

Processing

Bok 6



Figure 1: Bok 6: 3D

#	Beskriving
21	3D 1: klotet rör sig
22	3D 2: styra klotet
23	3D 3: klotet och kuben
24	3D 4: ljussättning

Contents

Förord	1
3D-lektion 1: klotet rör sig	2
3D Lektion 2: styra klotet	9
3D Lektion 3: klotet och kuben	15
3D Lektion 4: ljussättning	21

Förord

Detta är en bok om Processing för ungdomar. Processing är ett programmeringsspråk. Denna bok lär dig det programmeringsspråket.

Om den här boken

Denna bok är licensierad av CC-BY-SA.



Figure 1: Licensen för denna bok

(C) Richèl Bilderbeek och alla lärare och alla elever

Med det här häftet kan du göra vad du vill, så länge du hänvisar till originalversionen på denna webbplats: https://github.com/richelbilderbeek/processing_foer_ungdomar. Detta häfte kommer alltid att förbli gratis, fritt och öppet.

Det är fortfarande en lite slarvig bok. Det finns stafvel och *layouten är inte alltid vacker*. Eftersom den här boken finns på en webbplats kan alla som tycker att den här boken är för slarvig göra den mindre slarvig.

3D-lektion 1: klotet rör sig

Under den här lektionen ska vi programmera 3D.

3D 1: uppgift 1

Skriv denna kod över:

```
float x = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  translate(x, 100, 0);
  lights();
  sphere(20);
  x = x + 1;
}
```

Vad ser du?

3D 1: lösning 1



Figure 2: 3D 1: lösning 1

Du ser en boll som rör sig från vänster till höger.

Om man tittar noga kan man se att den vitaste delen av bollen också byter plats.

3D 1: uppgift 2

- Låt nu bollen gå längst ner till höger
- Skapa en ny variabel, till exempel y

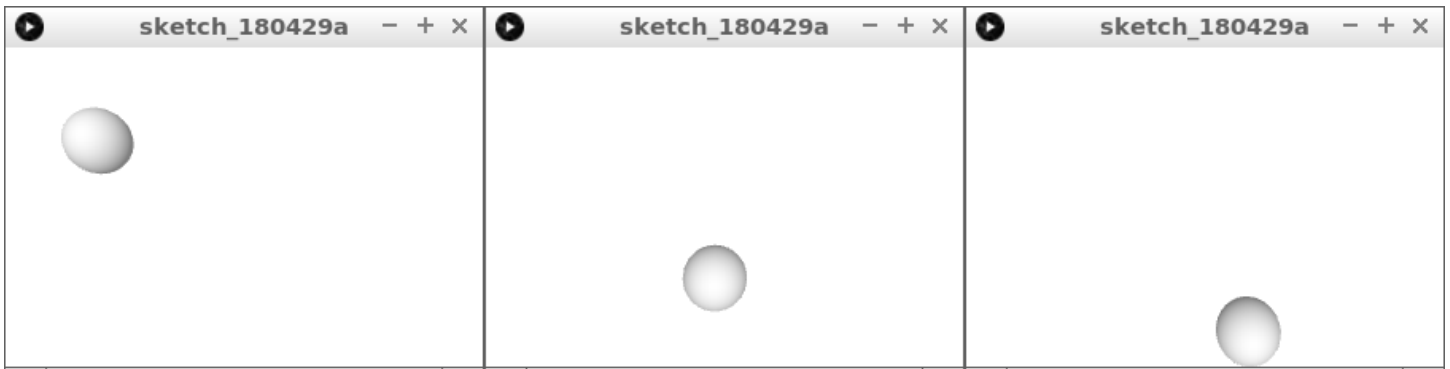


Figure 3: 3D 1: uppgift 2

3D 1: lösning 2

```
float x = 0;
float y = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  translate(x, y, 0);
  lights();
  sphere(20);
  x = x + 1;
  y = y + 1;
}
```

3D 1: uppgift 3

- Låt nu bollen gå framåt
- Skapa en ny variabel, till exempel z

Det här kan se annorlunda ut än du förväntar dig!

