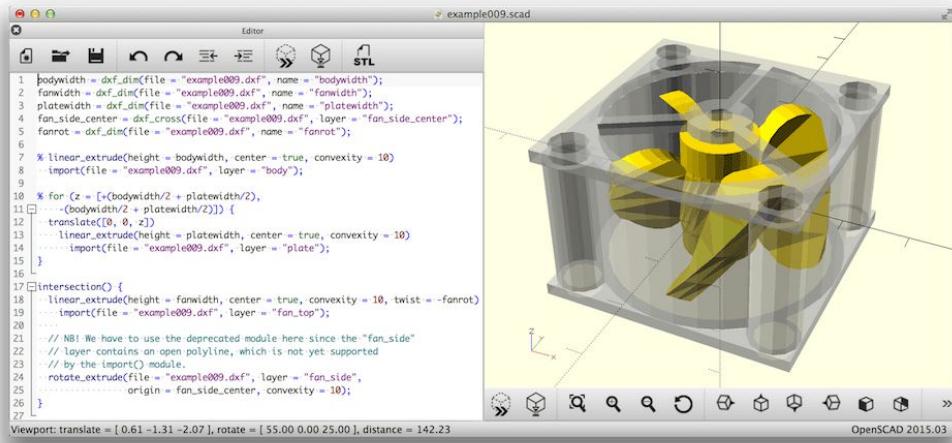


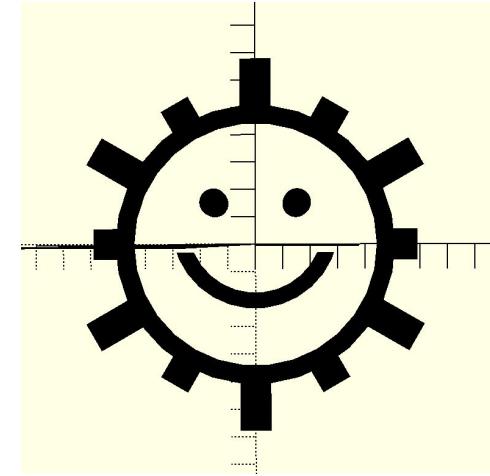
CAD for programmers!!!

by Lucas Gelfond



The screenshot shows the OpenSCAD 2015.03 software interface. On the left is the code editor with the file name "example009.scad". The code defines a fan assembly with various components like a body, fan blades, and a plate. On the right is the 3D preview window showing a yellow fan blade and a grey housing.

```
1 bodywidth = dxf_dim(file = "example009.dxf", name = "bodywidth");
2 fanwidth = dxf_dim(file = "example009.dxf", name = "fanwidth");
3 platewidth = dxf_dim(file = "example009.dxf", name = "platewidth");
4 fan_side_center = dxf_cross(file = "example009.dxf", layer = "fan_side_center");
5 fanrot = dxf_dim(file = "example009.dxf", name = "fanrot");
6
7 % linear_extrude(height = bodywidth, center = true, convexity = 10)
8 % import(file = "example009.dxf", layer = "body");
9
10 % For (z = [-bodywidth/2 + platewidth/2],
11 % -bodywidth/2 + platewidth/2)) {
12 % translate([0, 0, z])
13 % linear_extrude(height = platewidth, center = true, convexity = 10)
14 % ... import(file = "example009.dxf", layer = "plate");
15 %
16
17 intersection() {
18     linear_extrude(height = fanwidth, center = true, convexity = 10, twist = -fanrot)
19     ... import(file = "example009.dxf", layer = "fan_top");
20
21 // NB! We have to use the deprecated module here since the "fan_side"
22 // layer contains an open polyline, which is not yet supported
23 // by the import() module.
24 rotate_extrude(file = "example009.dxf", layer = "fan_side",
25                 origin = fan_side_center, convexity = 10);
26 }
27
Viewport: translate = [ 0.61 -1.31 -2.07 ], rotate = [ 55.00 0.00 25.00 ], distance = 142.23
```

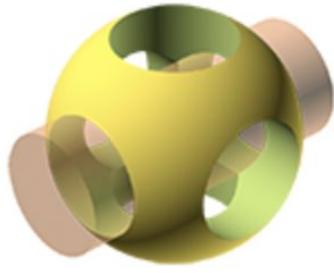


download if you haven't already

- download OpenSCAD (~27 MB)
 - <http://openscad.org/downloads.html>

OR

- brew install openscad
- the official docs (linked on their site) are *incredible*



OpenSCAD

The Programmers Solid 3D CAD Modeller

The screenshot shows the OpenSCAD application interface. On the left is the 'Editor' window containing the following SCAD code:

```
1 bodywidth = dxf_dim(file = "example009.dxf", name = "bodywidth");
2 fanwidth = dxf_dim(file = "example009.dxf", name = "fanwidth");
3 platewidth = dxf_dim(file = "example009.dxf", name = "platewidth");
4 fan_side_center = dxf_cross(file = "example009.dxf", layer = "fan_side_center");
5 fanrot = dxf_dim(file = "example009.dxf", name = "fanrot");
6
7 % linear_extrude(height = bodywidth, center = true, convexity = 10)
8 - import(file = "example009.dxf", layer = "body");
9
10 * for (z = [-(bodywidth/2 + platewidth/2),
11 ---(bodywidth/2 + platewidth/2)]) {
12     translate([0, 0, z])
13     linear_extrude(height = platewidth, center = true, convexity = 10)
14     - import(file = "example009.dxf", layer = "plate");
15 }
16
17 intersection() {
18     linear_extrude(height = fanwidth, center = true, convexity = 10, twist = -fanrot)
19     - import(file = "example009.dxf", layer = "fan_top");
20 ...
21 // NB! We have to use the deprecated module here since the "fan_side"
22 // layer contains an open polyline, which is not yet supported
23 // by the import() module.
24 - rotate_extrude(file = "example009.dxf", layer = "fan_side",
25     origin = fan_side_center, convexity = 10);
26 }
27
```

Below the code editor is a status bar with the text: 'Viewport: translate = [0.61 -1.31 -2.07], rotate = [55.00 0.00 25.00], distance = 142.23'. At the bottom right of the interface is the text 'OpenSCAD 2015.03'.

The main window on the right displays a 3D preview of a mechanical part, specifically a fan assembly, showing the yellow fan blades and the surrounding housing and mounting holes. The 3D view includes a coordinate system and various toolbars at the bottom.



Customizable Garden Sign

by [TheNewHobbyist](#) April 23, 2013



[Download All Files](#)

[Open in Customizer](#)

[Collect Thing](#)

[Like](#) →

[Comment](#)

[Post a Make](#) →

[Watch](#)

[Remix it](#) →

[Copy Link](#)

[Tip Designer](#)

Share this thing



[BACK TO CUSTOMIZABLE GARDEN PLANT LABE...](#)Now Using: [Customizer](#)

Queue

▼ parameters

Text String

<https://www.thingiverse.com/ap>[Copy](#)[View Source](#)[>Create Thing](#)

Text Size



Text Thickness

Text Font

Text Y Offset Adjust

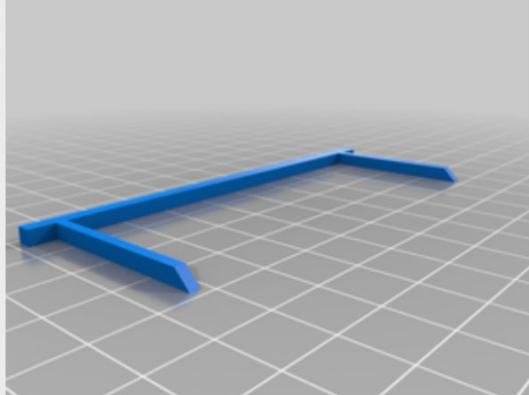


Fidget Camp

[Thing Details](#)**2**
Thing Files**17**
Comments**16**
Makes**2381**
Remixes**4**
Apps

Garden Sign Lavender

Aug 11, 2020

[+ Collect Thing](#)

BAYLEAFGarden Sign

Jun 29, 2020



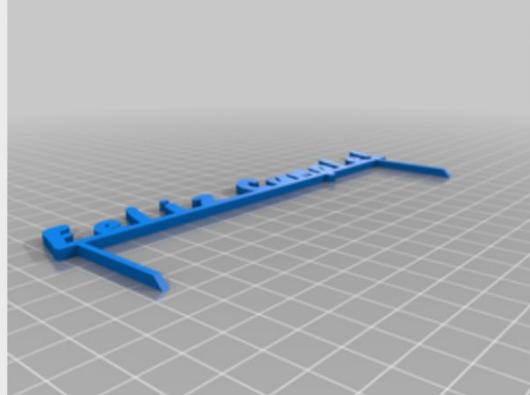
OREGANO Garden Si...

