OpenSCAD{

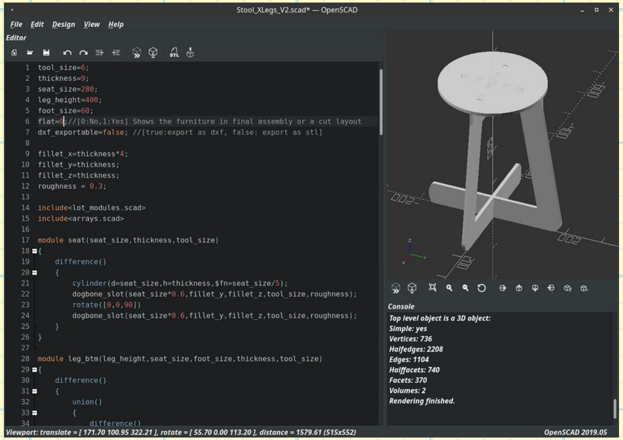
3D Modeling(using Code);

}

Tutor: Jens Meisner

Computational Artist and Media Designer

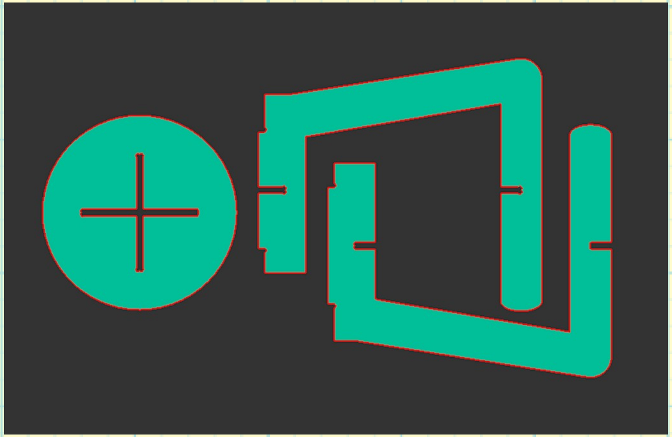
OpenSCAD - An Introduction



```
tool_size=6;
thickness=9;
seat_size=280;
leg_height=400;
foot_size=60;
flat=0;//[0:No,1:Yes] Shows the furniture in final assembly or a cut layout
dxf_exportable=false; //[true:export as dxf, false: export as stl]

fillet_x=thickness*4;
fillet_y=thickness;
fillet_z=thickness;
roughness = 0.3;

include<lot_modules.scad>
include<arrays.scad>

module seat(seat_size,thickness,tool_size)
{
    difference()
    {
        cylinder(d=seat_size,h=thickness,$fn=seat_size/5);
        dogbone_slot(seat_size*0.6,fillet_y,fillet_z,tool_size,roughness);
        rotate([0,0,90])
        dogbone_slot(seat_size*0.6,fillet_y,fillet_z,tool_size,roughness);
    }
}

module leg_btm(leg_height,seat_size,foot_size,thickness,tool_size)
{
    difference()
    {
        union()
        {
            difference()
```

```
Top level object is a 3D object:
Simple: yes
Vertices: 736
Halfedges: 2208
Edges: 1104
Halffacets: 740
Facets: 370
Volumes: 2
Rendering finished.
```

```
Viewport: translate = [ 171.70 100.95 322.21 ], rotate = [ 55.70 0.00 113.20 ], distance = 1579.61 (515x552)
OpenSCAD 2019.05
```



Modeling >> Exporting >> Manufacturing

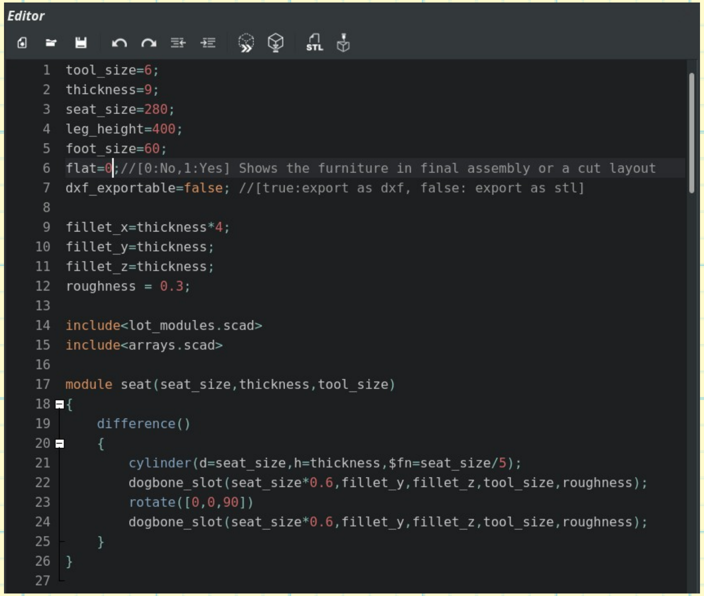
OpenSCAD - An Introduction

```
object();
   variable = value;
   operator() action();
   operator() { action();   action(); }
   operator() operator() { action();   action(); }
   operator() {operator()   action();
   operator() { action();   action(); } }
```