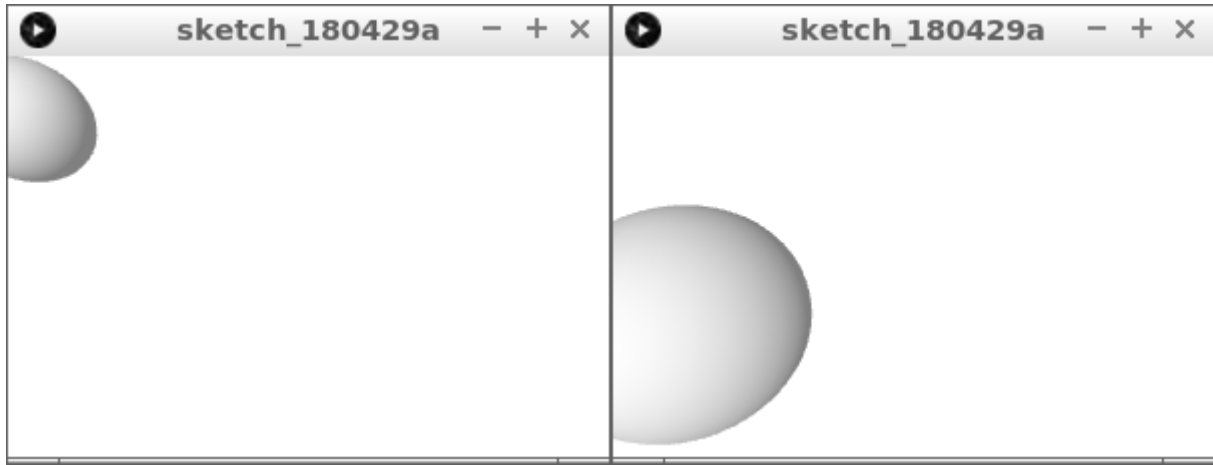


Figure 4: 3D 1: uppgift 3

3D 1: lösning 3

```
float x = 0;
float y = 0;
float z = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  translate(x, y, z);
  lights();
  sphere(20);
  x = x + 1;
  y = y + 1;
  z = z + 1;
}
```

3D 1: Slutuppgift

- Gör storleken på sfären lika med x.
- Ta bort `noStroke`

Det här kan se annorlunda ut än du förväntar dig!

