

# { } LET'S CODE BLACKSBURG!

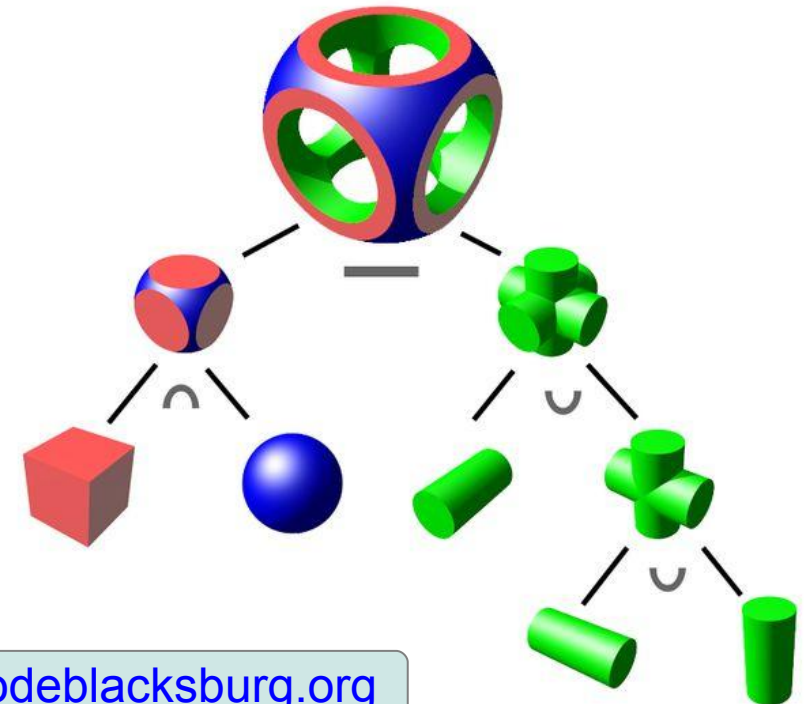
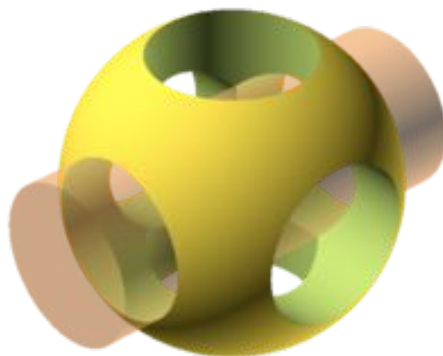


**General Series - 3D Printing Design  
Software Comparison (& More!)**

**OpenSCAD Overview**

Jim Stoll

[jstoll@letscodeblacksburg.org](mailto:jstoll@letscodeblacksburg.org)



[www.letscodeblacksburg.org](http://www.letscodeblacksburg.org)