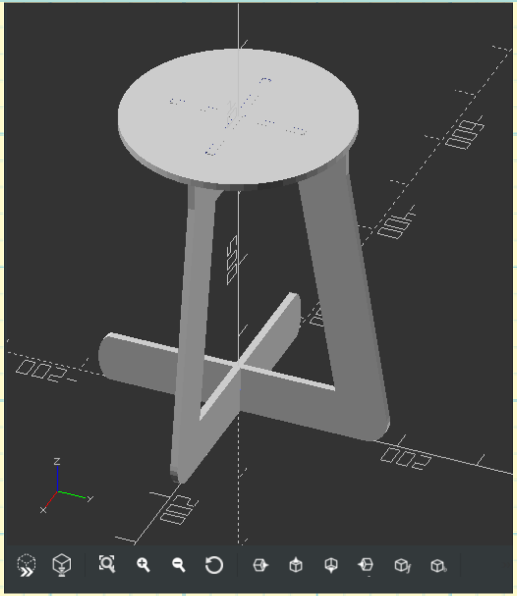
Examples of possible action statements

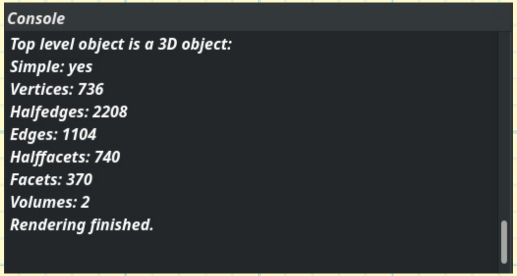
User Interface

OpenSCAD - An Introduction



```
tool_size=6;
thickness=9;
seat_size=280;
leg_height=400;
foot_size=60;
flat=0;//[0:No,1:Yes] Shows the furniture in final assembly or a cut layout
dxf_exportable=false; //[true:export as dxf, false: export as stl]

fillet_x=thickness*4;
fillet_y=thickness;
fillet_z=thickness;
roughness = 0.3;

include<lot_modules.scad>
include<arrays.scad>

module seat(seat_size,thickness,tool_size)
{
    difference()
    {
        cylinder(d=seat_size,h=thickness,$fn=seat_size/5);
        dogbone_slot(seat_size*0.6,fillet_y,fillet_z,tool_size,roughness);
        rotate([0,0,90])
        dogbone_slot(seat_size*0.6,fillet_y,fillet_z,tool_size,roughness);
    }
}

```





```
Top level object is a 3D object:
Simple: yes
Vertices: 736
Halfedges: 2208
Edges: 1104
Halffacets: 740
Facets: 370
Volumes: 2
Rendering finished.
```

Editor

Preview

Console

OpenSCAD has 3 main windows

Commenting Code

OpenSCAD - An Introduction

Examples of Commenting:

```
// Double slash will comment one line

/*
Slash and Asterisk will create a Multi-line comment,
while Asterisk and Slash will end the comment block
*/
```

Commenting is a very important part of coding

Boolean Operations

OpenSCAD - An Introduction

Boolean operations are:

```
    intersection(), difference(), union()
```

Example:

```
    intersection() {cube(30); sphere(30);}
```

There are 3 different Boolean operations

Transformations

OpenSCAD - An Introduction

Main transformations are:

translate(), rotate()

Further transformations available are:

scale(), color(), mirror(), multmatrix(),offset(), minkowski(), and hull()

