

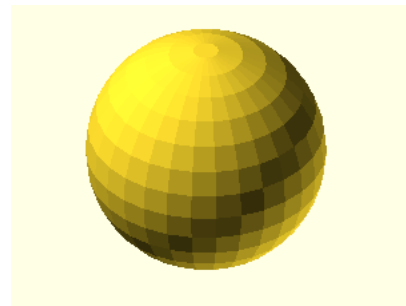
OpenSCAD Schachfigur zeichnen

Eine Schachfigur zeichnen in 10 Schritten

OpenSCAD ist eine freie CAD-Software auf einer textbasierten Programmiersprache mit der 3D-Modelle erzeugt werden. Die Objekte bestehen aus einfachen geometrischen Grundkörpern und werden mit Transformationen und Modifikationen zu einem komplexen 3D-Modell vereinigt. Im folgenden Beispiel erstellen wir auf diesen Weg eine Schachfigur.

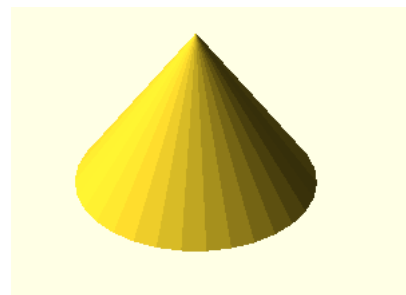
Schritt 1: Kugel

```
sphere(r = 20);
```



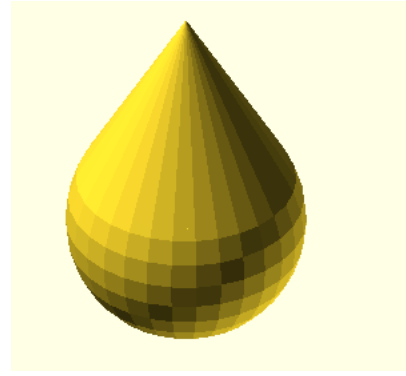
Schritt 2: Kegel

```
cylinder(h = 30, r1 = 20, r2 = 0);
```



Schritt 3: Kegel positionieren

```
union()  
{  
  sphere(r = 20);  
  translate([0, 0, 10 ])  
  cylinder(h = 30, r1 = 17.3, r2 = 0);  
}
```



Schritt 4: Einschnitt

```
difference()  
{  
  union()  
  {  
    sphere(r = 20);  
    translate([0, 0, 10])  
    cylinder(h = 30, r1 = 17.3, r2 = 0);  
  }  
  cube([40, 5, 40]);  
}
```

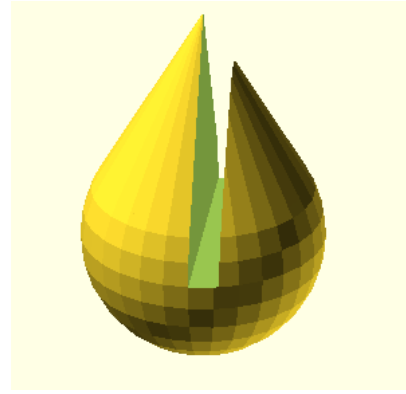


Schritt 5: Einschnitt positionieren

```

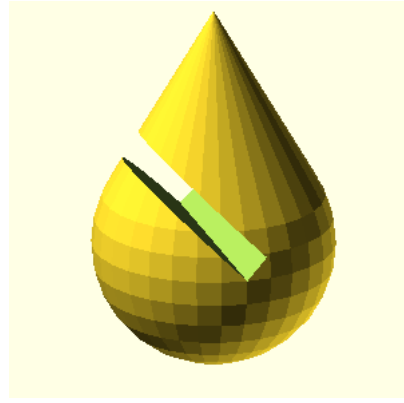
union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);
}
//Einschnitt
translate([-20, 0, 0])
cube([40, 5, 40]);
}
}