# OpenSCAD Overview

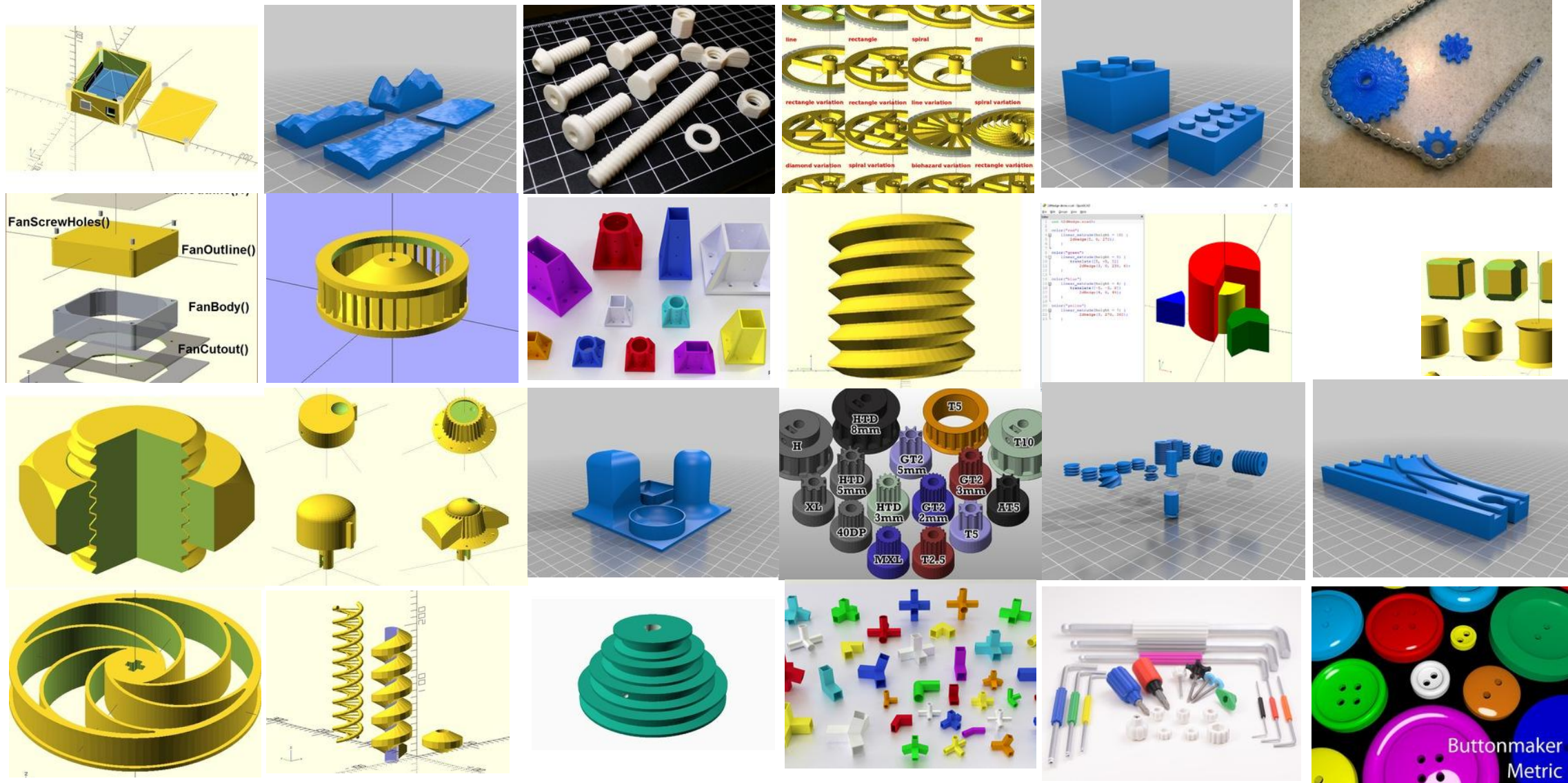
---

---

- Script-based 3D modeling
- **Constructive Solid Geometry**
  - Designs are **Constructed** via Operations and Transformations on **Solid Geometry Primitives**
    - All designs are a combination of Cubes, Cylinders, Spheres and Polyhedrons!
- Designer has precise numerical control over all objects
- Models are parameterized, so easy to adjust and tweak
- Wide assortment of libraries for complex objects
- OpenSCAD is based on a **Functional Programming** philosophy