There are 2 ways of navigating an object in space

OpenSCAD - An Introduction

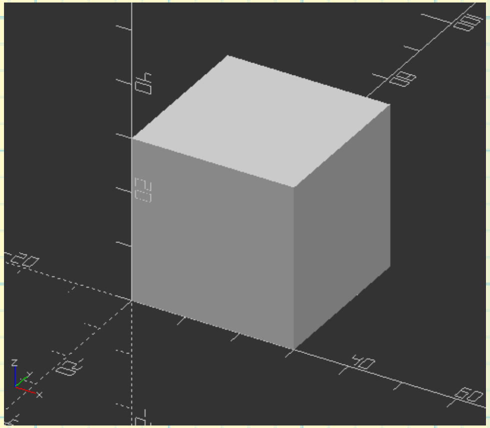
Examples:

```
//Numbers represent distance from offset in mm
translate([20,10,45]) sphere(20);
//Numbers represent degree from offset in °
rotate([55,25,45]) cube(30);
//Numbers represent multiples of original size
scale([-2,4,5]) cube(10);
```

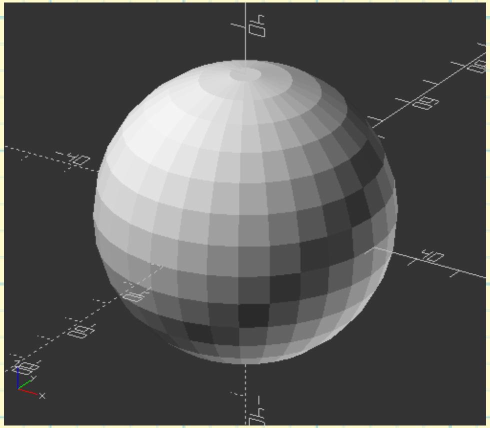
Vectors within transformations are embedded in []

3D Primitives

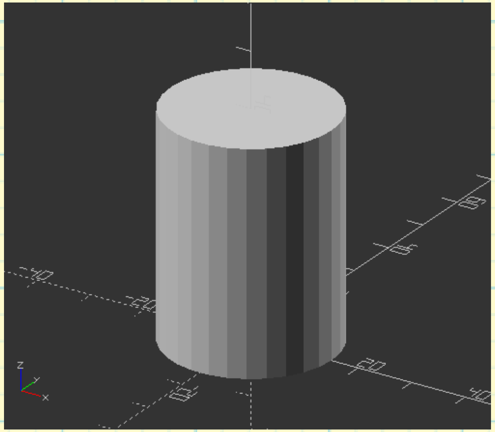
OpenSCAD - An Introduction



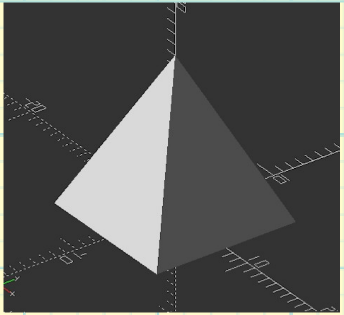
cube();

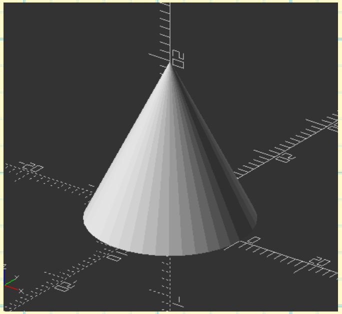


sphere();



cylinder();





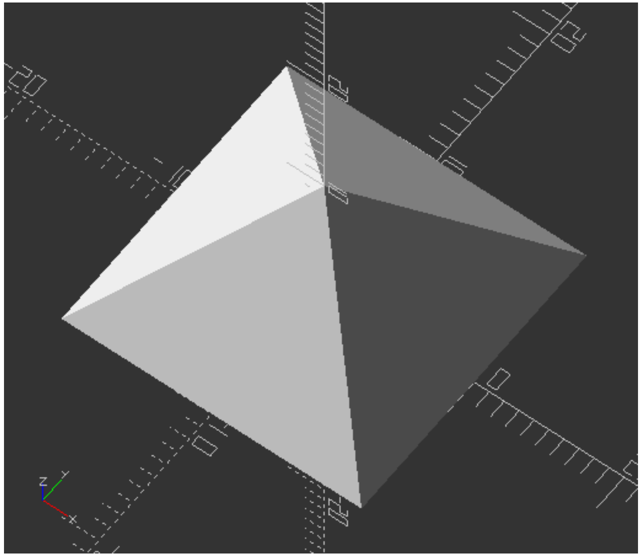
Basic 3D shapes: Cube, Sphere, Cylinder (Pyramid, Cone)