Jun 29, 2020



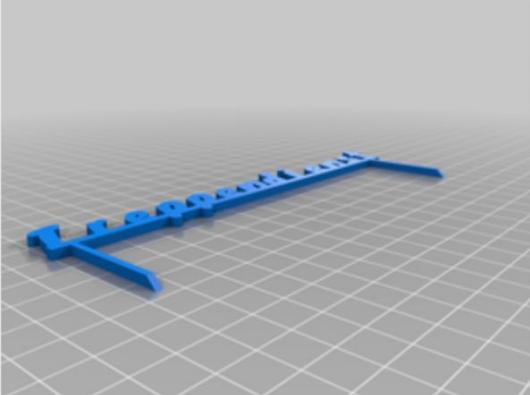
My Customized prueba ...

Jul 05, 2020

[+ Collect Thing](#)

My Customized Garde...

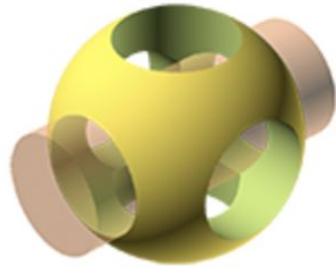
Jul 02, 2020

[+ Collect Thing](#)

DILL Garden Sign

Jun 28, 2020

[Back to Top](#)



OpenSCAD

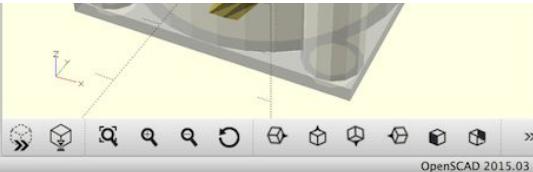
The Programmers Solid 3D CAD Modeller

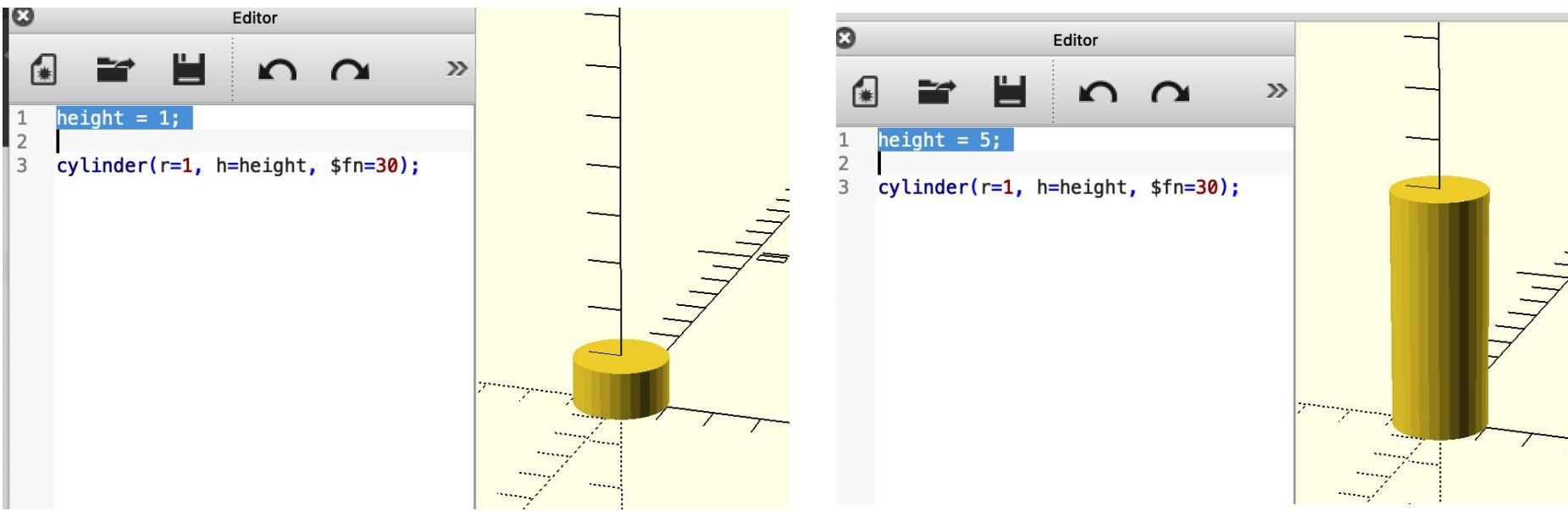


Totally parametric!!!

```
1 // intersection();
2
3 linear_extrude(height = fanwidth, center = true, convexity = 10, twist = -fanrot)
4   import(file = "example009.dxf", layer = "fan_top");
5
6
7 // NB! We have to use the deprecated module here since the "fan_side"
8 // layer contains an open polyline, which is not yet supported
9 // by the import() module.
10 rotate_extrude(file = "example009.dxf", layer = "fan_side",
11   origin = fan_side_center, convexity = 10);
12
13 }
```

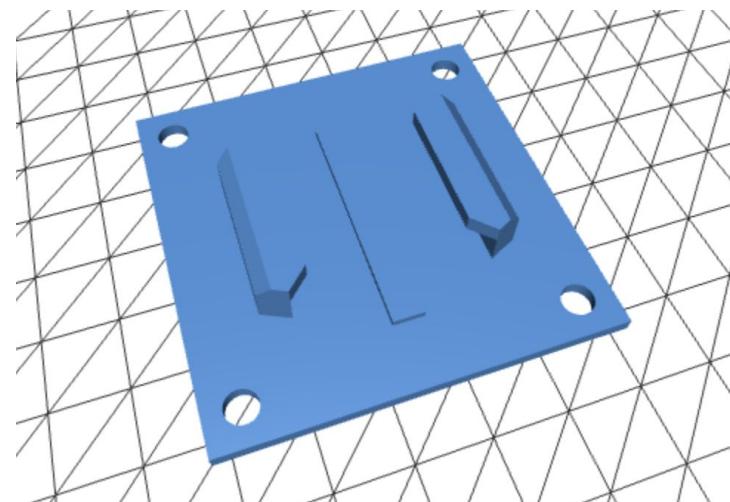
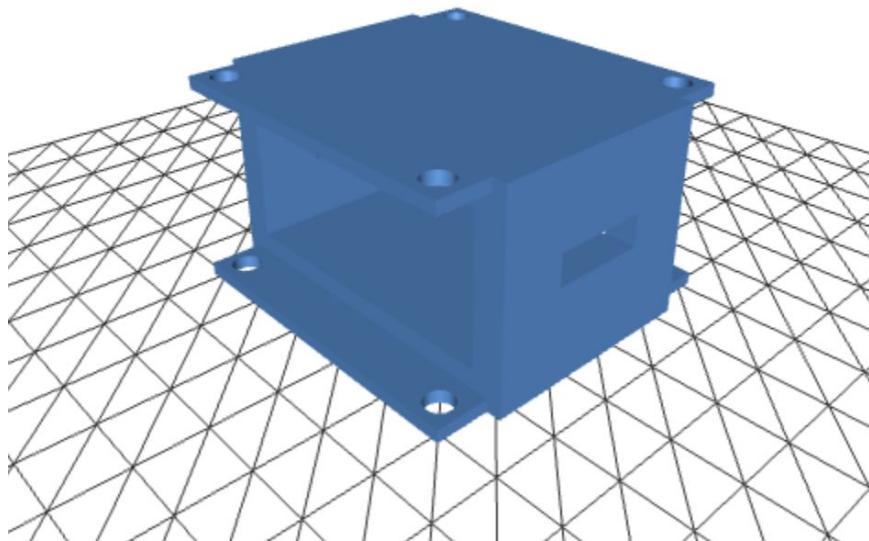
Viewport: translate = [0.61 -1.31 -2.07], rotate = [55.00 0.00 25.00], distance = 142.23





**(some) stuff i've made with
OpenSCAD**

drone gopro mount and battery housing



architectural models

Providence, R. I.
March 19, 1916

Special Features

The Providence Sunday Journal

Fifth Section

A STRIKING PLAN FOR DIGNIFYING CIVIC CENTRE

Design Copyright, 1916, by Raymond M. Hood.

The plan shows a large, rectangular civic center bounded by Exchange Place to the south, State Street to the west, and Post Office Square to the north. The design includes a central tower and various municipal buildings. Labels indicate "COURT HOUSE", "CITY HALL", "POST OFFICE", and "LIBRARY". A circular inset shows a typical floor plan of one of the buildings.

The Main Entrance to the Court House.

Typical Floor Plan of the Tower.

Former Rhode Islander Suggests Imposing State and Municipal Group. With Tower, to Occupy Entire Square South of Exchange Place

*S*TARTING with the proposed addition to the City Hall, Raymond M. Hood, a New York architect, has suggested a plan for the development of certain civic structures in Providence which would result in the erection of a group of buildings of great architectural value and interest.

