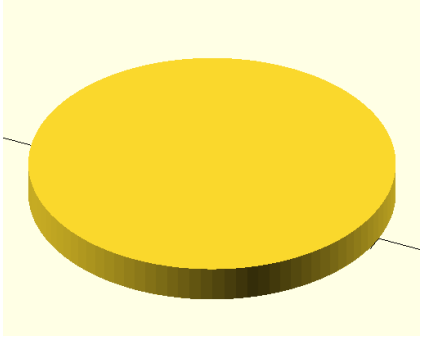
Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

Kanten glätten

Eure Zylinder sind nicht wirklich rund und passen nicht ganz zueinander, weil sich alle Rundungen durch eckige Flächen und Kanten zusammensetzen. Die Anzahl dieser Flächen könnt ihr mit **Feinheit** bestimmen:

```
cylinder( <Höhe/Radius/etc.> , $fn = <Feinheit>)

translate([50,0,0]) {
    cylinder(h=4, r=20, $fn=100);
}
```



[10]

Alle Zylinder und Kugeln sollen die gleiche Feinheit verwenden, und die Feinheit muss immer wieder mal angepasst werden. Damit man in dem Fall nicht für alle Kugeln und Zylinder die Feinheit neu eingeben muss, benutzt man eine sog. **Variable**, wie ihr sie auch schon aus der Mathematik, z. B. für den Radius oder den Satz des Pythagoras, kennt.

Eine Variable legt man fest mit:

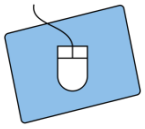
```
<Name> = <Wert>;
```

Anschließend wird überall, wo ihr den Namen der Variablen angebt, der gleiche Wert verwendet.

```
feinheit = 50;

translate([50,0,0]){
    cylinder(r=20,h=4,$fn=feinheit);
}

translate([50,0,4]){
    cylinder(r1=20, r2= 16,h=2,$fn=feinheit);
}
```



1. Ändert bei allen Zylindern die Feinheit.
2. Benutzt eine Variable, um bei beiden Zylindern die Feinheit auf den gleichen Wert zu setzen.

Je höher die Feinheit gewählt wird, desto länger wird der Computer zum Rendern (Darstellen) eures 3D-Objekts brauchen. Nehmt darum keine Zahlen größer als 100 für die Feinheit. Ihr könnt am Ende immer noch die Feinheit ändern.

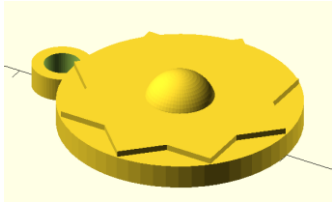
Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.

Das war es auch schon. Wenn noch Zeit ist bis zum Druckbeginn, sucht euch eine Bonusaufgabe aus, um euren Anhänger zu individualisieren.

Bonusaufgabe A: Verzierungen

Es gibt noch weitere Funktionen und Körper, die ihr benutzen könnt:

- `sphere(<Radius>,$fn=<Feinheit>);`
- `cube([<Breite>,<Länge>,<Höhe>], center = <true/false>);`
- `rotate() { … }`
- `resize() { … }`
- `intersection() { … }`
- `union() { … }`
- …



[11]

Bonusaufgabe B: Text

Um Text zu erstellen, braucht ihr zwei Befehle: `text()` und `linear_extrude()`.

`text()` erstellt einen zweidimensionalen Text:

```
text("<Hier kommt euer Text>", size = <Schriftgröße>, halign="center",
                                valign="center");
```

halign und **valign** bestimmen die Ausrichtung des Texts.

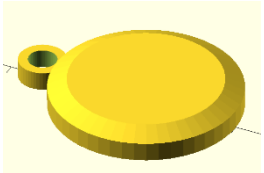
`linear_extrude()` kann einem zweidimensionalen Objekt eine Höhe zuweisen, z. B. eurem Text:

[12]

```
linear_extrude( height = <Höhe> ) {
      text("<Euer Text>");
}
```

Bonusaufgabe C: Kegel sind auch nur Zylinder

Kegel sind in OpenSCAD auch nur Zylinder, bei denen man zwei verschiedene Radien angibt: einen unteren Radius r1 und einen oberen Radius r2:



[13]

```
cylinder( h = <Höhe>, r1 = <Radius unten>, r2 = <Radius oben> );
```

Quellenverzeichnis:

Abb. 1-13 – *Quelle: Screenshots der OpenSCAD Software (http://openscad.org), GNU General Public License (https://www.gnu.org/licenses/gpl-3.0.html)*

, , , – *Quelle: InfoSphere*

Dieses Werk ist lizensiert unter einer Creative Commons Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (http://creativecommons.org/licenses/by-sa/4.0/). Die Lizenz einzelner Materialien kann wie gekennzeichnet abweichen (siehe Quellenverzeichnis). Ausgenommen von dieser Lizenz sind außerdem alle Logos, insbesondere das RWTH- sowie das InfoSphere-Logo.