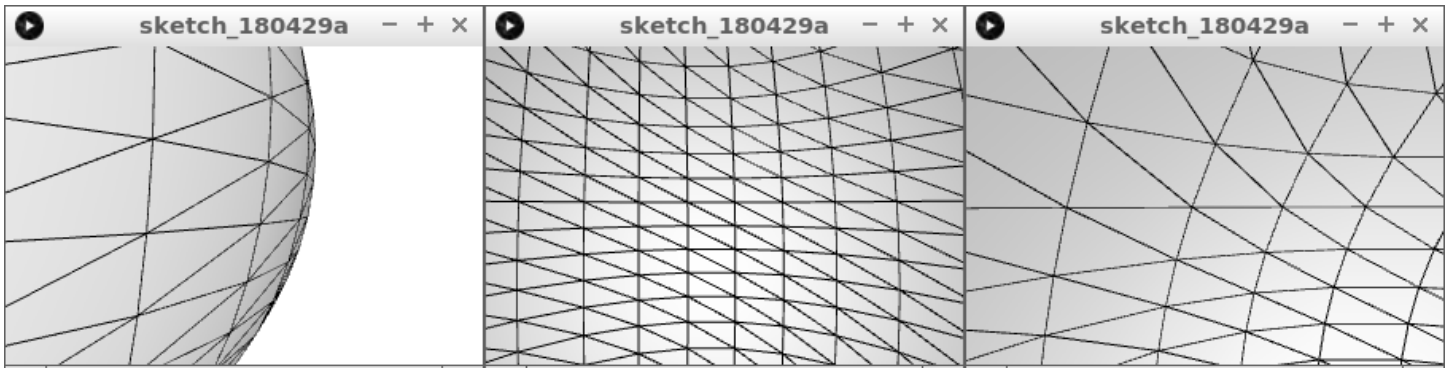


Figure 5: 3D 1: Slutuppgift

3D Lektion 2: styra klotet

Under den här lektionen ska vi göra en 3D-bollrörelse.

3D 2: uppgift 1

Skriv denna kod över:

```
float x = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
  x = width / 2;
}

void draw()
{
  background(255, 255, 255);
  lights();
  translate(x, height / 2, 0);
  sphere(20);
  if (keyPressed)
  {
    if (key == 'a') x = x - 1;
    if (key == 'd') x = x + 1;
  }
}
```

Vad ser du?

3D 2: lösning 1



Figure 6: 3D 2: lösning 1

Du ser en boll som du kan gå till med **a** och **d** kan flytta åt vänster och höger.

Om man tittar noga kan man se att den vitaste delen av bollen också byter plats.

3D 2: uppgift 2

- Se till att bollen nu också kan gå upp och ner, med tangenterna **w** och **s**
- Skapa en ny variabel, t.ex. “y”, med initialvärdet “höjd / 2”.



Figure 7: 3D 2: uppgift 2

3D 2: lösning 2

```
float x = 0;
float y = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  lights();
  translate(x, y, 0);
  sphere(20);
  x = x + 1;
  y = y + 1;
}
```

3D 2: Slutuppgift

- Få bollen att gå fram och tillbaka med tangenterna q och e
- Skapa en ny variabel, t.ex. “ z ”, med initialt värde noll

Det här kan se annorlunda ut än du förväntar dig!

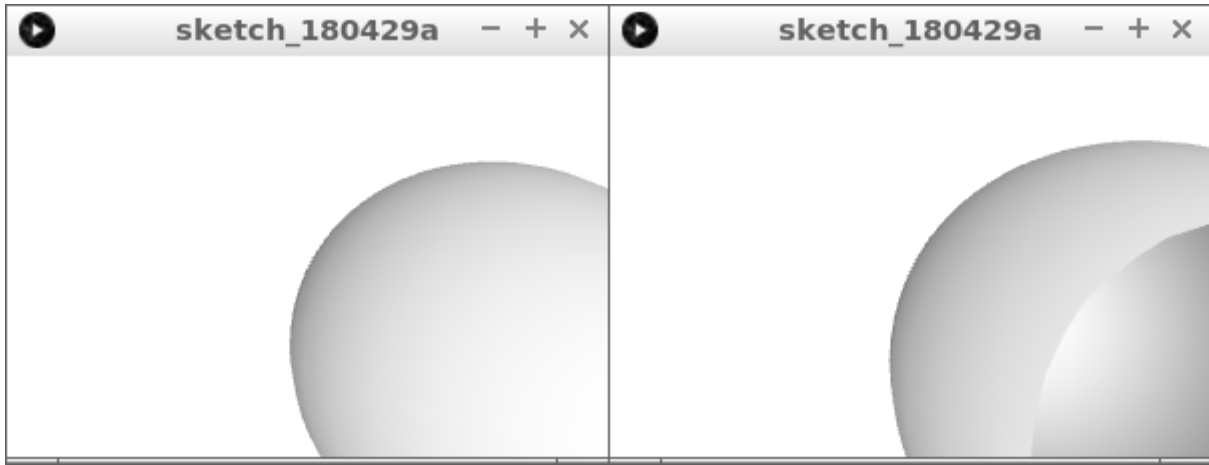


Figure 8: 3D 2: uppgift 3

3D Lektion 3: klotet och kuben

Under den här lektionen ska vi göra en 3D-kula och kub-rörelse.

