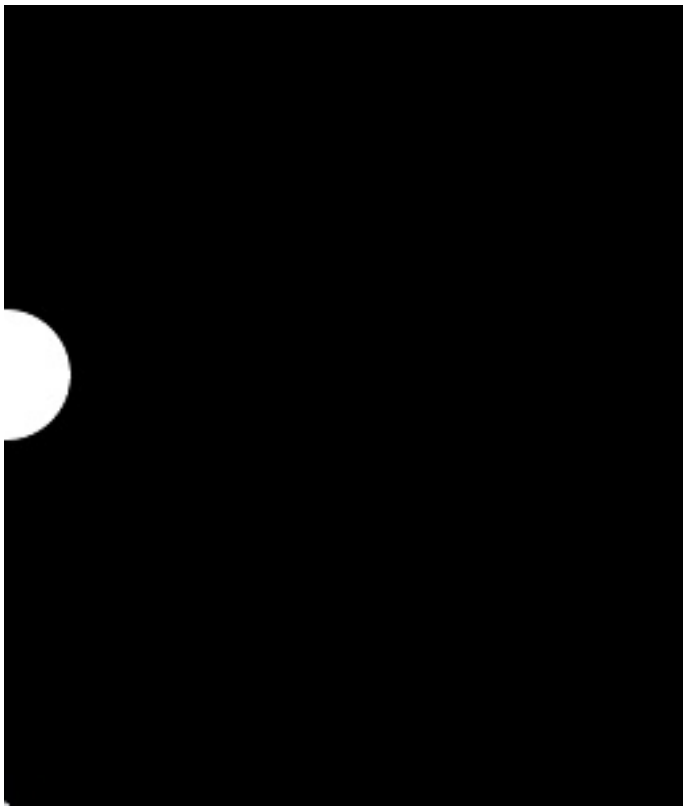
Sprettende ball

Introduksjon:

I denne modulen skal vi lære et programmeringsspråk som heter Proc
designere og andre som ikke har programmert før. Processing egner s
Mange liker Processing fordi det er raskt å lage programmer som vanl



Steg 1: Vindu

Nå skal vi begynne helt enkelt med å lage et vindu og fylle det med e
alle programmer som du lager med Processing, så det er et fint sted å

Sjekkliste

- [ ] Start Processing og skriv dette:

```
void setup() {
  size(640, 480);
}

void draw() {
  background(0);
}
```

- [ ] Kjør programmet ved å trykke på pilen  .

- [ ] Lagre programmet som Ball ved å velge **File --> Save** i menye

Tips: Hvordan skrive krøllparenteser {

Her er en oversikt over hvordan man skriver tegn som ofte brukes

Tegn	Windows/Linux	Mac
;	Shift + ,	Shift +
"	Shift + 2	Shift +
'	' (til høyre for Æ)	' (til ve
|	| (til venstre for 1)	Alt + 7
&	Shift + 6	Shift +
+	+ (til høyre for 0)	+ (til h

-	- (til høyre for .)	- (til h
*	Shift + '	Shift +
/	Shift + 7	Shift +
[	Alt Gr + 8	Alt + 8
]	Alt Gr + 9	Alt + 9
{	Alt Gr + 7	Shift +
}	Alt Gr + 0	Shift +

Utforsking

Hva skjer hvis du:

- [ ] Endrer `640` i `size(640, 480);`?
- [ ] Endrer `480`?
- [ ] Hva om du endrer `0` i `background(0);`?
- [ ] Hva skjer hvis tallet er høyere enn `255`?
- [ ] Hva skjer hvis tallet er negativt?
- [ ] Før du går videre, fjern endringene du gjorde i utforskingen.