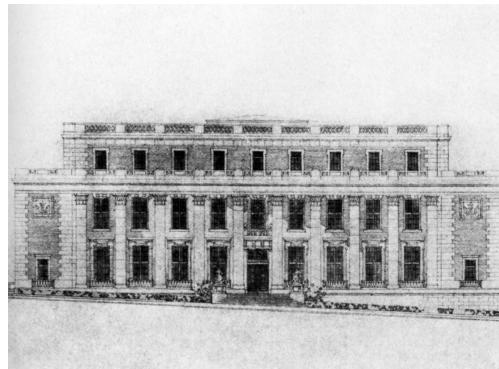
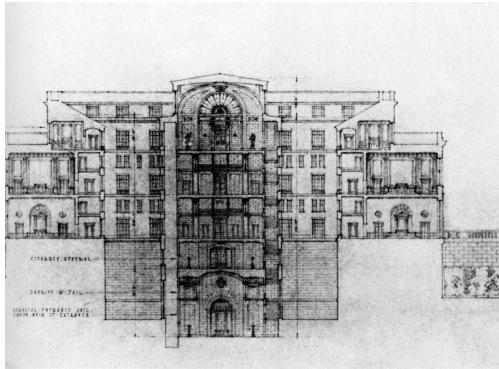
The plan, which has been submitted to the Providence City Council, is based upon a suggestion made by a former resident of Providence, George E. Nichols, who recently visited the city. Nichols' suggestion was that the State and Municipal Government should occupy the entire square bounded by Exchange Place, State Street, and Post Office Square, and that the City Hall be moved from its present location on State Street to the center of the proposed group of buildings.

Hood's plan provides for the erection of a large tower, which will be the dominant feature of the proposed group of buildings. The tower will be located in the center of the proposed group of buildings. The tower will be the dominant feature of the proposed group of buildings. The tower will be the dominant feature of the proposed group of buildings. The tower will be the dominant feature of the proposed group of buildings.

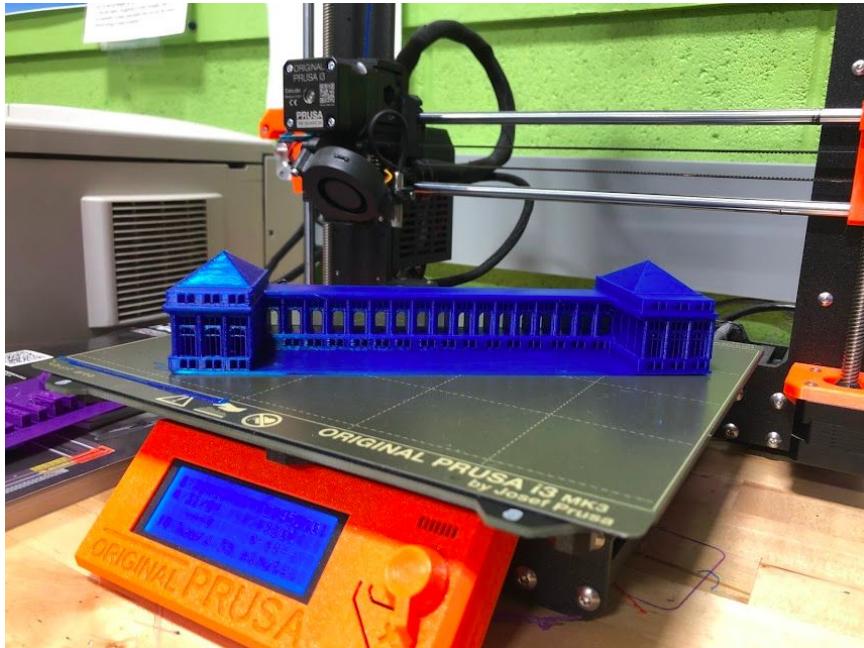
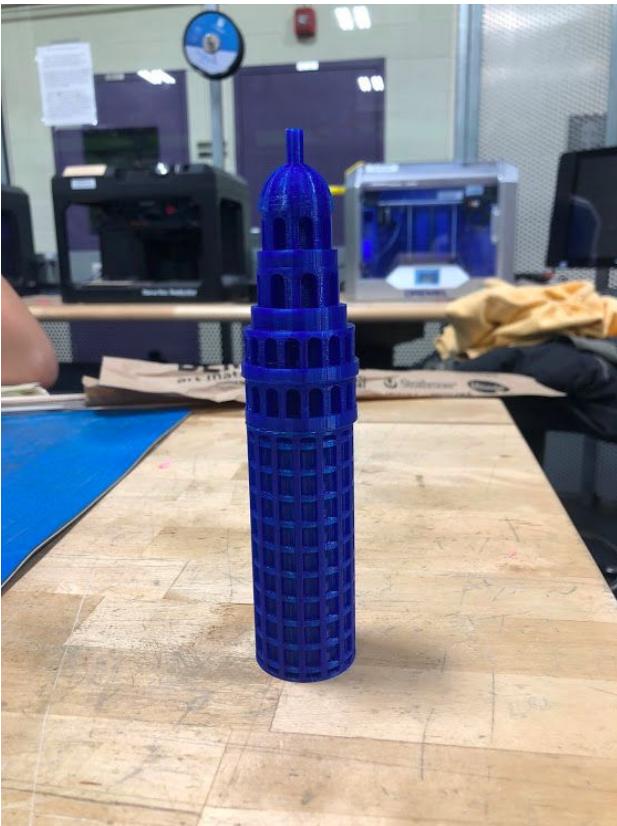
View of the Proposed Group from the North

Plan Showing Relation of Suggested Buildings to the City Hall and Post Office.

Elevation of Court House, Federal and Municipal Buildings.

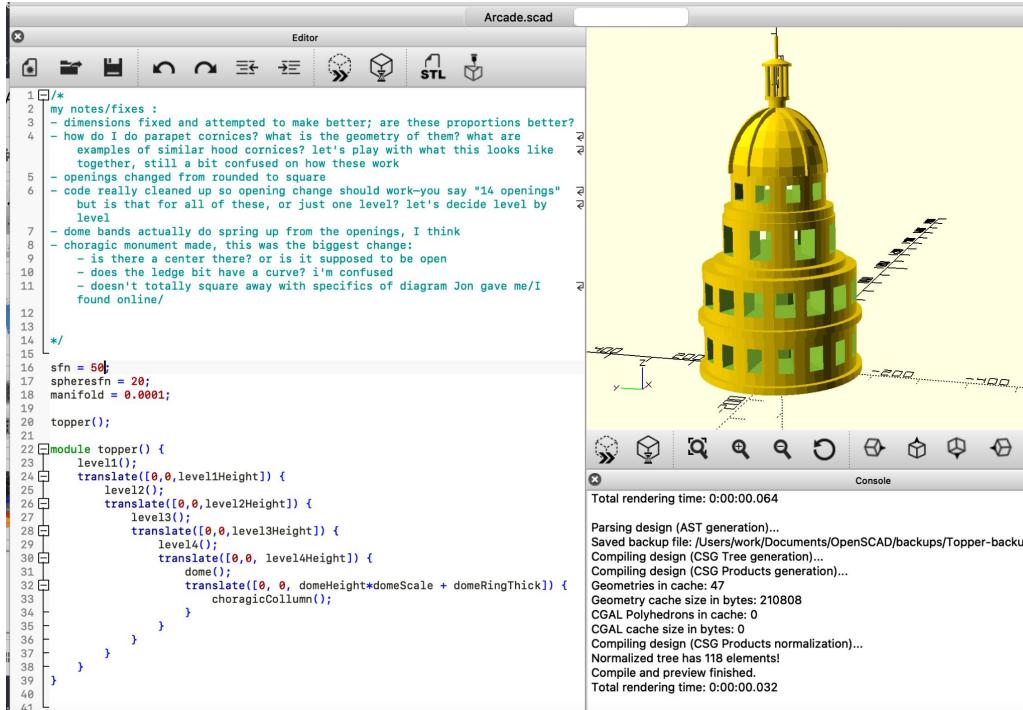


architectural models



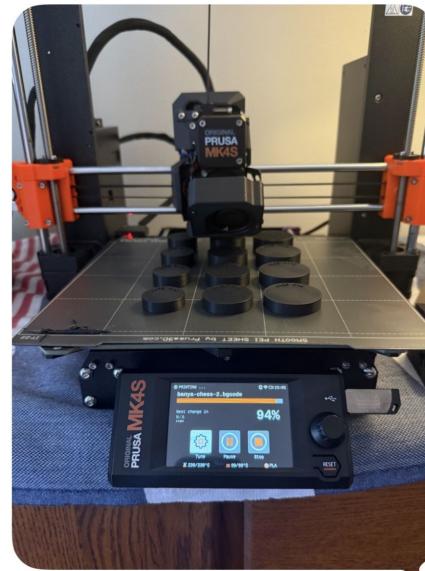
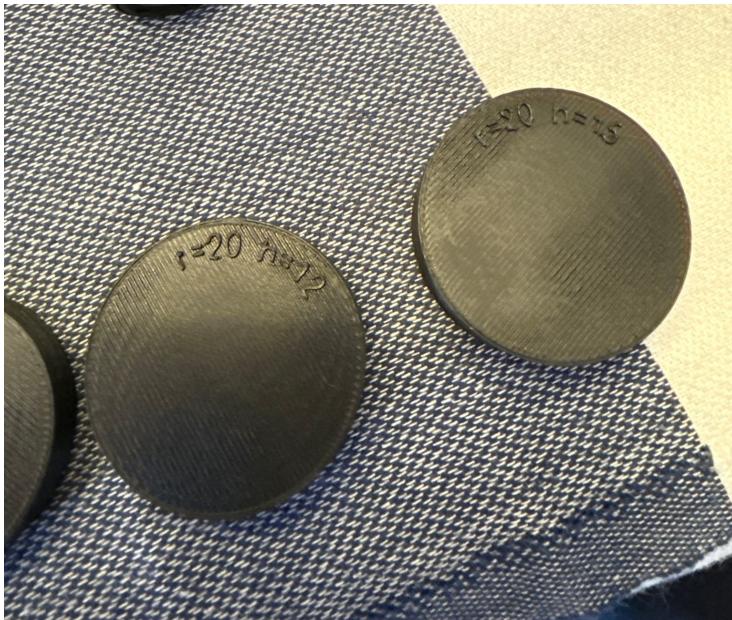
architectural models