3D 3: uppgift 1

Skriv denna kod över:

```
float x1 = 0;
float x2 = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
  x1 = width * 1 / 4;
  x2 = width * 3 / 4;
}

void draw()
{
  background(196, 196, 196);
  lights();
  translate( x1,  height / 2, 0);
  box(20);
  translate(-x1, -height / 2, 0);

  translate( x2,  height / 2, 0);
  sphere(20);
  translate(-x2, -height / 2, 0);
}
```

Vad ser du?

3D 3: lösning 1

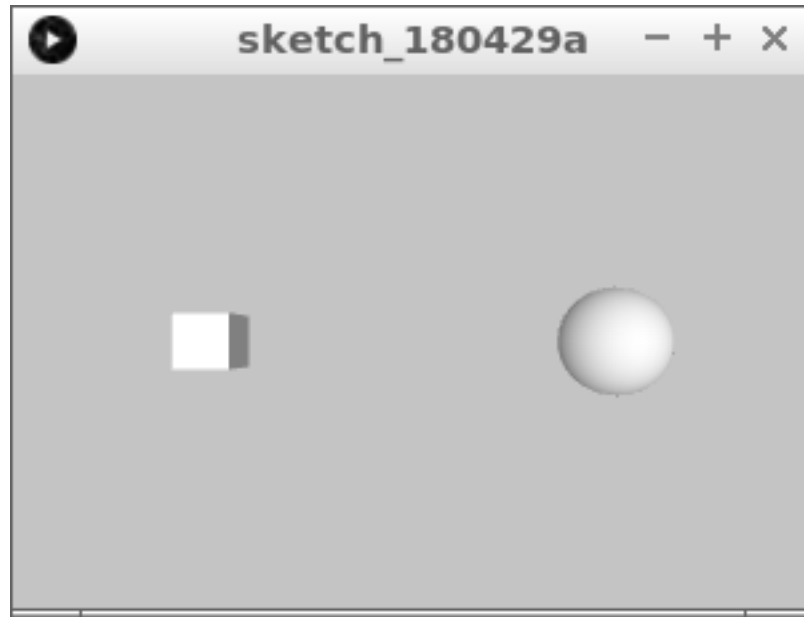


Figure 9: 3D 3: lösning 1

Du ser en kub och en boll som står stilla

3D 3: uppgift 2

- Se till att kuben nu också kan gå upp och ner, med tangenterna **w** och **s**
- Skapa en ny variabel, t.ex. “y1”, med initialvärdet “höjd / 2”.