Forklaring av koden

Selv om du har utforsket `size(640, 480)` og `background(0)`, lure
forklaring:

- [ ] `void setup() {` lager en *funksjon* som heter setup. Når `set`
  `{ }`. Du lurer kanskje på hva en funksjon er og hva det bety
  kode. Den navngitte koden kan kjøres når man trenger det, v

  en spesiell funksjon som alltid kjøres av Processing når du try

  Noen funksjoner gir tilbake en verdi som resultat når de er fe
  på funksjonen. Vi finner `void` foran `setup()` som betyr at fu
  eksempelvis ønsket å gi tilbake et heltall, ville man skrevet `i`
  integer, som er heltall på engelsk, altså 0, 1, 2, 3, -1, -2, -3, o

- [ ] `size(640, 480);` er et kall på funksjonen `size` som åpner e
  piksler. Legg merke til at du sender verdier inn til size med å
  setningen er ferdig. Dette gjør det mulig å ha flere setninger
  fordele en lang setning utover flere linjer.

- [ ] `}` på linje tre betyr at funksjonen `setup` er ferdig.

- [ ] `void draw() {` på linje 5 betyr at vi lager en funksjon som h
  om igjen så lenge programmet ditt kjører.

- [ ] `background(0);` setter bakgrunnsfargen i vinduet. Tallet `0` b
  verdien `255`. Tallene mellom 0-255 gir forskjellige gråtoner. S

- [ ] `}` på siste linje betyr at funksjonen `draw` er ferdig.

Steg 2: Sirkel

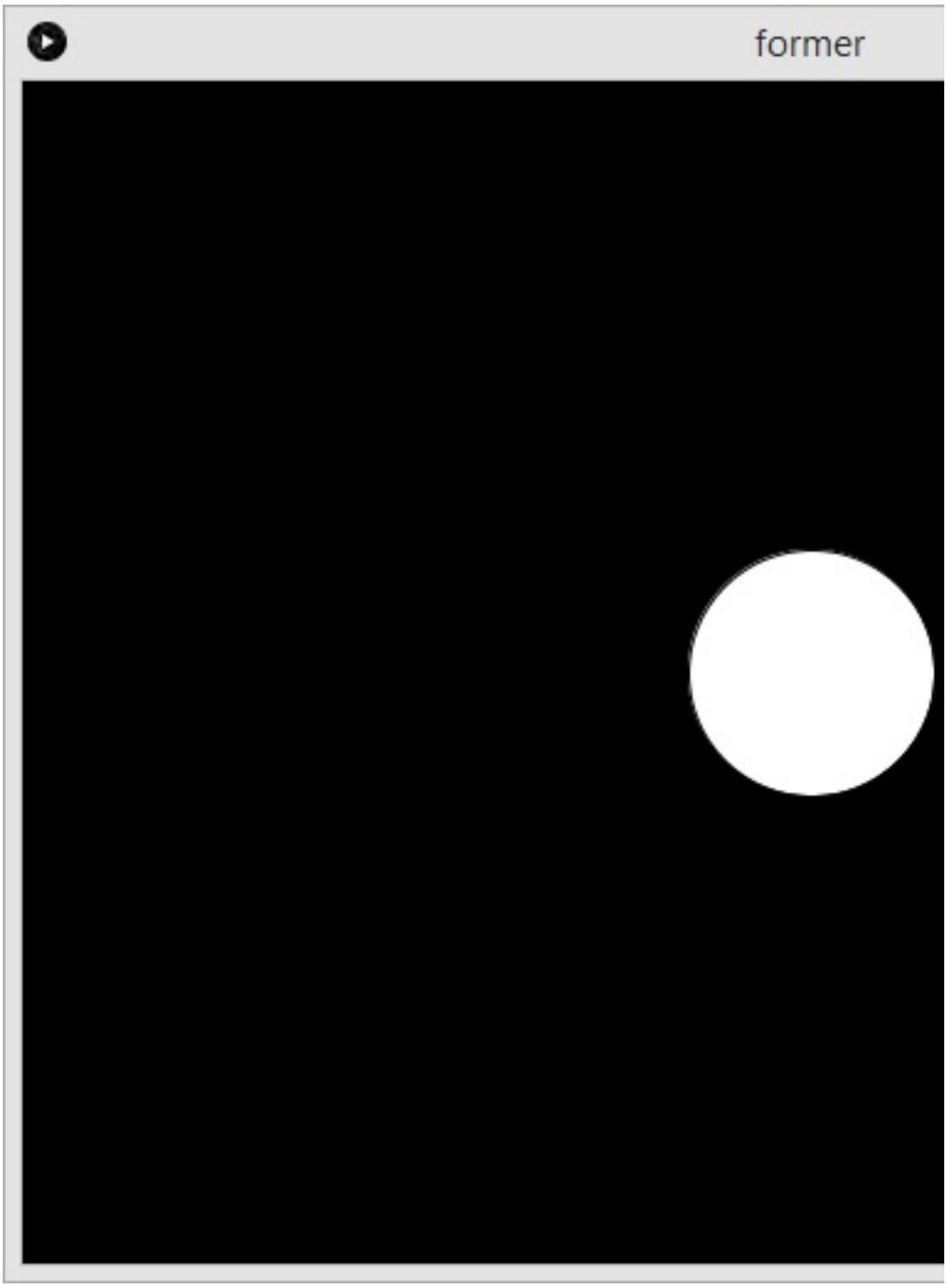
Siden denne oppgaven skal handle om en sprettende ball, er det på ti
i vinduet.