OpenSCAD - An Introduction

A Polyhedron is made of points (Vector of 3D points or vertices), and faces (Vector of faces that collectively enclose the solid)

Syntax:

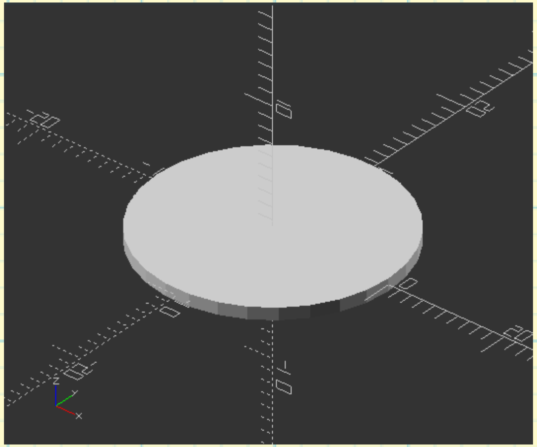
```
polyhedron(points=[],faces=[]);
```



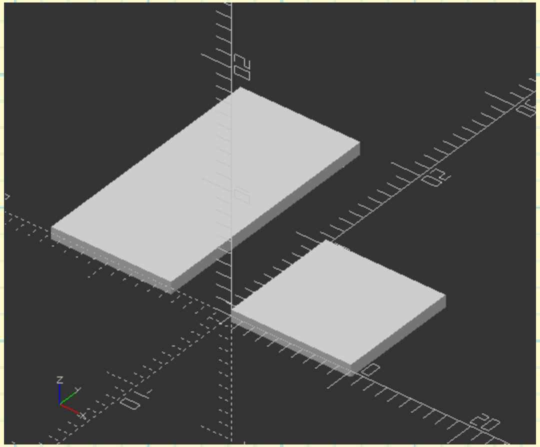
Pyramid made with polyhedron();

Most advanced 3D basic shape is the Polyhedron

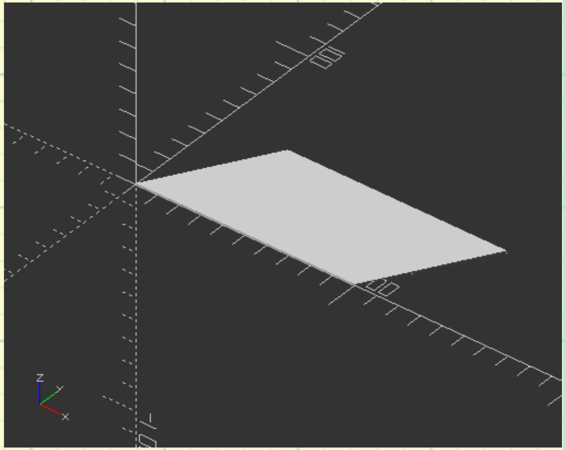
2D Primitives



circle();



square();

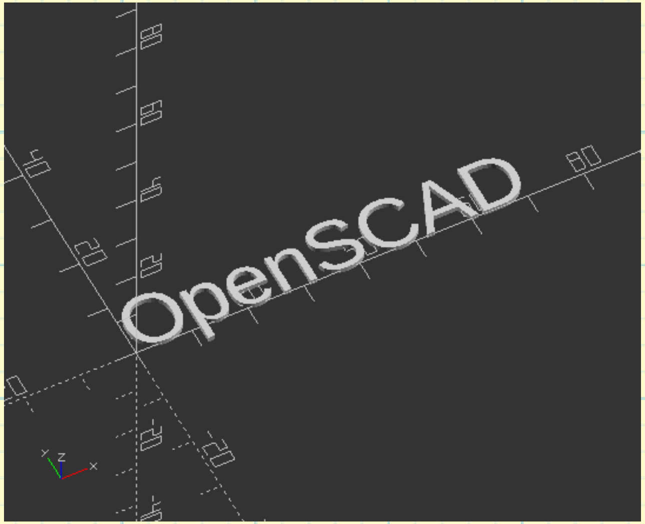


polygon();