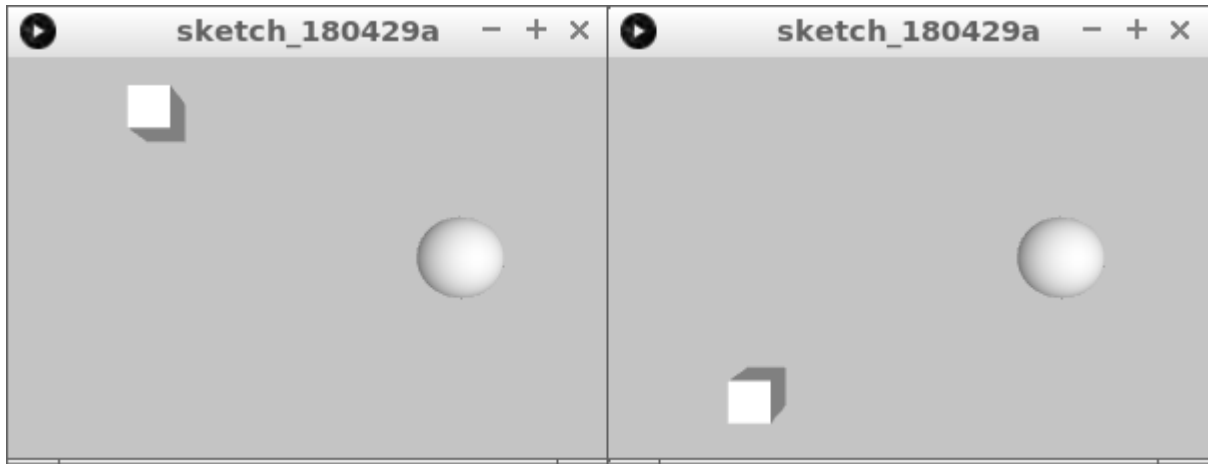


Figure 10: 3D 3: uppgift 2

3D 3: lösning 2

```
float x1 = 0;
float x2 = 0;
float y1 = 0;

void setup()
{
  size(300, 200, P3D);
  noStroke();
  x1 = width * 1 / 4;
  x2 = width * 3 / 4;
  y1 = height / 2;
}

void draw()
{
  background(196, 196, 196);
  lights();
  translate( x1,  y1, 0);
  box(20);
  translate(-x1, -y1, 0);

  translate( x2,  height / 2, 0);
  sphere(20);
  translate(-x2, -height / 2, 0);

  if (keyPressed)
  {
    if (key == 'w') y1 = y1 - 1;
    if (key == 's') y1 = y1 + 1;
  }
}
```

3D 3: Slutuppgift

- Låt nu också bollen gå fram och tillbaka med tangenterna **i** och **k**
- Skapa en ny variabel, t.ex. “y2”, med initialvärdet “höjd / 2”.

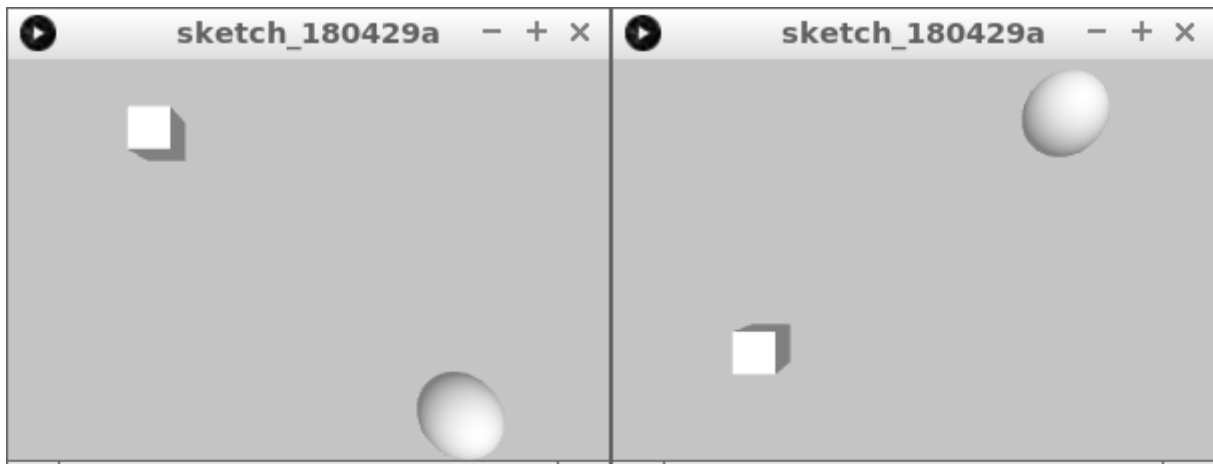


Figure 11: 3D 3: uppgift 3

3D Lektion 4: ljussättning

Under den här lektionen ska vi exponera en 3D-boll.

3D 4: uppgift 1

Skriv denna kod över:

```
float ljus_rikting_x = 1;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  directionalLight(255, 255, 255, ljus_rikting_x, 0, 0);
  translate(width / 2, height / 2, 0);
  sphere(50);
}
```

Vad ser du?