Sjekkliste

- [ ] Endre `draw` til følgende uten å endre `setup`:

```
void draw() {
  background(0);
  ellipse(320, 240, 100, 100);
}
```

- [ ] Lagre med **Ctrl+S** og kjør programmet med **Ctrl+R**. Du skal nå

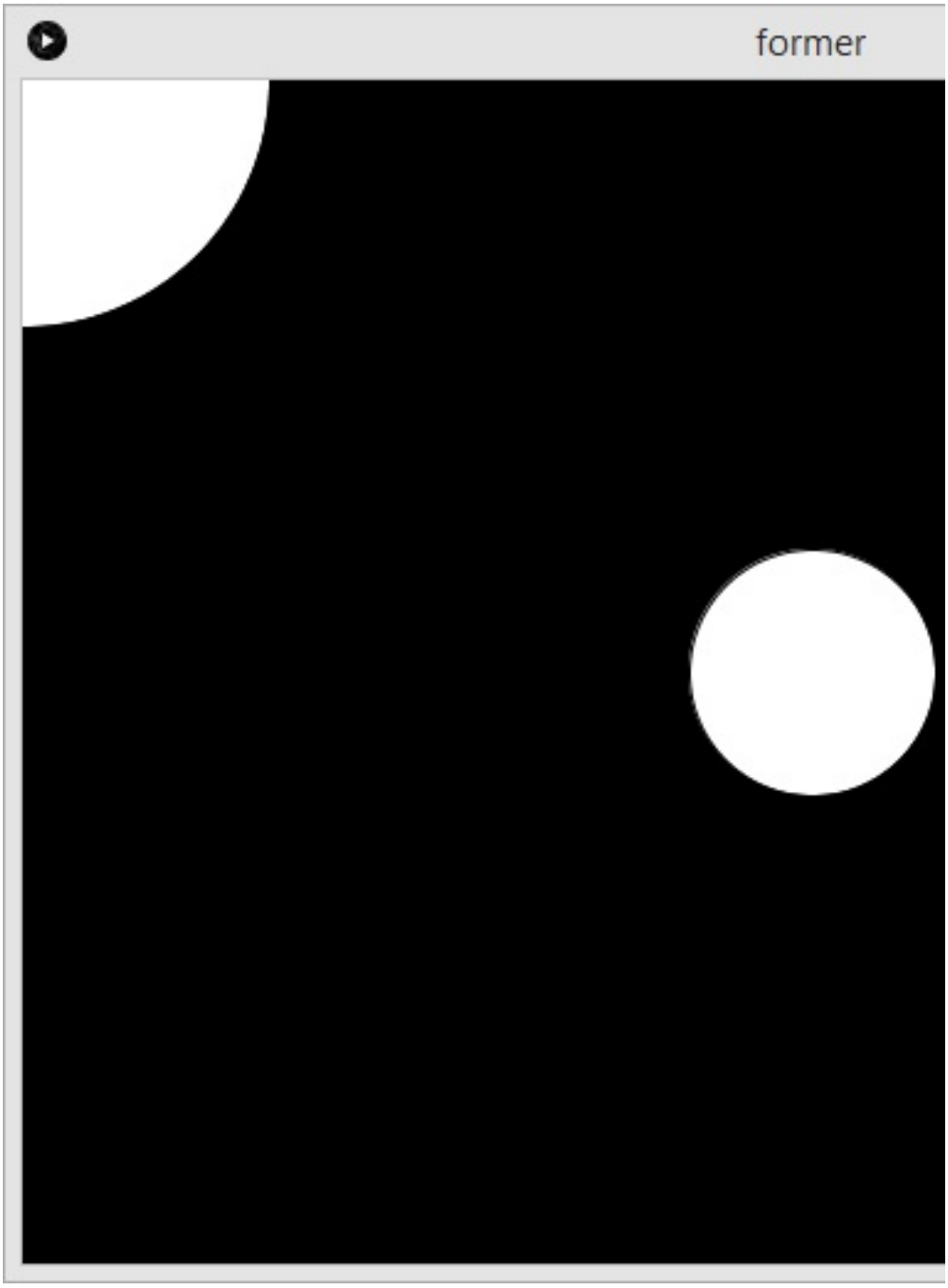


- [ ] Legg til en sirkel til og kjør programmet igjen:

```
void draw() {
  background(0);
  ellipse(320, 240, 100, 100);
  ellipse(0, 0, 200, 200);
}
```

- [ ] La oss legge til en siste sirkel og kjøre programmet enda en gar

```
void draw() {
  background(0);
  ellipse(320, 240, 100, 100);
  ellipse(0, 0, 200, 200);
  ellipse(640, 480, 50, 50);
}
```

