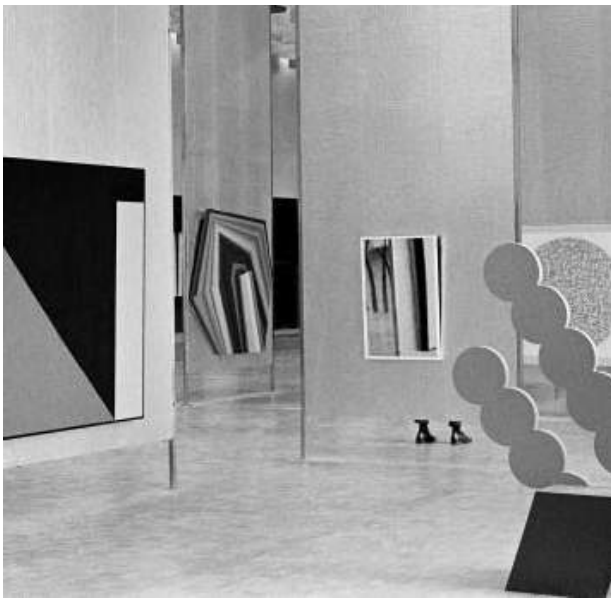
Week 2: The Field(1968) / The Field Revisited(2018)



What on earth is this spacey exhibition?
The short answer is that *The Field Revisited* recreates *The Field* exhibition(1968) for its fifty-year anniversary. *The Field* was the first comprehensive display of colour field painting and abstract sculpture in Australia. This was the inaugural and somewhat controversial exhibition in the St Kilda Road premises.

The long answer, we're going to leave up to our wonderful guides when we get out of our chairs, away from the whiteboards and go off-piste to the National Gallery of Australia.

Please meet at:
Russell St Ext entrance,
NGV Australia
8th August 2018,
9:45am (for a 10am start) - 11am
https://goo.gl/maps/Lzh1pWKhwPn

This is a private, pre-paid guided exhibition tour that has been organised as part of your coursework. You will be able to attend free of charge so please be prompt and remember you will be representing the RMIT School of Design.

“ For many, it seemed odd to open the Gallery’s new premises with such a narrowly focused exhibition, particularly for those who were accustomed to the prevailing style of figuration. Some artists were unhappy that they were not included and felt the inaugural exhibition should have instead highlighted the achievements of Australian art history. ”

– *National Gallery of Victoria*

“ Expecting certain things to happen in the work of art + finding different things actually happening, the natural response is to dispose of the new work as ‘empty’ and ‘meaningless’. These accusations have been levelled at the work of the painted and sculptors in this exhibition. ”

– *Clement Greenberg*

“Every work, however, does demand from the watcher his participation. He participates not through trying to read the artist’s state of mind or to make the shapes, forms and configurations into familiar images… the discovery that art can provide aesthetic experience as itself and nothing else only widens the scope of art”

– *Clement Greenberg*

Fun Facts

- The inaugural exhibition was deliberately curated as the beginning of the National Gallery of Victoria's commitment to be "a living gallery" - a gallery that represented current artists and contemporary movements.

- Only 3/74 artworks were already part of the NGV Collection, the others were sourced over the last 3 years from various institutions and collections

- Eighteen of the exhibiting artists in *The Field* were under the age of thirty, with Robert Hunter the youngest at twenty-one

- A notable omission from *The Field* was women artists. Five decades later, acutely aware of this imbalance, there is a display of colour field and abstract works by Australian female artists featured on Level Two at NGV Australia.