3D 4: lösning 1

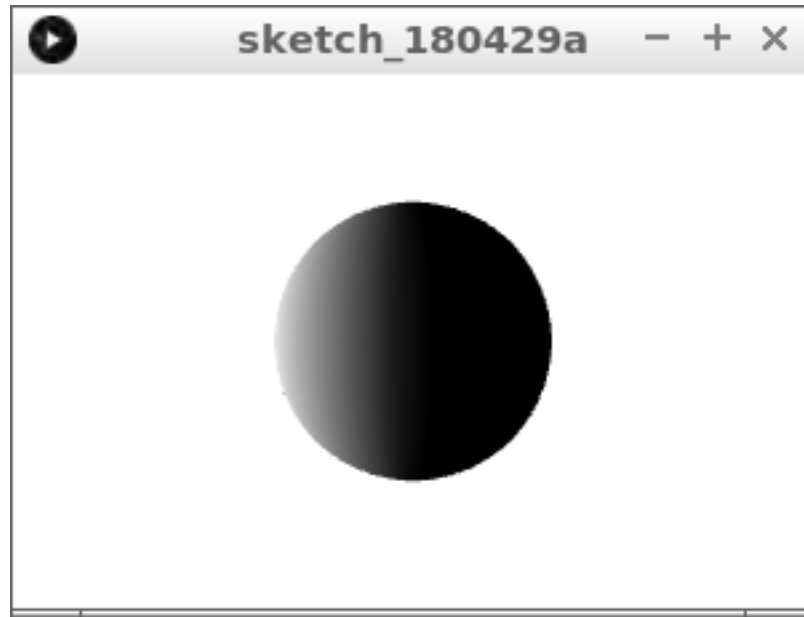


Figure 12: 3D 4: lösning 1

Du kommer att se en boll med ett vitt ljus på.

3D 4: uppgift 2

- Se till att `ljus_riktning_x` kan bli lägre och högre, med tangenterna `w` och `s`

Observera: endast tre positioner är möjliga!

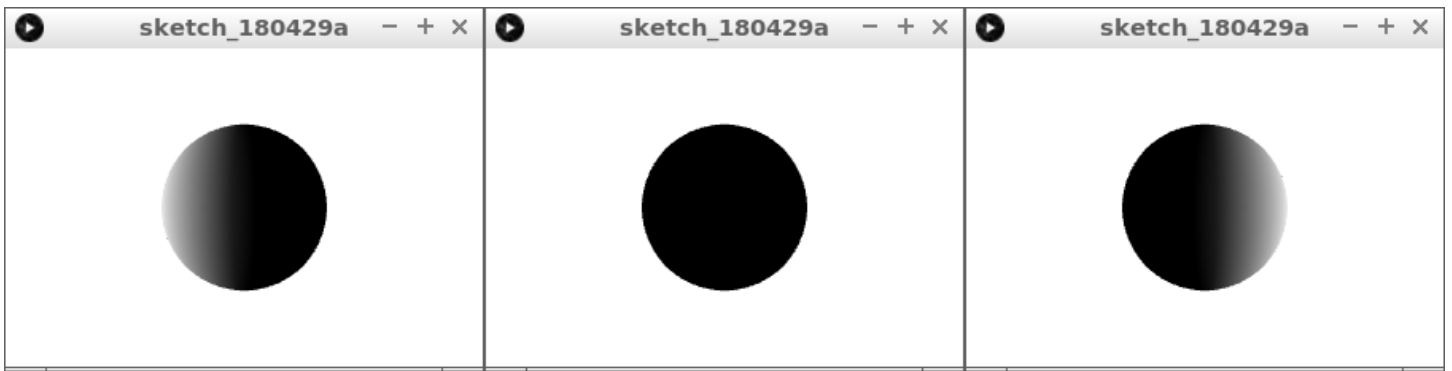


Figure 13: 3D 4: uppgift 2

3D 4: lösning 2

```
float ljus_rikting_x = 1;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  directionalLight(255, 255, 255, ljus_rikting_x, 0, 0);
  translate(width / 2, height / 2, 0);
  sphere(50);
  if (keyPressed)
  {
    if (key == 'a') ljus_rikting_x = ljus_rikting_x - 1;
    if (key == 'd') ljus_rikting_x = ljus_rikting_x + 1;
  }
}
```

3D 3: uppgift 3

- Ställ in ljusets y-riktning till ett.
- Gör färgen på ljusröd

Obs: endast fem positioner är möjliga!