<https://github.com/lucasgelfond/Providence-City-Center-Model>



Monday 3:06 PM

replacement banya chess pieces



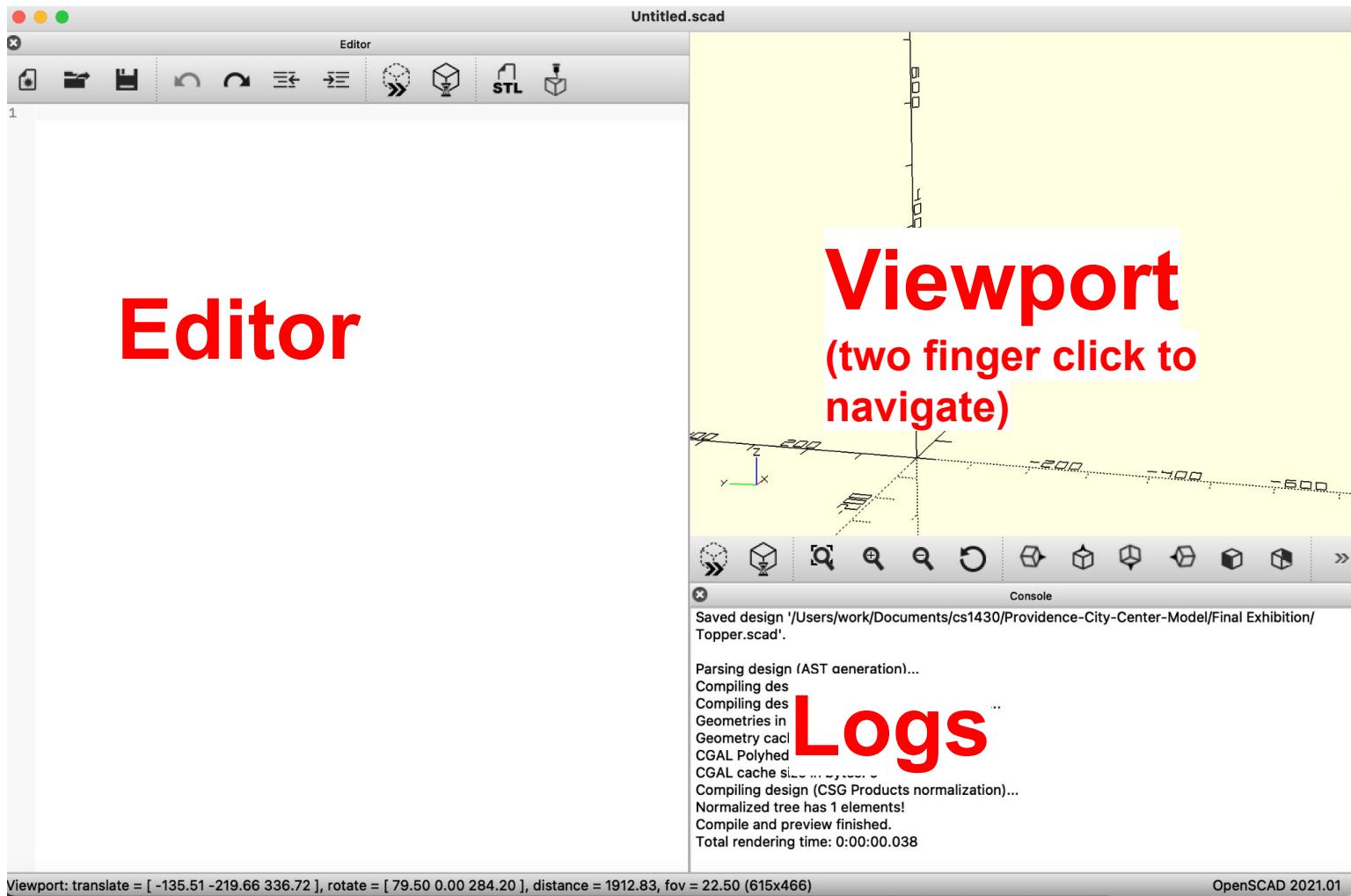
printed 12 cylinders of varying widths and height to bring to banya for chess piece replacement



labeled with radius and height so I can tell

interactive part

Editor



Viewport (two finger click to navigate)

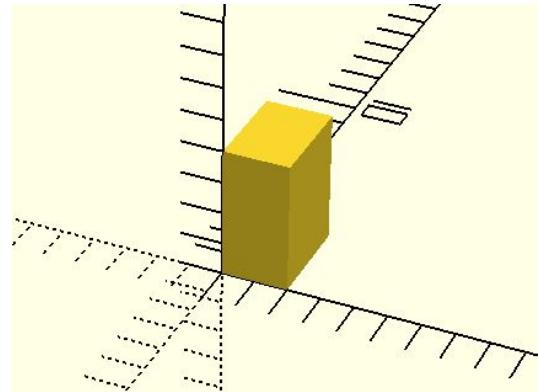
Logs

information dump / crash course

(and reminder to self to not speed through this)

```
cube([2, 3, 4]);
```

F5 to render



click  to Reset View if you can't see

cylinder($r = 3$, $h = 5$);

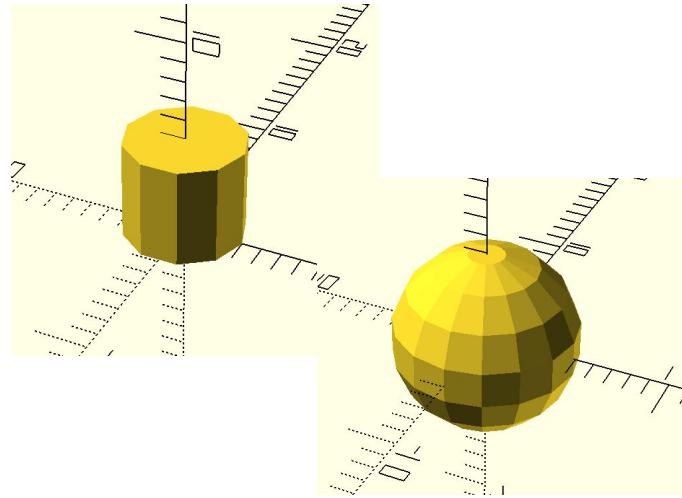
sphere($r = 5$)

object primitives:

cube([width, depth, height]);

cylinder($r = \text{radius}$, $h = \text{height}$)

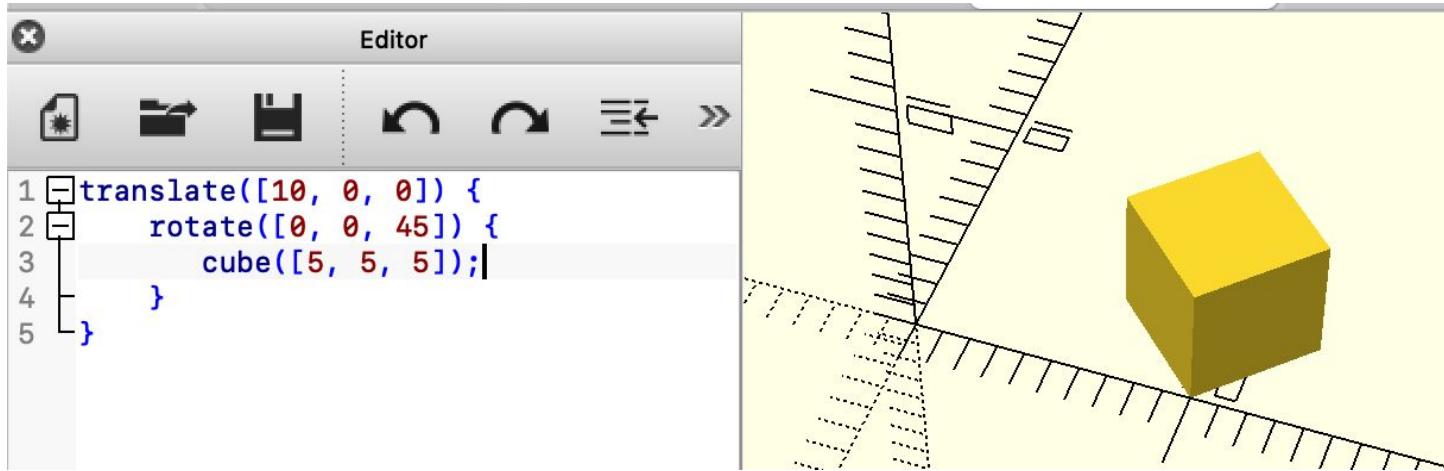
sphere($r = \text{radius}$)



transforms

- `translate([x, y, z]) { }`
- `rotate([x, y, z]) { } (degrees)`
- `scale ([x, y, z]) { } (default: [1, 1, 1])`

transforms can be nested!