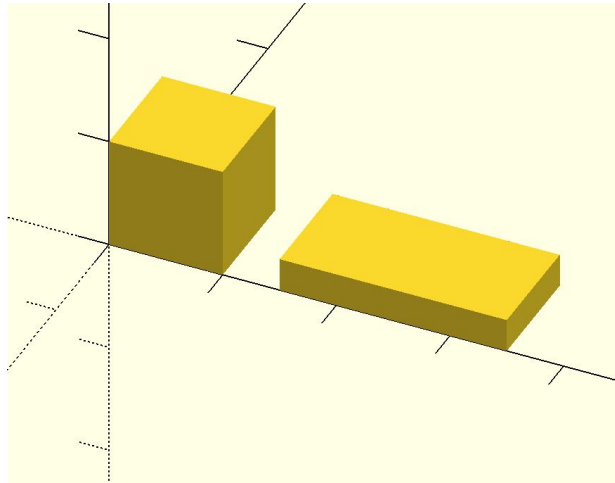
# A Few OpenSCAD Libraries





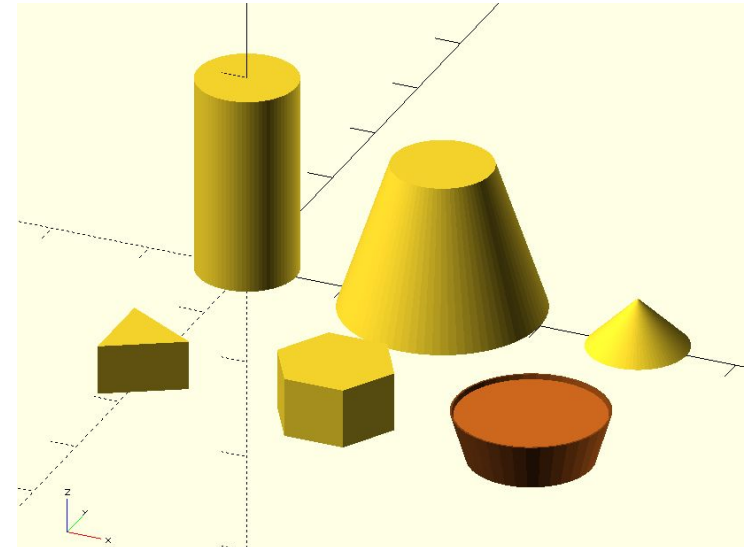
# OpenSCAD Primitives

```
cube(10);  
cube([20, 10, 3]);
```

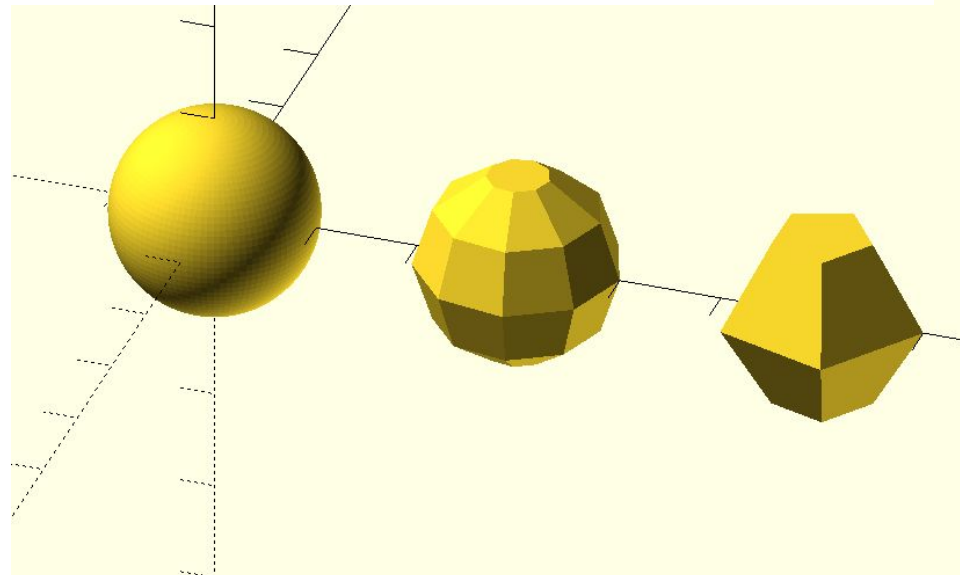


```
cylinder(r=5, h=20);  
cylinder(r1=10, r2=5, h=15);  
cylinder(r1=5, r2=0, h=5);
```

```
cylinder(r=5, h=5, $fn=3);  
cylinder(r=6, h=6, $fn=6);
```