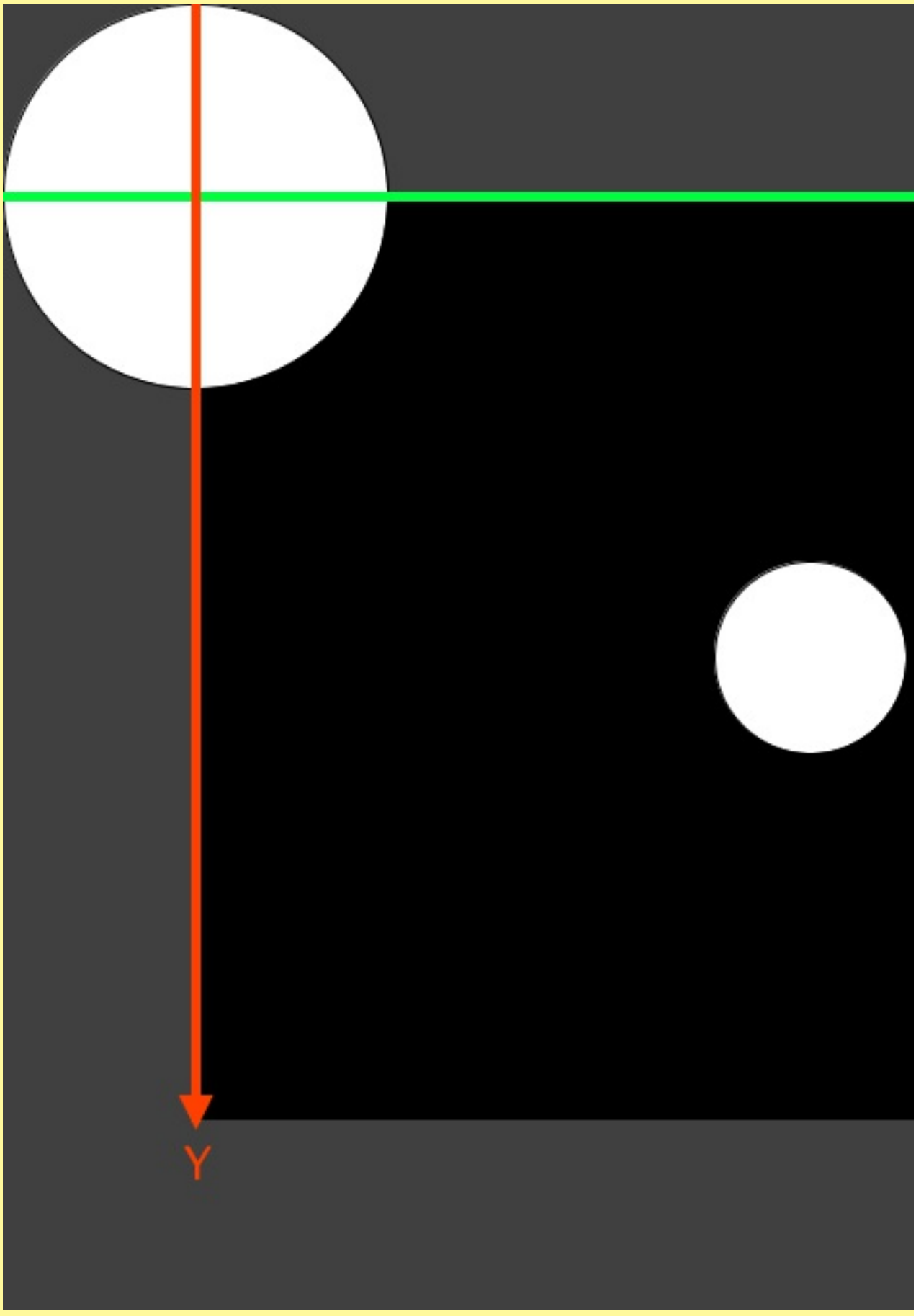


Forklaring av koden

Klarer du knekke koden for hvordan `ellipse` fungerer?

- [ ] Det første tallet bestemmer hvor langt til høyre i vinduet sirk
- [ ] Det andre tallet bestemmer hvor langt ned i vinduet den ska
- [ ] Det tredje tallet bestemmer hvor bred sirkelen skal være.
- [ ] Det siste tallet bestemmer hvor høy sirkelen er.

Det siste hørtes kanskje rart ut? En sirkel er jo like bred som den e
en sirkel, men ellipser kan også være bredere enn de er høye, eller
`ellipse` og ikke `circle`.



I bildet ovenfor vises også området utenfor bilderammen og to pile

- [ ] Det første tallet i `ellipse` angir posisjon langs X-aksen, vist
- [ ] Det andre tallet i `ellipse` angir posisjon lans Y-aksen, vist m

Der pilene krysser hverandre har både X og Y verdien `0`. Pilene str
for X og `480` for Y. Dette ble bestemt av `size(640, 480)`.