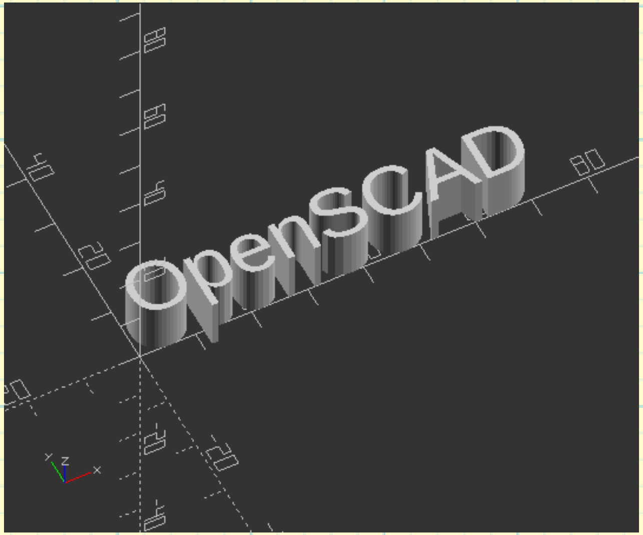
Basic 2D shapes: Circle, Square, Polygon

Text

OpenSCAD - An Introduction



```
text();
```



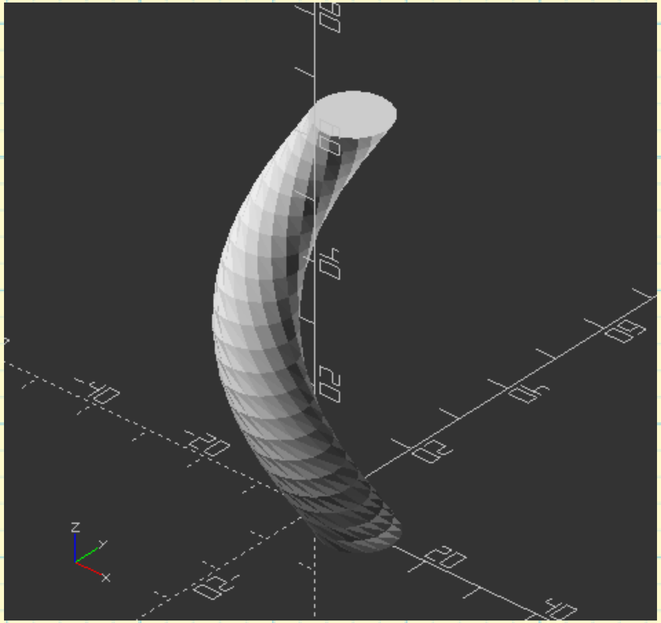
```
linear_extrude() text();
```

Text is a 2D object, that can be extruded

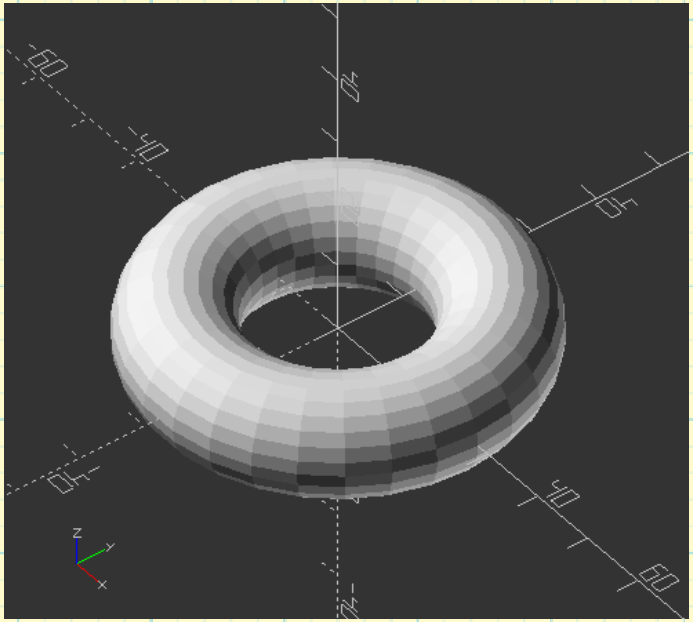
OpenSCAD - An Introduction

Extrusion

OpenSCAD - An Introduction



`linear_extrude()`



`rotate_extrude()`

Extrusion transforms 2D shapes into 3D

Projection

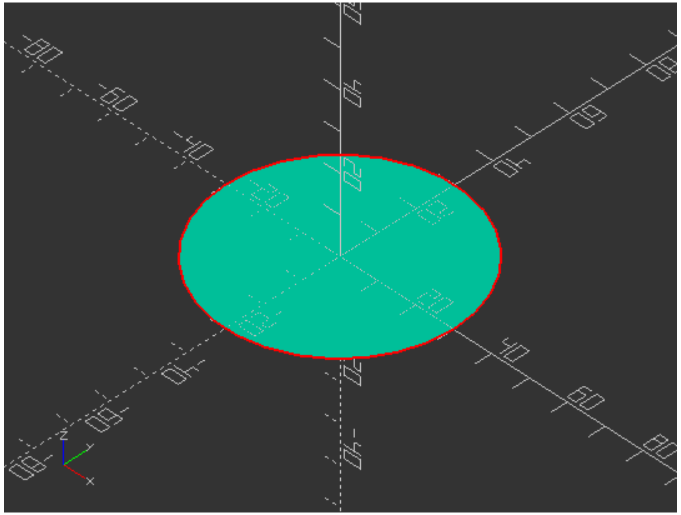
OpenSCAD - An Introduction

Projection is needed in order to export objects in DXF format.

The position of a cut can be set via parameter cut=true, and translate() or rotate()

Syntax:

```
projection() object();
```



Circle from sphere:

```
projection() sphere();
```

Projection() creates 2D drawings from 3D shapes

OpenSCAD - An Introduction

Export

OpenSCAD - An Introduction

In order to export an object is a full rendering via F6 necessary

- 3D Formats are STL, OFF, AMF, 3MF, or CSG
- 2D Formats are DXF, or SVG
- Image Format is PNG

If you want to export a 3D object in 2D, use projection()

Objects can be exported in 3D, 2D, or as an image

Import

OpenSCAD - An Introduction

- STL, OFF, AMF, 3MF are 3D Formats and DXF, SVG are 2D Formats you can import directly into OpenSCAD via the standard module import().

- surface() let you import a text file to be used as height map

- For an advanced version of import (for SCAD and CSG), use use<> and include<>

Objects only can be imported via code modules

Loops

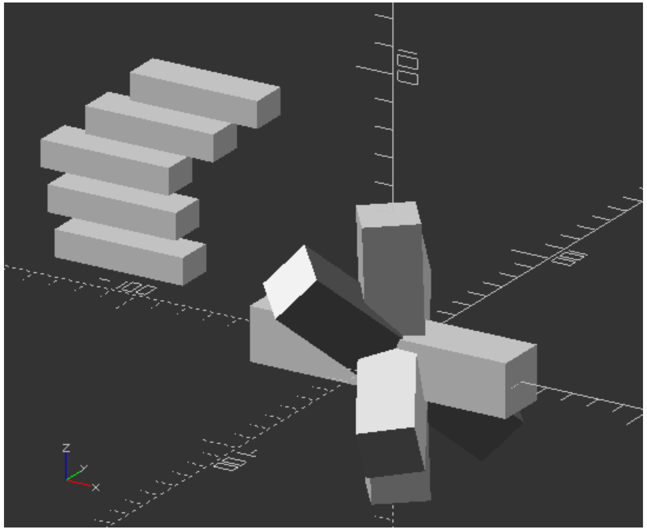
OpenSCAD - An Introduction