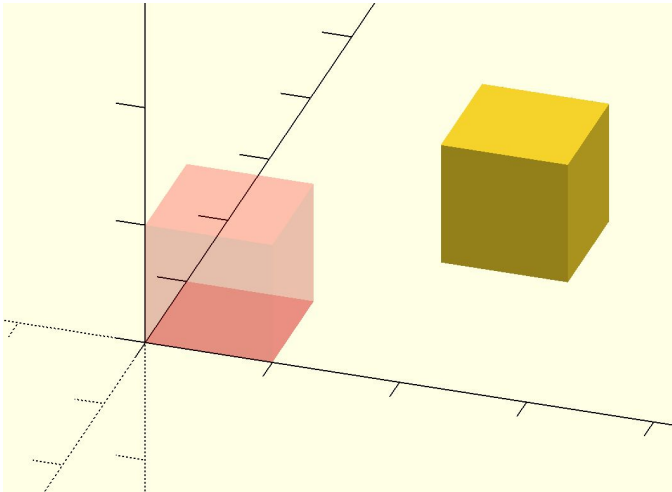
```
sphere(10, $fn=100);  
sphere(10, $fa=40);  
sphere(10, $fa=90);
```



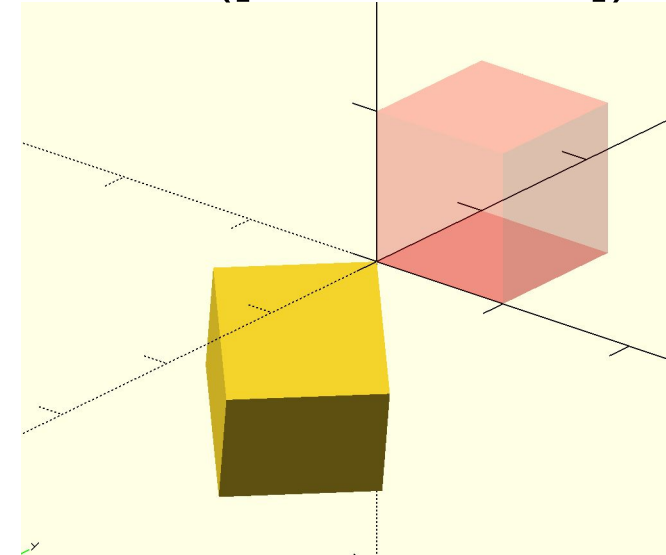


# OpenSCAD Transformations

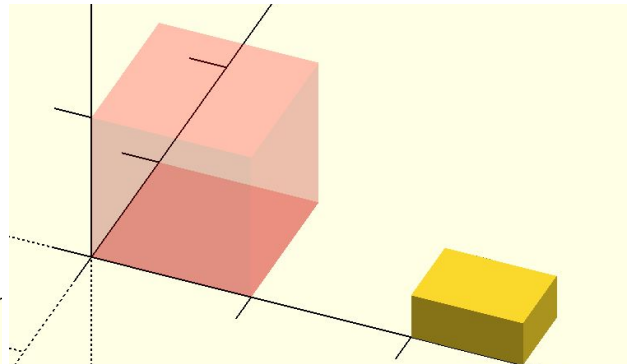
`translate([20, 10, 5]);`



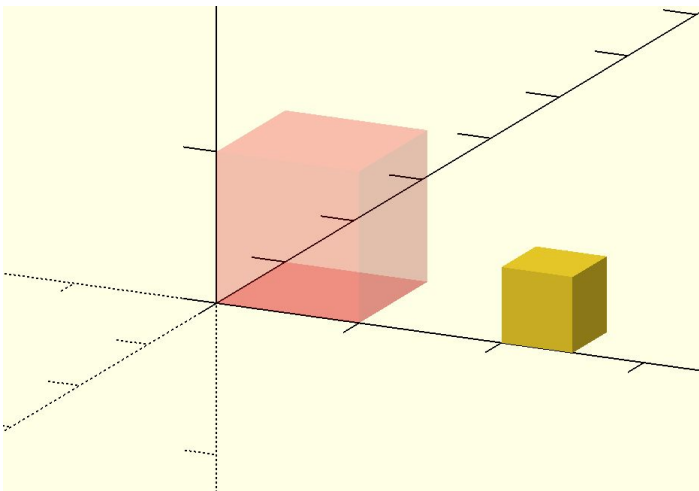
`rotate([90, 120, -45]);`



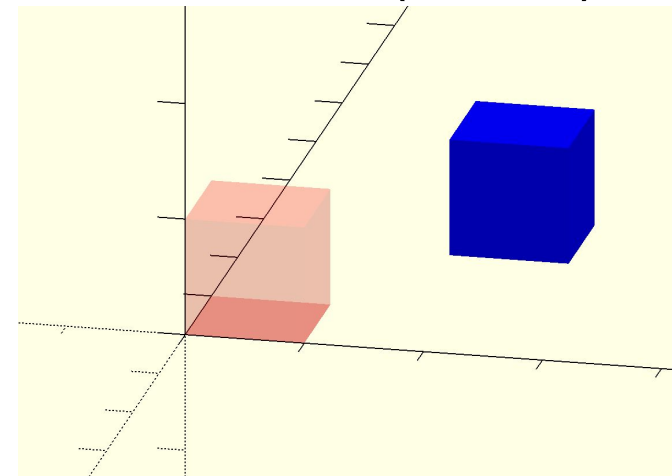
`resize([7, 5, 3]);`