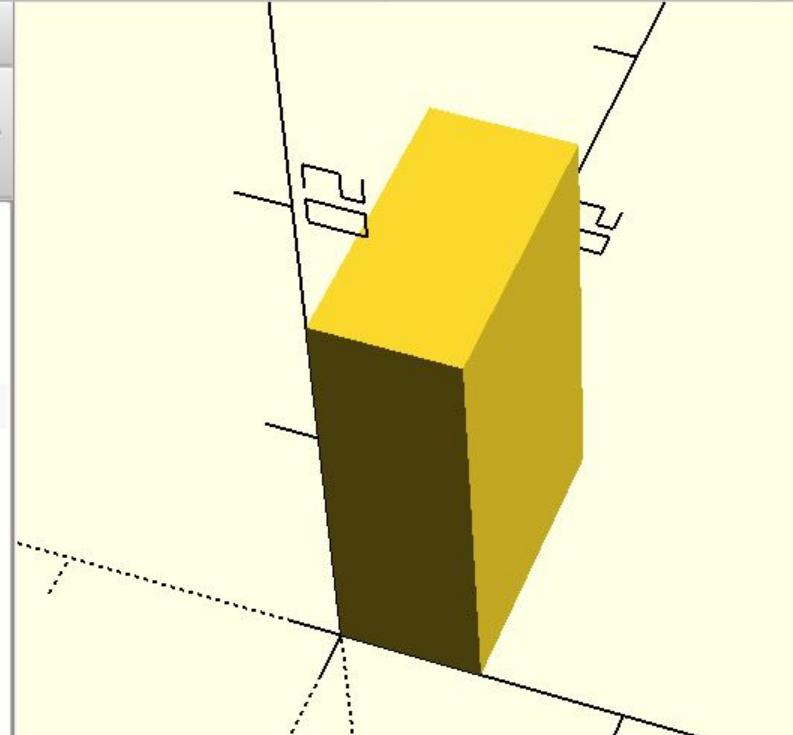


This would

- *create a 5x5x5 cube*
- *rotate it 45 degrees about the Z axis*
- *translate it 10 units across the X axis*
- NOTE: would get a different result if we did in a different order!

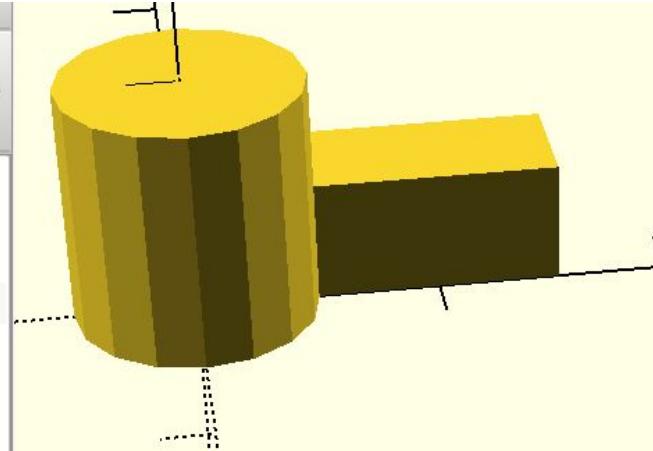
variables never change and need no declaration

Editor



```
1 boxWidth = 5;
2 boxDepth = 10;
3 boxHeight = 15;
4
5 cube([boxWidth, boxDepth, boxHeight]);|
```

union()



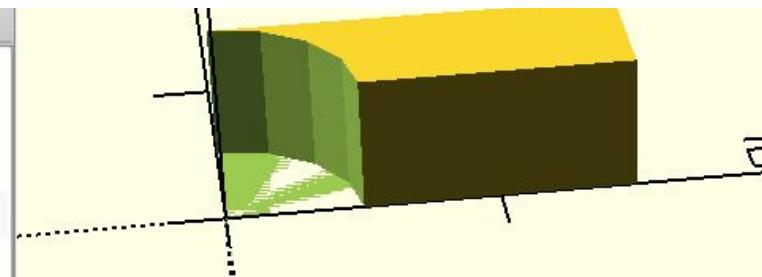
The image shows a 3D rendering of a yellow cylinder and a black cube joined together at their top edges, demonstrating the union() function.

The code in the editor window is:

```
1 union() {
2     cube([15, 5, 5]);
3     cylinder(r = 5, h = 10);
4 }
```

difference()

```
1 difference() {  
2     cube([15, 5, 5]);  
3     cylinder(r = 5, h = 10);  
4 }
```



intersection()



The image shows a 3D modeling application's interface. The top bar contains standard file and tool icons. Below the toolbar is a code editor window displaying the following script:

```
1 intersection() {
2     cube([15, 5, 5]);
3     cylinder(r = 5, h = 10);
4 }
```

The preview window on the right shows a 3D scene with a large gray cube and a smaller gray cylinder intersecting it. A yellow highlight indicates the intersection volume.

“debugging” with % and

The image shows a CAD software interface with a code editor on the left and a 3D preview on the right.

Code Editor (Left):

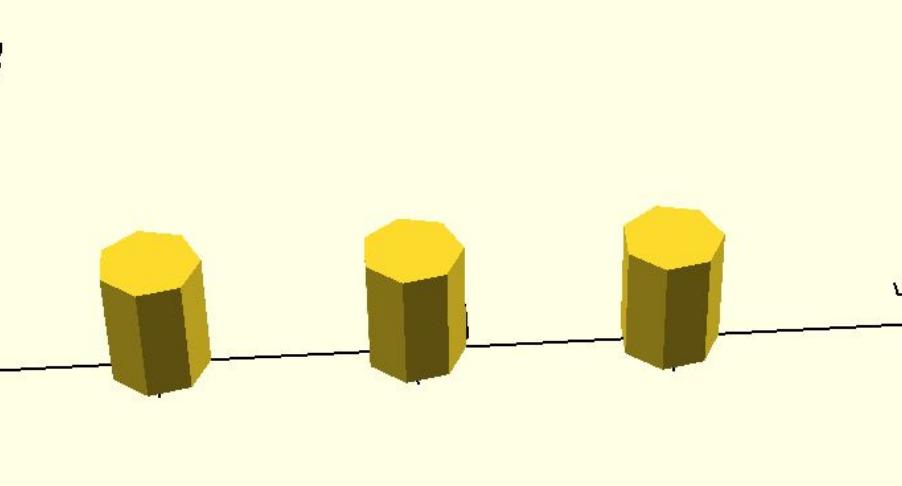
```
1 difference() {  
2     #cube([15, 5, 5]);  
3     %cylinder(r = 5, h = 10);  
4 }
```

3D Preview (Right):

The 3D preview shows a yellow cube and a brown cylinder. The cylinder is positioned such that it appears to be inside or intersecting the cube. The background is a light yellow color, and there are coordinate axes and a scale bar labeled "20" visible.

for() and repetition

EDITOR



```
1 for(i = [1: 3]) {
2     translate([10 * i, 0, 0]) {
3         cylinder(r = 2, h = 5);
4     }
5 }
```

miscellany: \$fn

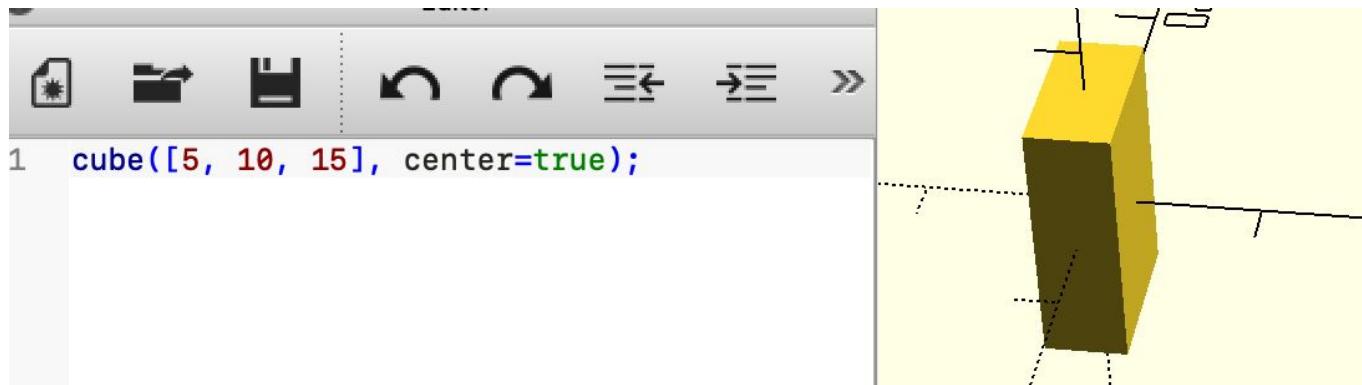
The image displays three vertically stacked screenshots of a 3D modeling application's interface. Each screenshot shows a 3D view of a yellow cylinder on a coordinate system with dashed axes.