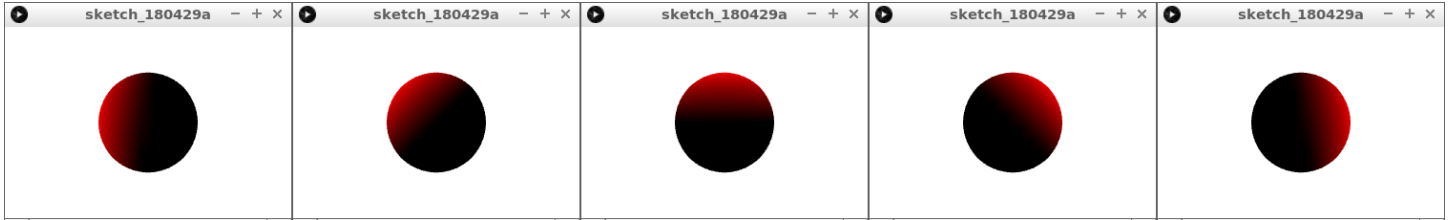


Figure 14: 3D 4: uppgift 3

3D 4: lösning 3

```
float ljus_rikting_x = 1;

void setup()
{
  size(300, 200, P3D);
  noStroke();
}

void draw()
{
  background(255, 255, 255);
  directionalLight(255, 0, 0, ljus_rikting_x, 1, 0);
  translate(width / 2, height / 2, 0);
  sphere(50);
  if (keyPressed)
  {
    if (key == 'a') ljus_rikting_x = ljus_rikting_x - 1;
    if (key == 'd') ljus_rikting_x = ljus_rikting_x + 1;
  }
}
```

3D 4: Slutuppgift

- Skapa en ny variabel, t.ex. `ljus_riktning_y`, med initialvärdet noll
- Se till att `ljus_riktning_y` kan bli lägre och högre, med tangenterna `w` och `s`
- Gör den ljusa färgen grön

Nu kan ljusstrålen rotera fint.

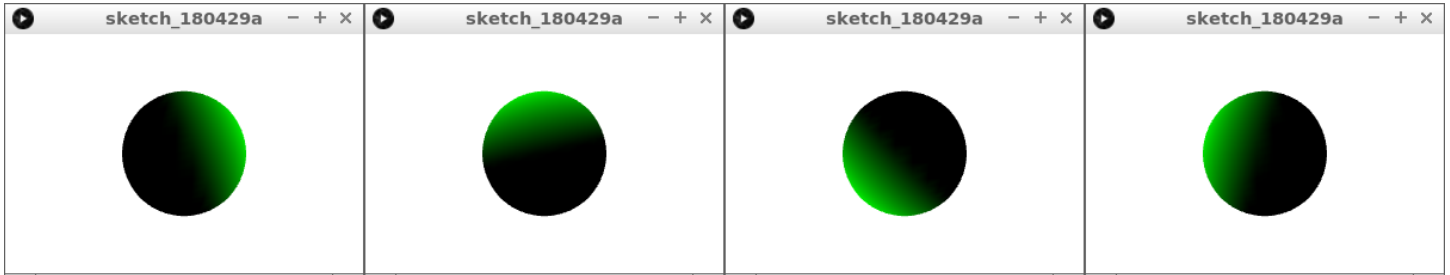


Figure 15: 3D 4: slutuppgift