`scale([.5, .5, .5]);`



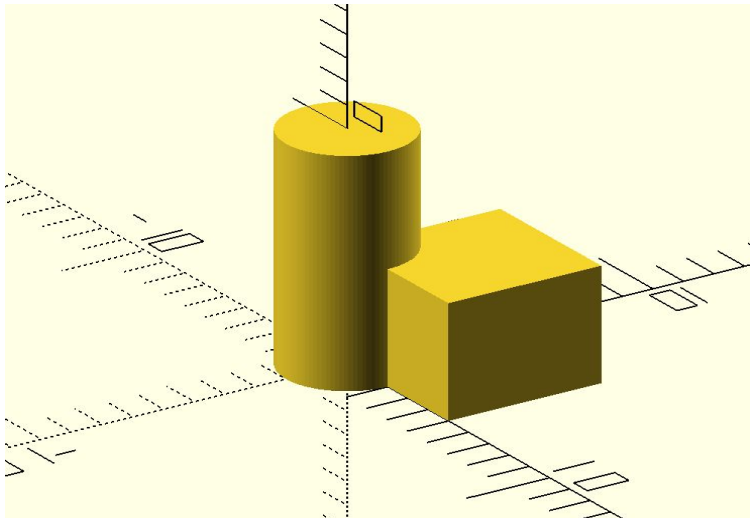
`color("Blue");`



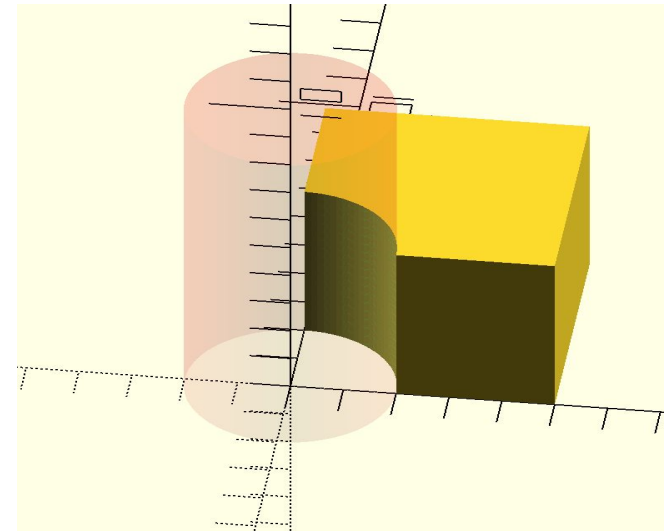


# OpenSCAD Boolean Operations

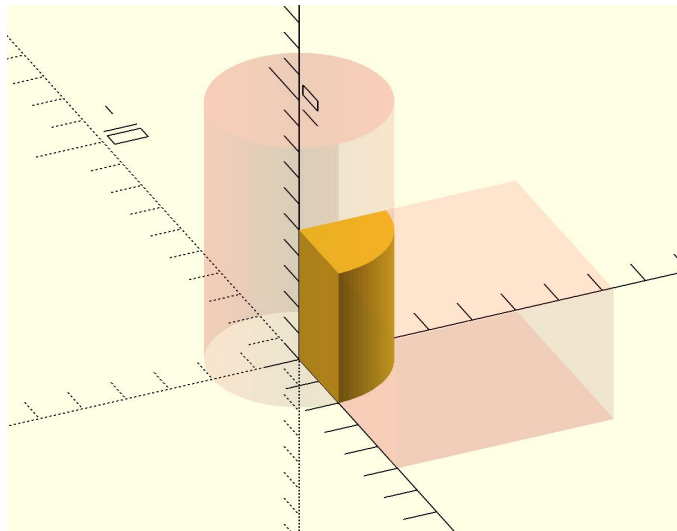
```
union() {  
  cube(10);  
  cylinder(r=2, h=10);  
}
```



```
difference() {  
  cube(10);  
  cylinder(r=2, h=10);  
}
```



```
intersection() {  
  cube(10);  
  cylinder(r=2, h=10);  
}
```