- Screenshot 1:** Shows a cylinder with 30 facets. The facets are very thin and numerous, giving it a smooth, rounded appearance.
- Screenshot 2:** Shows a cylinder with 10 facets. The facets are thicker and fewer in number, resulting in a more polygonal and less smooth appearance.
- Screenshot 3:** Shows a cylinder with 3 facets. The facets are very thick and large, giving it a very angular and blocky appearance.

Each screenshot includes a toolbar at the top with icons for file operations (New, Open, Save, etc.) and navigation (Back, Forward, Home, Zoom). Below the toolbar, the code for creating the cylinder is displayed in a text editor:

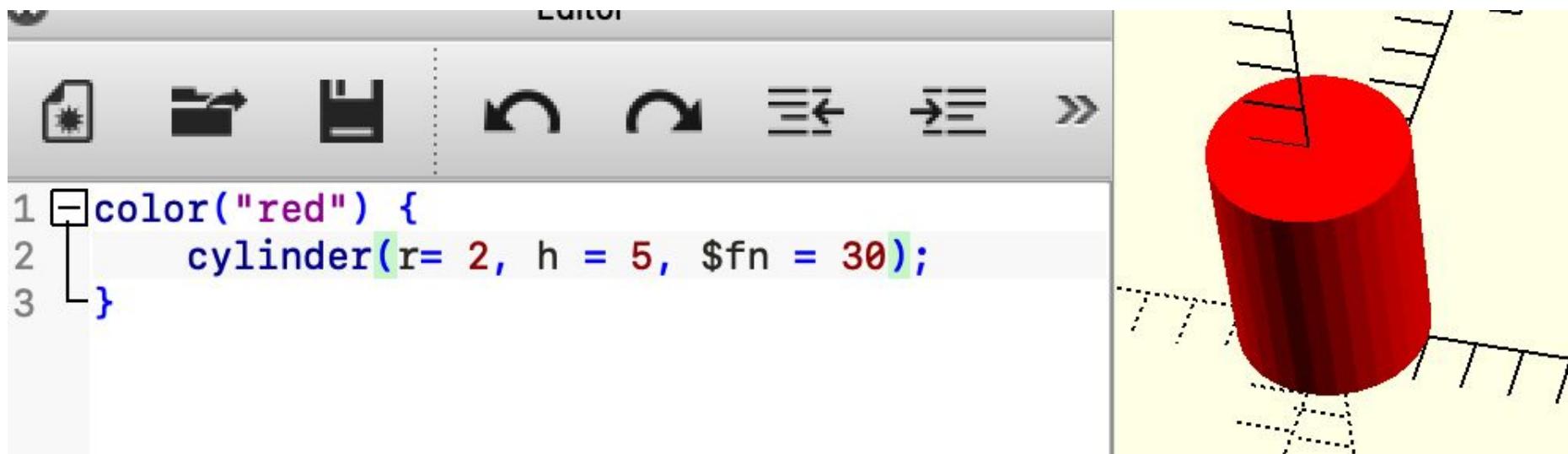
- Screenshot 1: `cylinder(r = 2, h = 2, $fn = 30);`
- Screenshot 2: `cylinder(r = 2, h = 2, $fn = 10);`
- Screenshot 3: `cylinder(r = 2, h = 2, $fn = 3);`

miscellany: center=true



miscellany: color("colorname")

Editor



```
1 color("red") {  
2     cylinder(r= 2, h = 5, $fn = 30);  
3 }
```

fun bonus: minkowski()

About 1,220,000 results (0.43 seconds)

<https://www.wikiwand.com> › Minkowski_addition ...

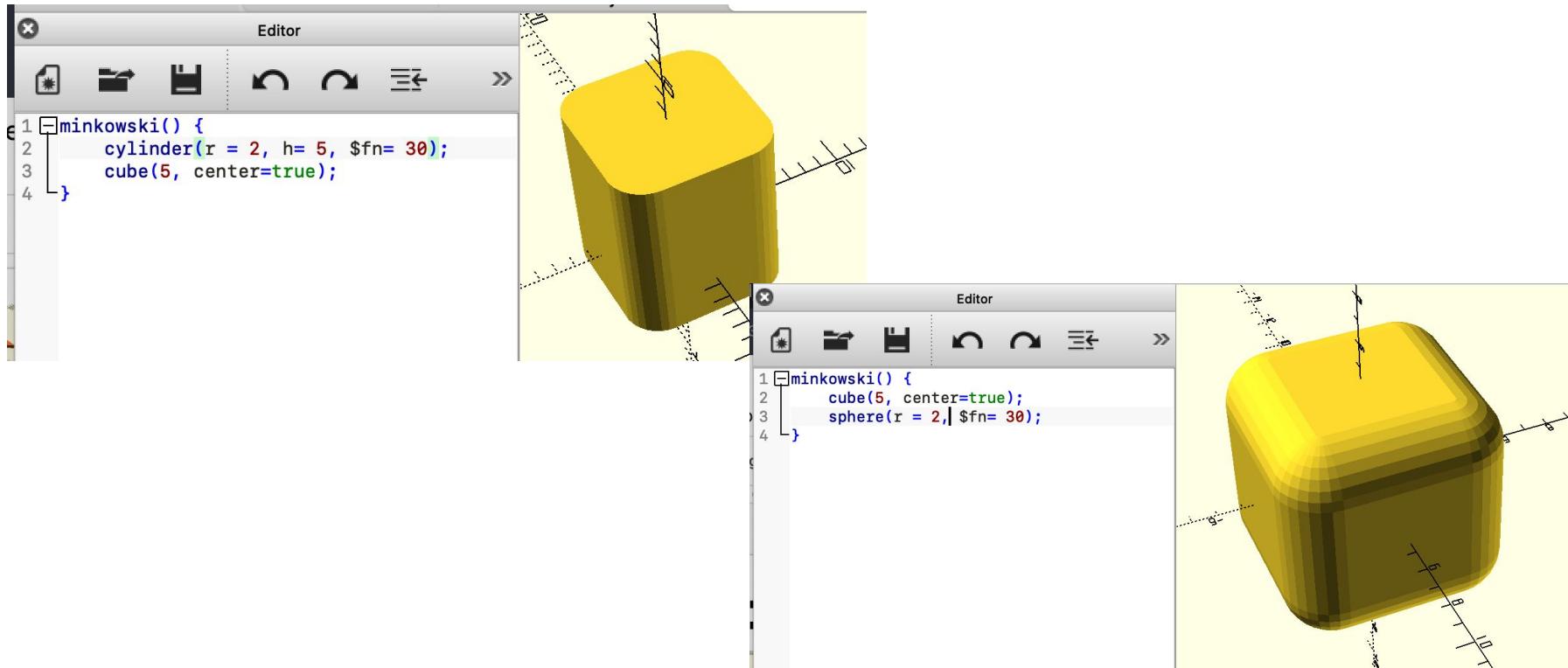
Minkowski addition - Wikiwand

In geometry, the **Minkowski sum** (also known as dilation) of two sets of position vectors A and B in Euclidean space is formed by adding each vector in A to each ...

Example · Convex hulls of Minkowski sums · Algorithms for computing...

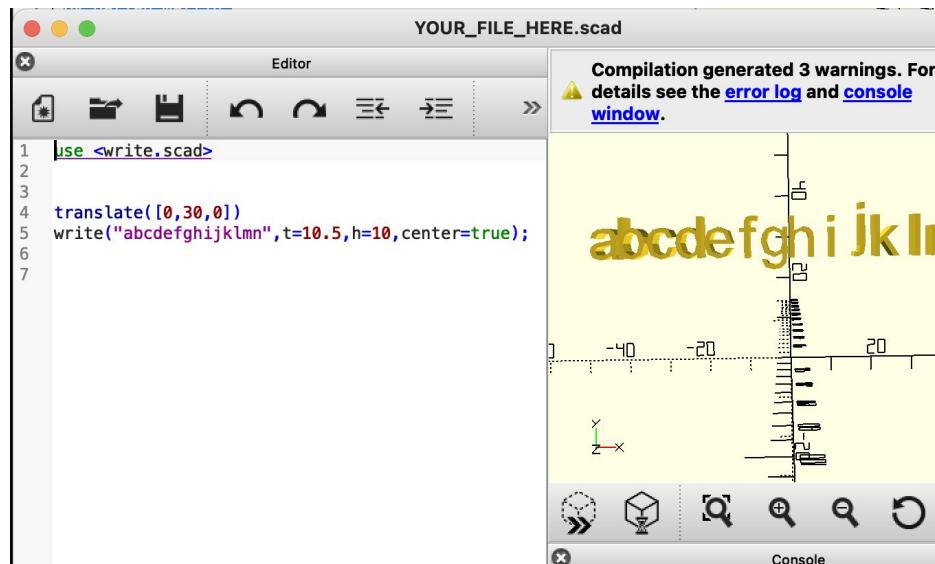


fun bonus: minkowski()



fun bonus: Write.scad

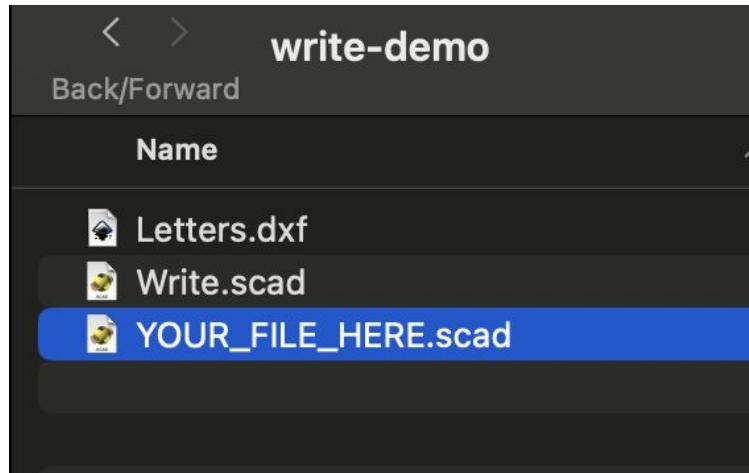
this setup is annoying; there's a ZIP in Slack!