```



Schritt 6: Einschnitt im richtigen Winkel

```
union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);
}
//Einschnitt
rotate([45, 0, 0])
translate([-20, 0, 0])
cube([40, 5, 40]);
}
}
```



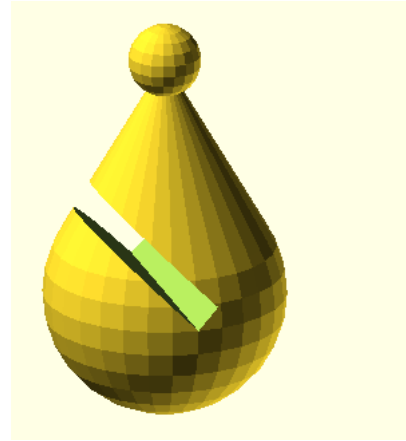
Schritt 7: Kugel an der Spitze

```

union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);

// Kugel an der Spitze
translate([0, 0, 30 + 10])
sphere(r = 6);
}
//Einschnitt
rotate([45, 0, 0])
translate([-20, 0, 0])
cube([40, 5, 40]);
}
}

```



```

union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);

// Kugel an der Spitze
translate([0, 0, 30 + 10])
sphere(r = 6);
}
//Einschnitt
rotate([45, 0, 0])
translate([-20, 0, 0])
cube([40, 5, 40]);
}
// Hals
cylinder(h = 120, r1 = 18, r2 = 12);
}

```



```

union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);

// Kugel an der Spitze
translate([0, 0, 30 + 10])
sphere(r = 6);
}
//Einschnitt
rotate([45, 0, 0])
translate([-20, 0, 0])
cube([40, 5, 40]);
}
// Hals
cylinder(h = 120, r1 = 18, r2 = 12);

// Sockel
cylinder(h = 20, r1 = 35, r2 = 25);
}

```



```

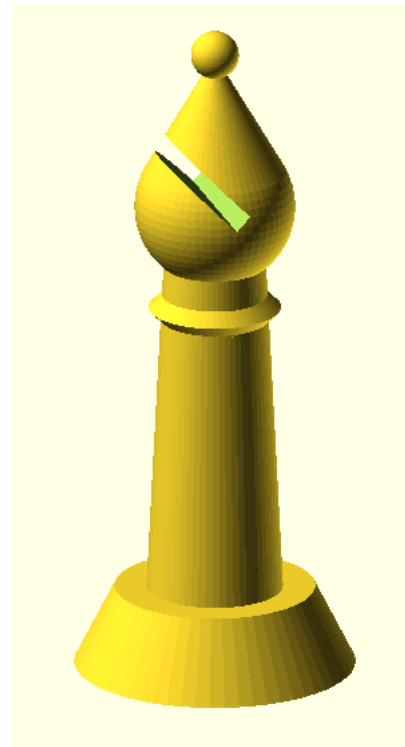
$fn=60; // Rendern
union()
{
// Kopf
translate([0, 0, 120])
difference()
{
union()
{
// Tropfenform
sphere(r = 20);
translate([0, 0, 10])
cylinder(h = 30, r1 = 17.3, r2 = 0);

// Kugel an der Spitze
translate([0, 0, 30 + 10])
sphere(r = 6);
}
//Einschnitt
rotate([45, 0, 0])
translate([-20, 0, 0])
cube([40, 5, 40]);
}
// Hals
cylinder(h = 120, r1 = 18, r2 = 12);

// Sockel
cylinder(h = 20, r1 = 35, r2 = 25);

// Kragen
translate([0, 0, 90])
intersection()
//Erstellt den Schnittpunkt aller
untergeordneten Knoten.
{
cylinder(h = 20, r1 = 20, r2 = 0);
translate([0, 0, 7])
mirror([0, 0, 1])
cylinder(h = 20, r1 = 20, r2 = 0);
}
}

```



Version #1

Erstellt: 2025-03-30 22:36:26 UTC von Joel Hatsch

Zuletzt aktualisiert: 2025-03-30 22:46:07 UTC von Joel Hatsch