# Advantages/Disadvantages

---

---

## OpenSCAD Advantages

- Text based scripting language
  - Easy to see what's going on
  - Easy to see changes between versions
  - Highly compatible with Git workflows
- Very Precise Control
  - **Everything** is defined and controlled by the designer
- Supports Modular Design
  - Easy to build libraries, reuse code & designs
- Highly Parameterizable
- Works with Thingiverse Customizer
- Good for Functional/Mechanical designs

## OpenSCAD Disadvantages

- Text based scripting language
  - Not everyone enjoys the math/text-based approach to design
- **EVERYTHING** is defined and controlled by the designer
  - complicated models can become a bit overwhelming (good modular design really helps!)
- Does not enforce Modular Design
  - If scripts are not built in a modular fashion, complicated designs can quickly become an untamed beast
- Not a great tool for artsy, flowy type of designs



# OpenSCAD Environment

The screenshot displays the OpenSCAD interface with three main sections:

- Scripting Area:** A code editor on the left containing SCAD code for a key fob. The code defines parameters for dimensions, hole radius, loop width, text, and uses the `difference()` and `module` functions to create a 3D model.
- Rendering Area:** A 3D view on the right showing a yellow key fob with the name "Jeremy" engraved on it. The fob has a hole at one end. The background is a light yellow grid.
- Console Area:** A terminal window at the bottom right showing the output of the rendering process, including file paths, compilation steps, and performance statistics like "Total rendering time: 0 hours, 0 minutes, 0 seconds".

```
1 //key fob base length (X dimension)
2 fob_length = 50;
3
4 //key fob width (Y dimension)
5 fob_width = 20;
6
7 //key fob thickness (Z dimension)
8 fob_thickness = 3;
9
10 //radius of key loop hole
11 hole_radius = 5;
12
13 //width of key loop
14 loop_width = 3;
15
16 //offset of key loop hole - percent of hole diameter
17 hole_offset_percent = 50; //:[0:100]
18
19 //text to appear on the fob
20 name = "Jeremy";
21
22 //thickness of text (Z dimension) - positive values will protrude, negaive values will inset
23 text_thickness = .75;
24
25 //border spacing between edge of fob and text
26 text_border = 3;
27
28 //text font to use (can be any Google fonts from: https://fonts.google.com/)
29 text_font = "Classic Comic";
30
31 difference() {
32   fob();
33
34   //if negative text_thickness, then sink text into face, and difference it from the fob
35   //if positive text_thickness, then nothing will be differenced from the fob
36   if (text_thickness < 0)
37     translate([0, 0, text_thickness + .01])
38       name();
39 }
40
41 //if positive text_thickness, then just render it at its default Z position on the face
42 if (text_thickness > 0)
43   name();
44
45 //the rectangular body of the fob, with the offset key loop
46 module fob() {
47   cube([fob_length, fob_width, fob_thickness]);
48
49   translate([fob_length - hole_radius + 2*hole_radius * hole_offset_percent/100,
50             fob_width/2, 0]) {
51     difference() {
52       cylinder(r=hole_radius + loop_width, h=fob_thickness);
53       translate([0, 0, -.004])
54         cylinder(r=hole_radius, h=fob_thickness + .008);
55     }
56   }
57 }
```

Viewport: translate = [ 29.00 10.00 1.87 ], rotate = [ 55.00 0.00 25.00 ], distance = 270.90

OpenSCAD 2015.03