A screenshot of a CAD software interface. The title bar says "YOUR_FILE_HERE.scad". The toolbar includes standard file operations like Open, Save, and Print. Below the toolbar is a code editor window containing the following SCAD code:

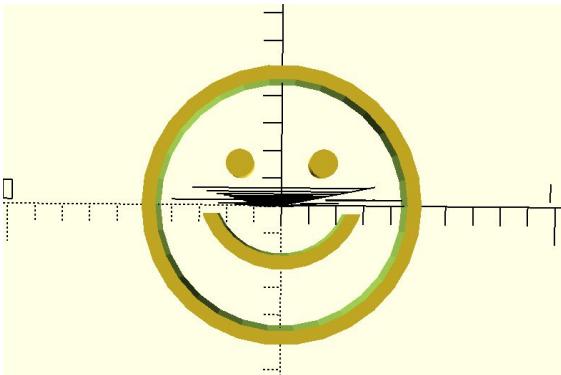
```
1 use <write.scad>
2
3
4 translate([0,30,0])
5 write("abcdefghijklmn", t=10.5, h=10, center=true);
6
7
```

The main workspace shows a 3D model of the letters "a" through "l" in a perspective view. A warning message is displayed at the top right: "Compilation generated 3 warnings. For details see the [error log](#) and [console window](#)". The bottom of the screen features a "Console" tab.

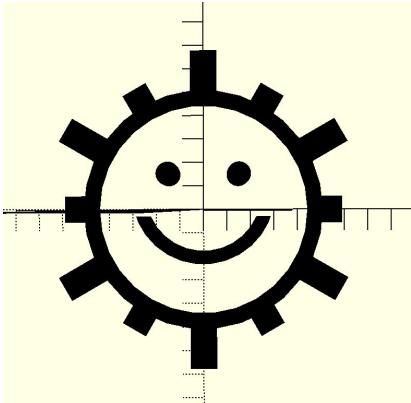


Your Challenge(s)

v1



v2



v3



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

OpenSCAD v2021.01

Syntax

```
var = value;  
var = cond ? value_if_true : value_if_false;  
var = function (x) x + x;  
module name(...) { ... }  
name();  
function name(...) = ...  
name();  
include <...>.scad>  
use <...>.scad>
```

Modifier Characters

*	disable
_	show only
#	highlight / debug
%	transparent / background

2D

```
circle(radius | d=diameter)  
square(size,center)  
square([width,height],center)  
polygon([points])  
polygon([points],[paths])  
text(t, size, font,  
     halign, valign, spacing,  
     direction, language, script)  
import("...ext", convavity)  
projection(cut)
```

Constants

```
undef undefined value  
PI mathematical constant π (~3.14159)
```

Operators

```
n + m Addition  
n - m Subtraction  
n * m Multiplication  
n / m Division  
n % m Modulo
```

3D

```
sphere(radius | d=diameter)  
cube(size, center)
```

(spoiler ahead!)

smiley-1.scad

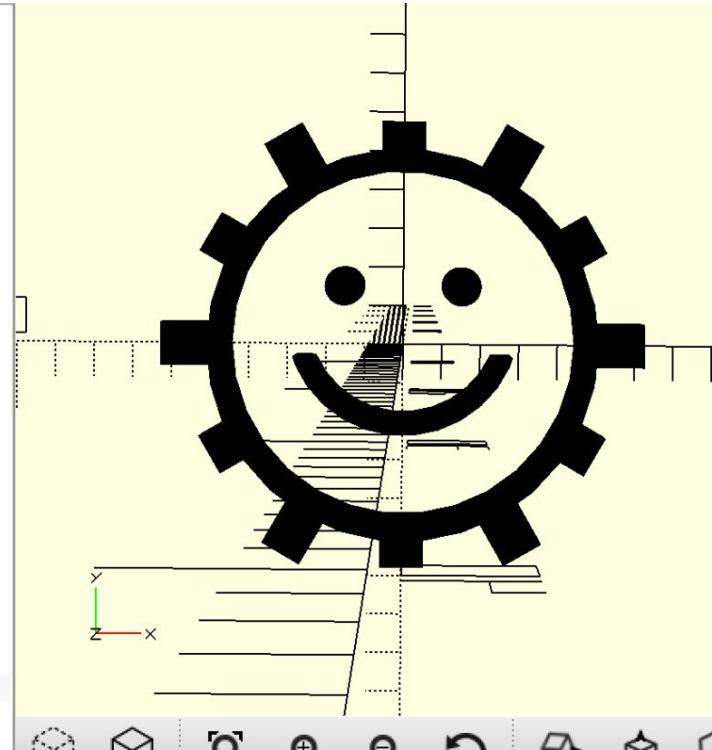
Editor

```
1 sfn=30;
2
3
4
5
6 difference() {
7     cylinder(r=5, h=2, $fn=sfn, center=true);
8     cylinder(r=4.5, h=5, $fn=sfn, center=true);
9 }
10
11 for(i = [-1, 1]) {
12     translate([1.5*i,1.5,0])cylinder(r=0.5,h=2
13     , $fn=sfn, center=true);
14 }
15
16 translate([0,0.7,0]) difference() {
17     cylinder(r=3, h=2, $fn=sfn, center=true);
18     cylinder(r=2.5, h=5, $fn=sfn, center=true);
19     translate([0,4,0]) cube([10,10,5], center=
20     true);
```

Normalized tree has 7 elements!
Compile and preview finished.

smiley-2.scad

```
1  sfn=30;  
2  
3  color("black", 1.0) {  
4      difference() {  
5          cylinder(r=5, h=2, $fn=sfn, center=true);  
6          cylinder(r=4.5, h=5, $fn=sfn, center=true);  
7      }  
8  
9      for(i = [-1, 1]) {  
0          translate([1.5*i,1.5,0])cylinder(r=0.5,h=2, $fn=sfn, center=true);  
1      }  
2  
3  
4      translate([0,0.7,0]) difference() {  
5          cylinder(r=3, h=2, $fn=sfn, center=true);  
6          cylinder(r=2.5, h=5, $fn=sfn, center=true);  
7          translate([0,4,0]) cube([10,10,5], center=true);  
8      }  
9      for(i=[0:6]) {  
0          rotate([0,0,i*60 + 30])  
1              translate([0,5.35, 0])  
2                  cube([1.12, 1.75, 1], center=true);  
3          rotate([0,0,i*60])  
4              translate([0,5.1,0])  
5                  cube([1.12, 1.25, 1], center=true);  
6      }  
7  }
```





thanks!

(310) 989-8334

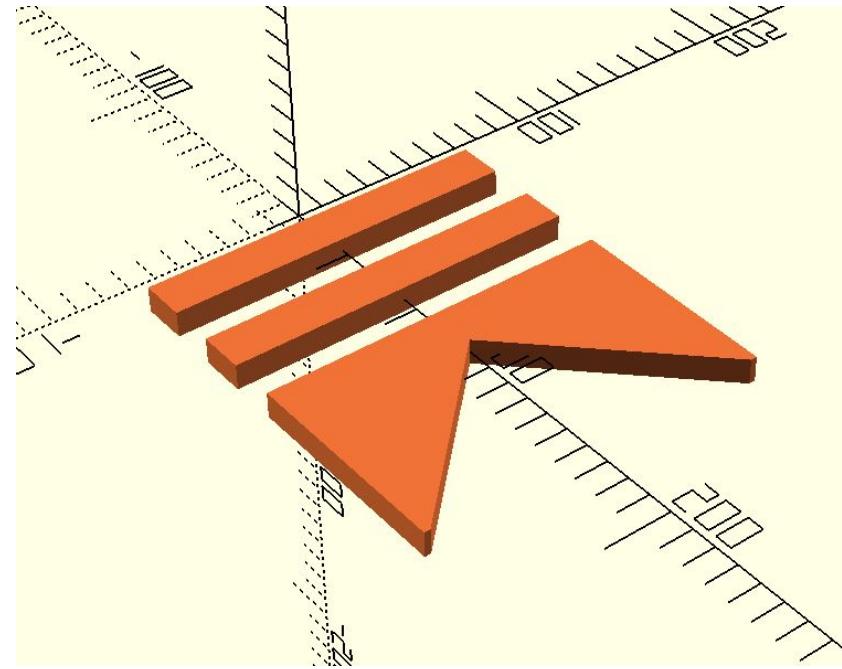
lucasgelfond@gmail.com

<http://lucasgelfond.online>

twitter/instagram: @gucasgelfond

slides from when i did this as a lunch 'n learn

Your Challenge



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

OpenSCAD v2021.01

Syntax

```
var = value;
var = cond _ value_if_true ± value_if_false;
var = function (x) x + x;
module name(...) { ... }
name();
function name(...) = ...
name();
include <...>.scad>
use <..>.scad>
```

