****NOTE: ALL CODE IS CASE SENSITIVE!**

OpenScad Basic Shape Functions

cube ([x, y, z]); *(creates a cube with dimensions x, y and z)*

sphere ();

r = *(radius)*

cylinder ();

r = *(radius if cylinder is the same on both sides. Don't use if using r1 and r2)*

r1 = *(radius of bottom)*

r2 = *(radius of top)*

h = *(height)*

Example: cylinder (r1=5, r2 = 3, h = 10);

Basic Move Functions

translate ([x,y,z]) *(moves a shape in the x/y/z directions. Variables in mm. Note the lack of semi-colon)*

rotate ([x,y,z]) *(rotates a shape in the x/y/z directions. Variables in degrees. Note the lack of semi-colon)*

Basic Combination Functions

union () *(empty brackets)*

{

    Object 1;

    Object 2;

    Object 3;

}

(This joins objects 1-3 together. You can have any number of objects involved in the union)

difference () *(empty brackets)*

{

    Object 1;

    Object 2;

}

(This subtracts object 2 from object 1. I like setting up differences using a combination of the two. See code on next page)

```
difference ()    (empty brackets)
{
        union ()
        {
                Positive object 1;
                Positive object 2;  (etc)
        }

        union ()
        {
                Negative object 1;
                Negative object 2;  (etc)
        }
}
```

Advanced Functions

for loop

```
for ( i = [0 : 5] )
{
   rotate ( i * 360 / 6, [1, 0, 0])
   translate ([0, 10, 0])
   sphere (r = 1);
}
```

if statement

```
if (a==b) dosomething();
```

show hex

Modules

In main body

NameOfModule (variable1, variable2, ...);

In modules

module NameOfModule (localvariable1, localvariable2, ...)

{

WhateverTheModuleDoes ();

}

Drawing Your Own Shapes

polygon ([[point0 x, point 0 y], [point1 x, point 1 y] , [point2 x, point 2 y]] , [[point 0, point 1, point 2]]);

(point order (the second [[ ]]) doesn't have to be 0,1,2. It should be in whatever order the polygon is drawn. Point order is a single number corresponding the point x and y you defined first. Note that it starts with 0)

linear_extrude (height = z)

*(note lack of semi-colon. This goes **before** the polygon to extrude it into 3d space.)*