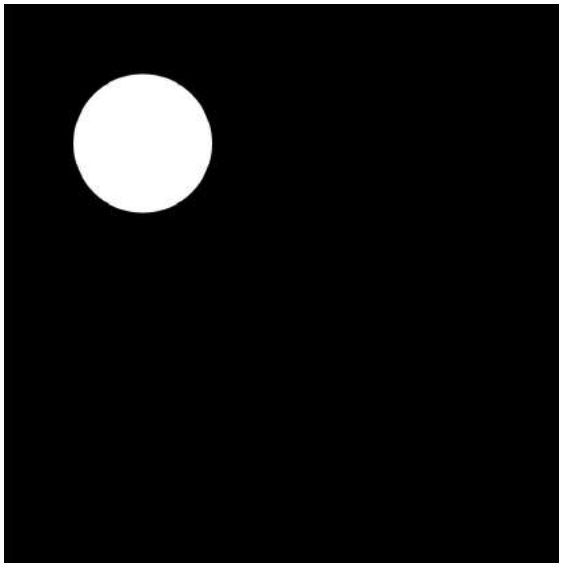
- The idea of silver foil cover walls came from Andy Warhol at The Factory in New York at the time. This is authentically recreated as a crucial detail for *The Field Revisited* from the stylings of *The Field*.

The Missing Works

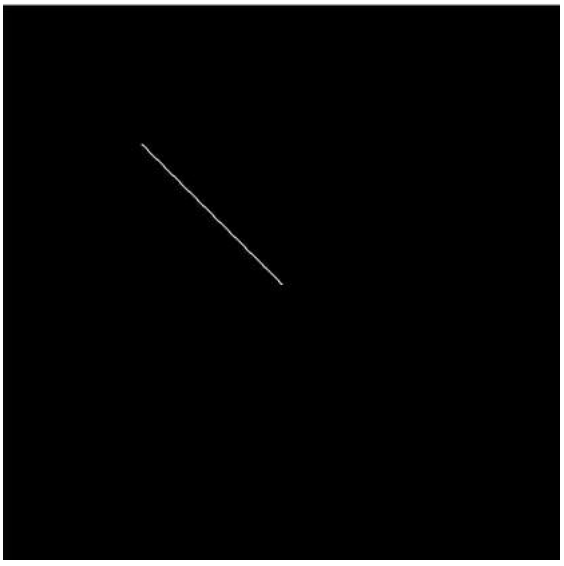
This week, to get deep into our coding exercises (and to not ruin the excitement too much of what we'll be seeing next week), we're going to look at three of the missing works in *The Field Revisited* to get us thinking and looking at the works programmatically.

Shapes glorious shapes!

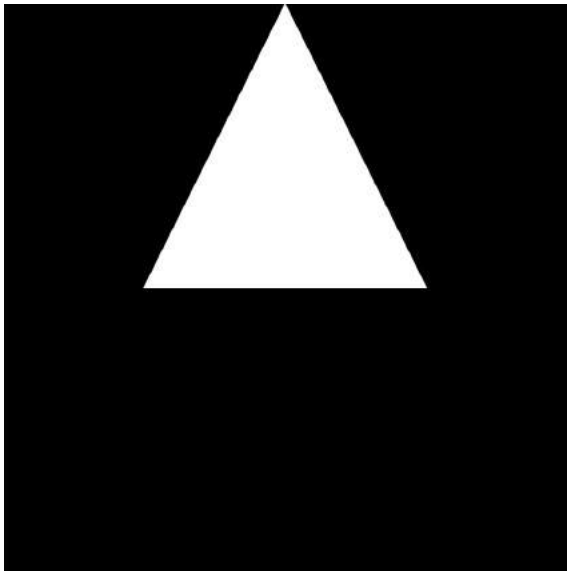
There are SO many different shapes we can create.
With your experience drawing rectangles from last week, give some of these 2D primitives a whirl! If you get stuck, just punch in random numbers and spot the difference! With very little knowledge of coding, you can still visually discern what the numbers and parameters may stand for.