Modifier Characters

<u>*</u>	disable
<u>!</u>	show only
<u>#</u>	highlight / debug
<u>%</u>	transparent / background

2D

```
circle(radius | d=diameter)
square(size,center)
square([width,height],center)
polygon([points])
polygon([points],[paths])
text(t, size, font,
      halign, valign, spacing,
      direction, language, script)
import("...ext", convexity)
projection(cut)
```

Constants

```
undefined undefined value
PI mathematical constant π (~3.14159)
```

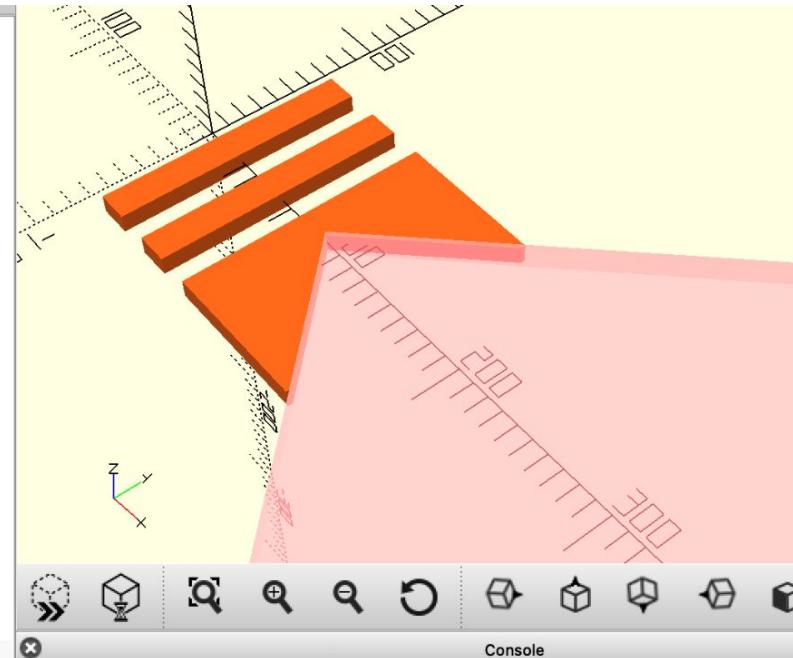
Operators

<u>n + m</u>	Addition
<u>n - m</u>	Subtraction
<u>n * m</u>	Multiplication
<u>n / m</u>	Division
<u>n % m</u>	Modulo

3D

```
sphere(radius | d=diameter)
cube(size, center)
```

```
1  logoWidth = 136;
2  barThickness = 18;
3  depth = 10;
4  bottomLength = 84;
5
6 color("#ed7135") difference() {
7   union() { // group first before subtracting stuff
8     //the two top things
9     for(i = [1, 3]) { //renders two; one at i = 3, one at i = 3
10      translate([ barThickness*i, 0, 0]) {
11        cube([barThickness, logoWidth, depth], center = true); // bars
12      }
13    }
14   translate([barThickness*4.5 + bottomLength/2, 0, 0]) {
15     // bottom thick rectangle we cut the notch out of
16     cube([bottomLength,logoWidth, depth], center=true);
17   }
18 }
19 // this is what we are cutting out of the top
20 translate([barThickness*4.5 + bottomLength*1.7 + logoWidth/2, 0,0]) {
21   rotate([0,0,45]) { // rotate 45 so we get the corner
22     // this is the notch
23     #cube([logoWidth*2, logoWidth*2, depth+5], center=true);
24   }
25 }
26 }
27 }
```



[BACK TO CUSTOMIZABLE GARDEN SIGN](#)

Now Using: Customizer

Queue

▼ let-s-grow-something-awesome

🔗 <https://www.thingiverse.com/ap>

Copy

[View Source](#)

[Create Thing](#)

Label Text What are we planting?

Substack

Font

Letters



Build Plate Selector For display only, not part of final model

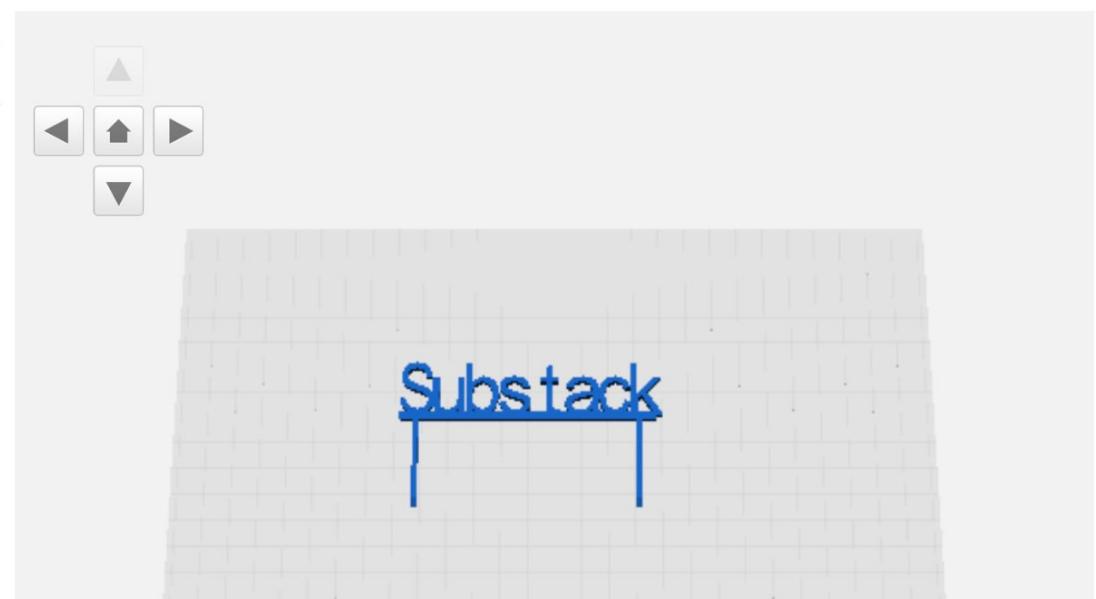
Replicator 2/2X

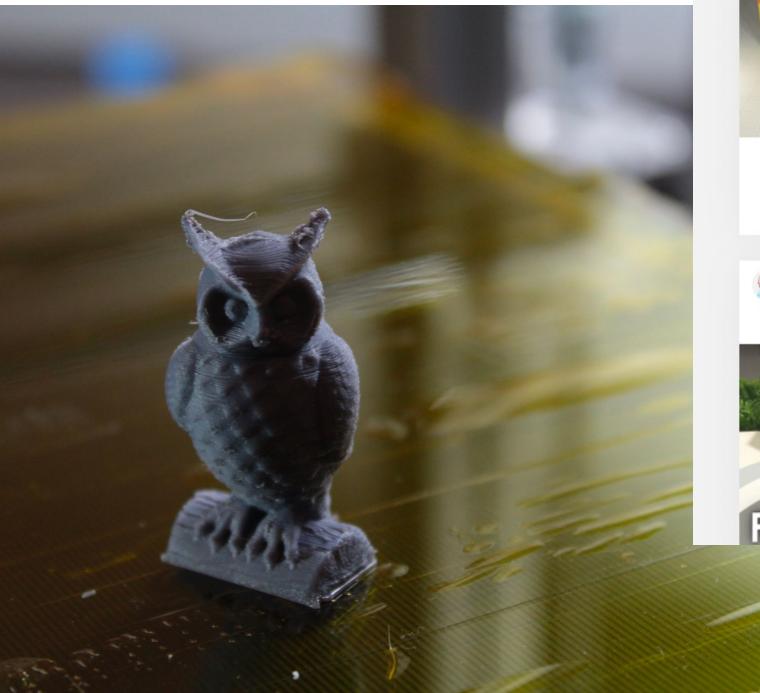
Build Plate Manual X "Manual" build plate X dimension

10

Build Plate Manual Y "Manual" build plate Y dimension

10





MakerBot Thingiverse

Popular Last 30 Days

All Things

Filter By

Stackable Storage Caddies

No supports required! Fully Customizable

Collect Thing Share

Goooose

Collect Thing Share

Magazine Battery holder (AAA)

Collect Thing Share

Filament Machine II - Petamentor2

FILAMENT MACHINE II

Vase #650

Playing Card Deck Holder

Explore Education Create +

Search Thingiverse