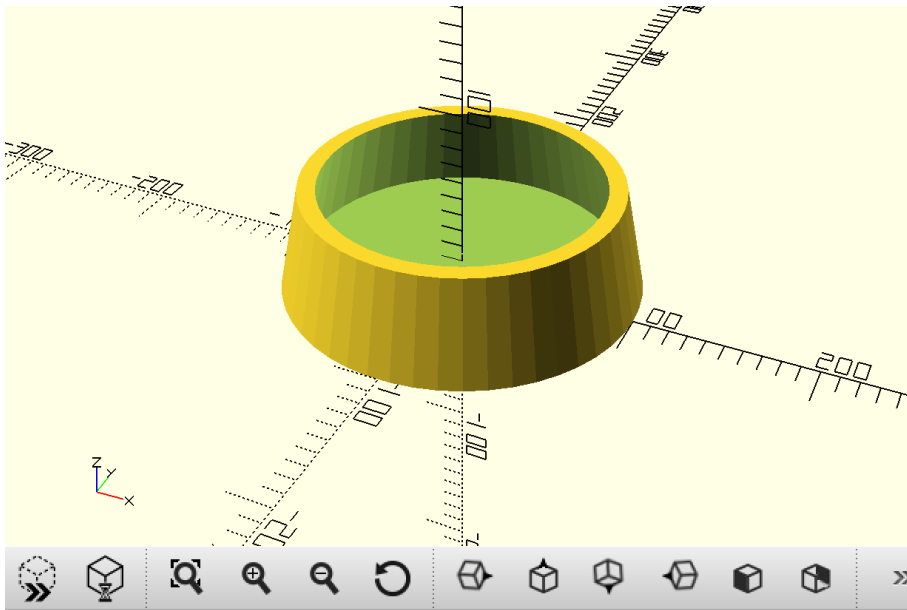


Malvina OLIVERIO

Exercice 1 : créer un objet utile / fonctionnel

Mon objet est une gamèle pour animaux domestiques (chats, chiens, lapins etc.) J'ai choisi cet objet car il est utile et aide au quotidien pour les propriétaires d'animaux et les animaux eux-même. Il permet de contenir de la nourriture pour que les animaux puissent manger en autonomie une fois que leur propriétaires en a mis dedans et ainsi favorise le confort de l'animal. De plus, créer cet objet dans OpenSCAD est un bon exercice pour s'entrainer à créer des formes simples comme les cylindres ou des commandes comme rotations, translations etc et donc à manipuler le logiciel.

[objet utile Malvina OLIVERIO.scad](#)



Paramètres d'impression :

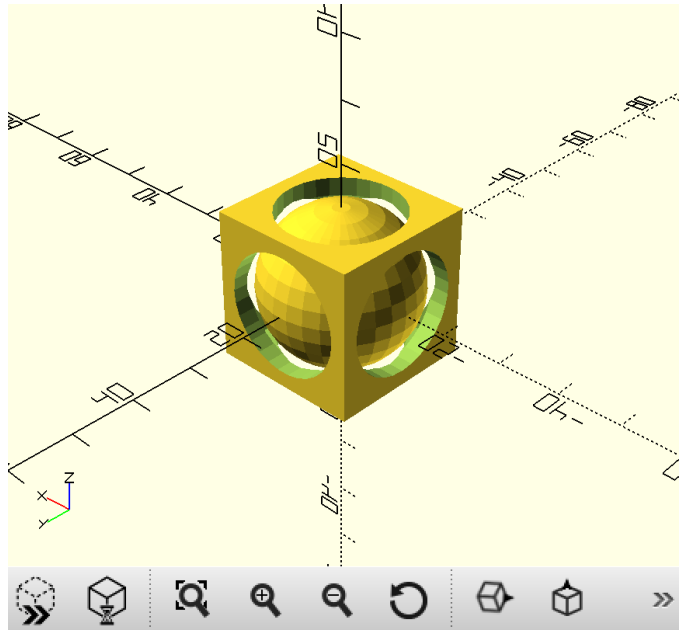
J'ai utilisé les paramètres d'impression standard, par défaut de 0,20 mm SPEED avec le filament Prusament PLA, avec un remplissage de 15% en ligne monotone, une température de 230°. J'ai modifié seulement la hauteur de couche et hauteur de la première couche de 0,2 mm, avec 3 couches solides au dessus et en dessous, avec une épaisseur de la couche minimale au dessus de 0,8 mm et en dessous de 0,6 mm.

La buse est par défaut donc 0,4mm pour qu'elle soit compatible avec l'imprimante Prusa.

Je n'ai pas utilisé de support ni de jupe ni bordure car mon objet à une base plane et relativement simple donc j'ai considéré que ceux-ci n'étaient pas nécessaires.

Exercice 2 : [cube sphère 3D.scad](#)

// Paramètres



```
taille_cube = 25;    // Taille du cube extérieur
```

```
rayon_sphere = 13;    // Rayon de la sphère intérieure  
espace = 3;           // Distance entre la sphère et le cube  
rayon_trou = 10;      // Rayon des trous sur les faces du cube
```

// Module cube extérieur avec des trous

```
module cube_avec_trous(taille_cube, espace, rayon_trou) {  
  difference() {  
    // Cube extérieur  
    cube([taille_cube, taille_cube, taille_cube], center = true);  
  
    // Creux intérieur  
    cube([taille_cube - 2 * espace, taille_cube - 2 * espace, taille_cube - 2 * espace], center =  
true);
```

// Trous

```
for (i = [-1, 1]) {  
  // Trous sur les faces X  
  translate([i * taille_cube / 2, 0, 0])  
    rotate([0, 90, 0])  
    cylinder(h = taille_cube, r = rayon_trou, center = true);
```

// Trous sur Y

```
translate([0, i * taille_cube / 2, 0])  
  rotate([90, 0, 0])  
  cylinder(h = taille_cube, r = rayon_trou, center = true);
```

```
// Trous sur Z
translate([0, 0, i * taille_cube / 2])
  cylinder(h = taille_cube, r = rayon_trou, center = true);
}
}
}
// Module sphère intérieure
module sphere_interieure(rayon) {
  translate([0, 0, 0])
    sphere(r = rayon);
}

// Combinaison cube+sphère
union() {
  // Cube extérieur avec trous
  cube_avec_trous(taille_cube, espace, rayon_trou);

  // Sphère intérieure
  sphere_interieure(rayon_sphere);
}
```

Revision #10

Created 25 November 2024 08:29:32 by Oliverio Malvina

Updated 2 December 2024 12:56:23 by Oliverio Malvina