- Loops are operators and need to embed its content in {}

- for() builds a tree of objects, one branch for each item

- Syntax: for() {action() object();}

- Another operator: intersection_for()



Translate and rotate loop using for()

Loops can be used to create group objects as an union

Statements

Syntax for if() statement is:

```
if(test) scope1
if(test) {scope1}
if(test) scope1 else scope2
if(test) {scope1} else {scope2}
```

There are 2 more functions (let(), Conditional ?), which are not discussed in this course

An if() statement tests one or more condition/s

Modules

OpenSCAD - An Introduction

Syntax:

```
module_name(parameters){actions}
```

- There is no naming convention on the module name
- Best approach is using variables to keep it universal
- Although the variables will be used inside of the module, no variables can send back outside of the module

Modules are like libraries you can import and use

Variables

OpenSCAD - An Introduction

Syntax:

```
   variable = value;
```

- Variables in OpenSCAD are more likely constants, the last value of a variable is the one, that will be used
- Self assignments are not working (x=x+1)
- Variables are usually on top of the code, so changing them is easy

Variables are essential for parametric modeling

Examples

OpenSCAD - An Introduction

openSCAD_CodingCADModels_Examples are some full examples as an addition to the workshop file.

Please feel free and experiment with this files, and if you have any questions, please let me know!

contact@jensmeisner.net

!THANK YOU FOR LISTENING!

Example file: /openSCAD_CodingCADModels_Examples_en.scad