Tips: Man kan tegne opp ting utenfor bildet med negative tall eller

Steg 3: Variabler

Til nå har vi brukt faste tall overalt. Dette fungerer ikke alltid bra. For
vinduet? Vil den første sirkelen være i midten? Og hvordan kan vi få s

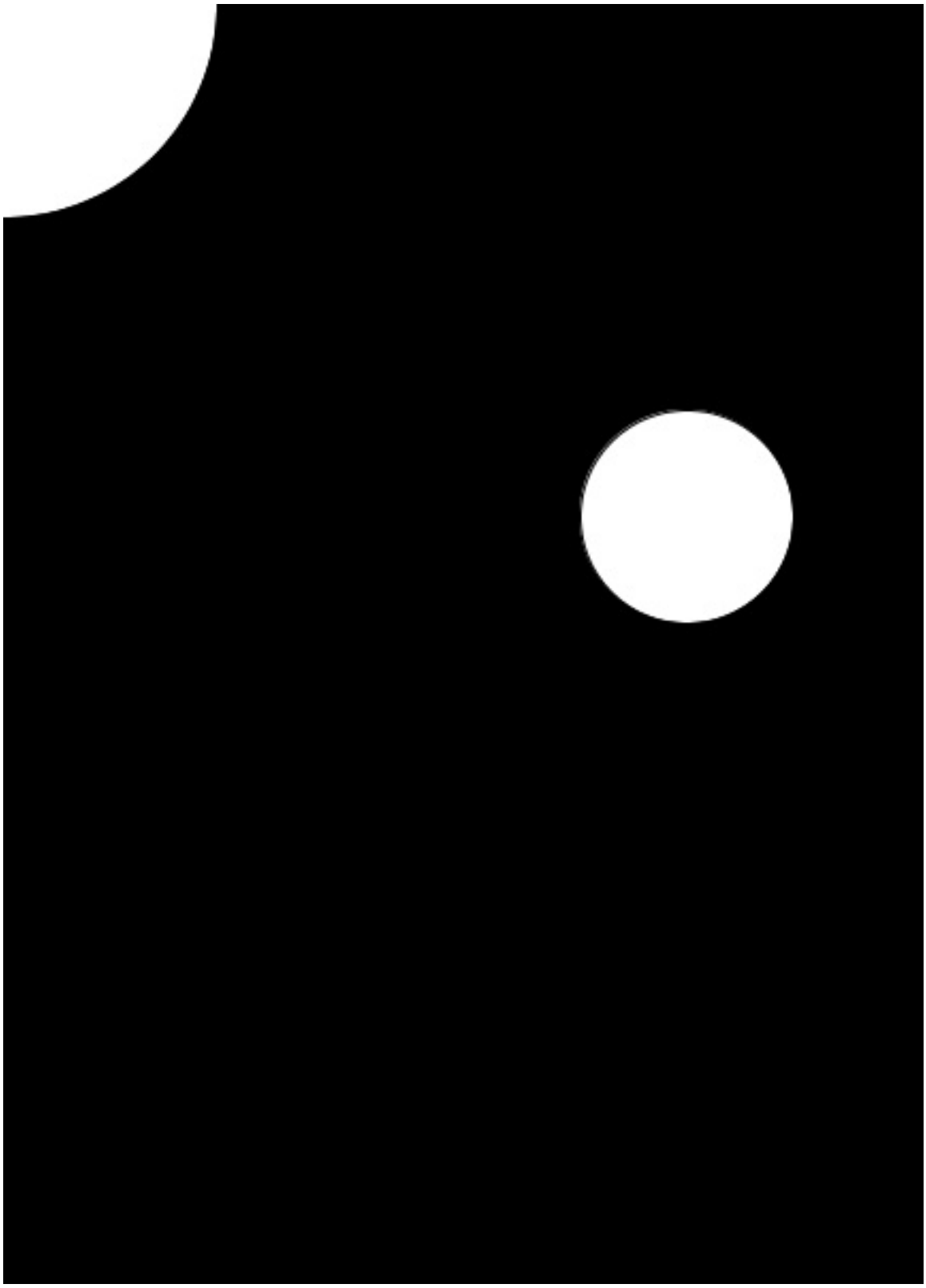
Dette løser vi ved hjelp av noe som heter *variabler*. En variabel er en
verdien til variabelen, og det er derfor det heter variabel: *verdien kan*

Sjekkliste

- [ ] La oss begynne med å endre størrelsen på vinduet i `setup`:

```
void setup() {
  size(800, 600);
}
```

Legg merke til at to av sirkelene har "flyttet" på seg. De er ikke



Vi skal nå ta i bruk to variabler som heter `width` og `height`, alt
størrelsen på vinduet når `size` kalles. Endre `draw` til å bruke `w`

```
void draw() {
  background(0);
```

```
  ellipse(width / 2, height / 2, 100, 100);
  ellipse(0, 0, 200, 200);
  ellipse(width, height, 50, 50);
}
```

Her har vi brukt regnestykkene `width / 2` og `height / 2` for å
delt på, altså gir `width / 2` halvparten av bredden. Hva gir `he`

Tips: Vi kunne også brukt `width * 0.5` for å oppnå det samme
enklere med deling og andre ganger ganging.

- [ ] Lagre og kjør programmet, om du ikke har gjort det allerede.

- [ ] La oss lage våre egne variabler, slik at vi kan få formene til å be

```
float x;
float y;

void setup() {
  size(800, 600);
  x = width / 2;
  y = height / 2;
}
```

`float x;` og `float y;` lager to variabler med navn `x` og `y`. Ty
flyttall. Inne i `setup` gir vi variablene verdier, som er de samme

- [ ] Det er ikke nok å bare ha variabler, vi må bruke dem også. Endr

```
void draw() {
  x = x + 1;

  background(0);
  ellipse(x, y, 100, 100);
  ellipse(0, 0, 200, 200);
  ellipse(width, height, 50, 50);
}
```

Den første sirkelen bruker nå `x` og `y` som posisjon. I tillegg la v
for hver gang `draw` kjøres. Hva vil skje med den ene sirkelen nå

- [ ] Lagre og kjør programmet, om du ikke har gjort det allerede.

Tips: Lagre som

Hvis du ikke vil miste de forskjellige stegene i denne oppgaven kan
(*Lagre*). Du finner dette under **File -> Save as** ved å trykke **Ctrl +**
slik at du beholder de forskjellige variantene.

Utforsking

Kan du endre `x` og `y` inni `draw` slik at sirkelen beveger seg:

- [ ] Mot venstre istedenfor høyre?
- [ ] Opp istedenfor sidelengs?
- [ ] Ned istedenfor for opp?
- [ ] På skrå?

Steg 4: Sprette i veggen

Det er kjedelig når sirkelen forsvinner ut av vinduet hele tiden. Vi skal
vinduet, slik en ball spretter tilbake hvis den kastes i en vegg.

Sjekkliste

- [ ] Vi trenger et par nye variabler for å styre retningen til sirkelen.

```
float xFart = 1.5;
float yFart = 2;
```

Variablene har type `float`, altså flyttall eller desimaltall. Eksem
farten og retningen til sirkelen. I Processing, og i de fleste progr
på desimaltall. Dette er fordi punktum er det som brukes i enge

Notis: La du merke til at variablene ble gitt verdier med en gan
Grunnen til dette er at `width` og `height` kan ikke brukes før `si`
en gang.

- [ ] Legg til koden under i `draw` for å få ballen til å snu. Merk at vi e
sirklene.

```
void draw() {
  x = x + xFart;
  y = y + yFart;

  if (x < 50) {
    xFart = -xFart;
  }

  if (x > width - 50) {
    xFart = -xFart;
  }

  if (y < 50) {
    yFart = -yFart;
  }

  if (y > height - 50) {
    yFart = -yFart;
  }
```

```
  background(0);
  ellipse(x, y, 100, 100);
}
```

- [ ] Lagre og kjør programmet.

Forklaring

I `draw` ser vi en del nytt som du ikke har sett før.

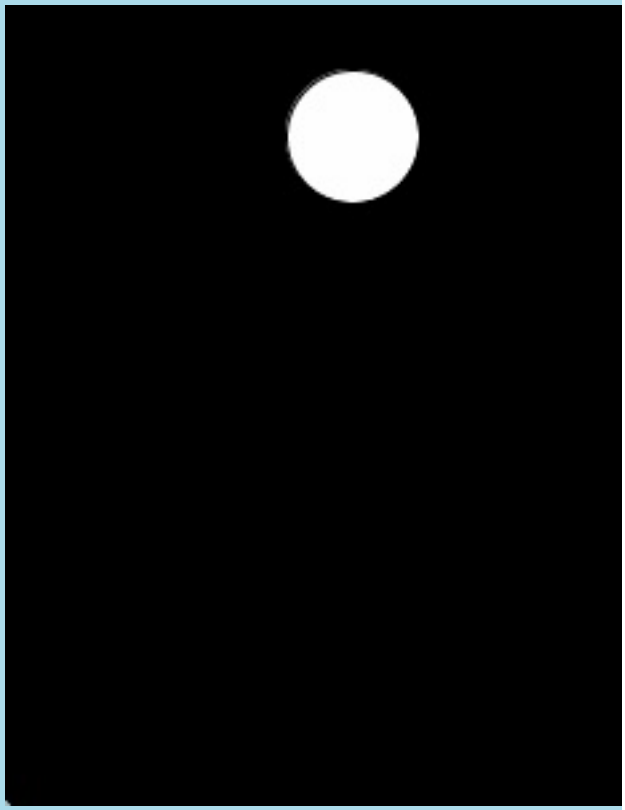
- [ ] `if (x < 50)`, *hvis x er under 50*, lar oss kjøre `xFart = -xFa` `if`-setningene?

- [ ] `xFart = -xFart;` endrer fortegnet på farten. Dersom farten -). Dersom farten er negativ får vi to minus, som er positiv (-

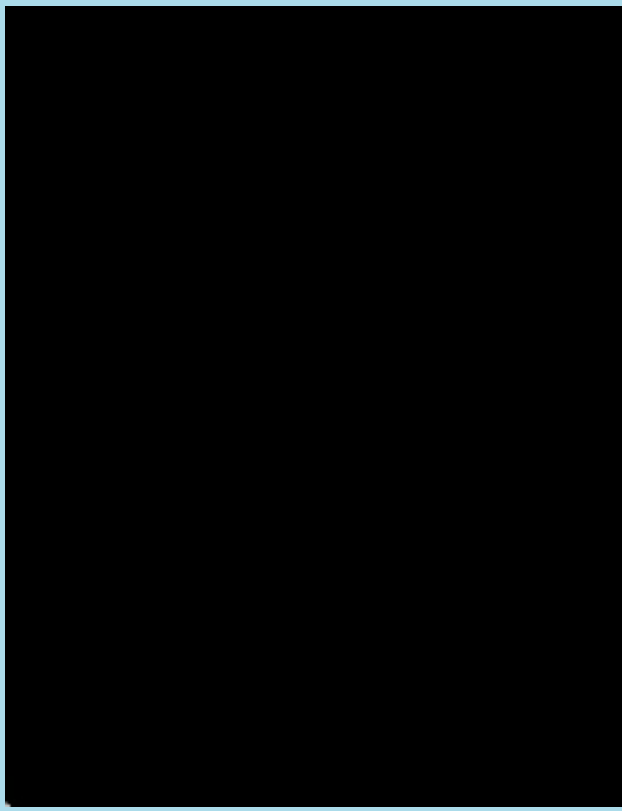
Utfordringer

- [ ] Kan du kombinere to og to av `if`-setningene ved å bruke `||` `x > 10)`, hvis x er under 1 *eller* x er over 10.

- [ ] Kan du få ballen til å sprette sideleng som dette?



- [ ] Kan du få ballen til å endre form samtidig?



- [ ] Kan du få ballen til å endre fart avhengig av posisjonen?