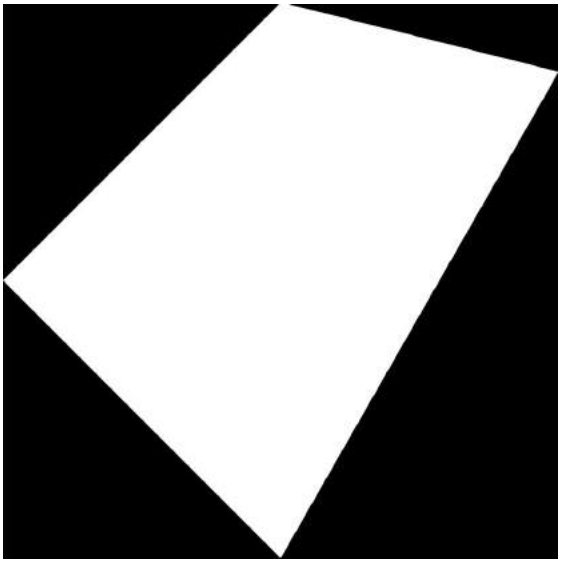
```
ellipse(x, y, width, height);
```



```
line(x1, y1, x2, y2);
```

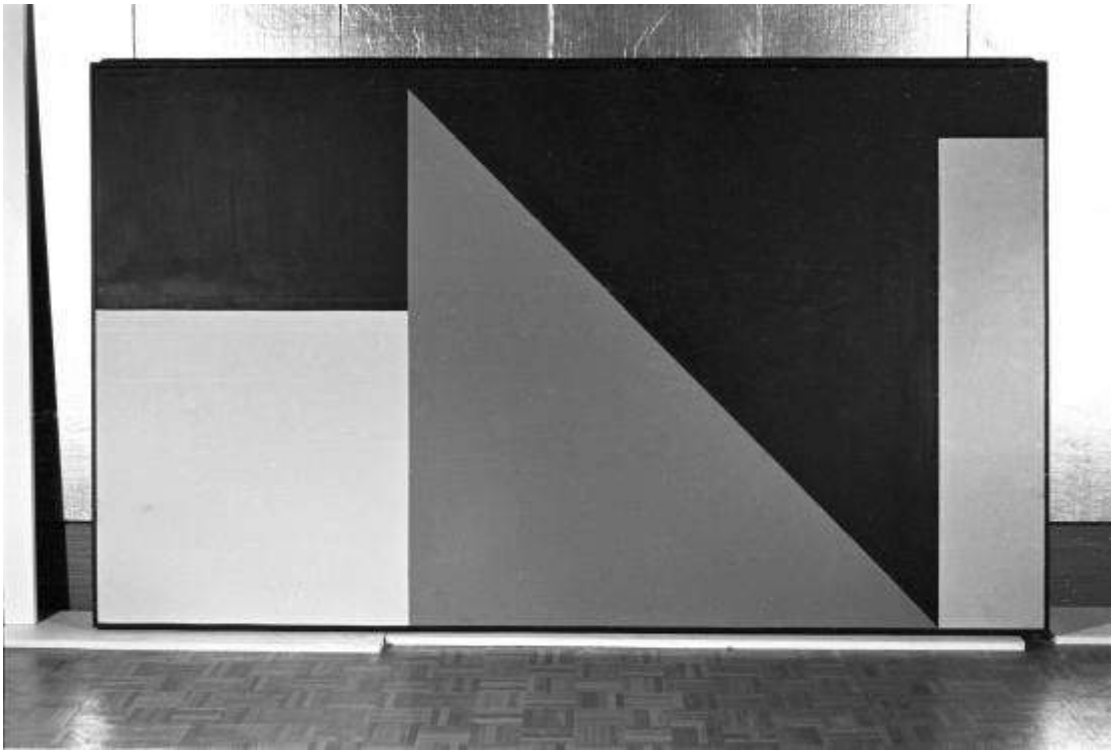


```
triangle(x1, y1, x2, y2, x3, y3);
```



```
quad(x1, y1, x2, y2, x3, y3, x4,
y4);
```

Even though some works still remain missing, they are represented in the exhibition as black and white prints. This gives us the opportunity to have an image of the how the original flow of the exhibition may have felt. We can imagine the colours the artist may have used or even better, what colours we'd use!



Peter Booth
England born 1940, Australia from 1958

Untitled painting
1968
synthetic polymer paint on canvas
Collection unknown, facsimile of missing work

Have a go making a composition similar to Peter Booth's *Untitled Painting*.

1. What colours would you use?
2. How would compose the shapes differently?
3. What different shapes would you add?

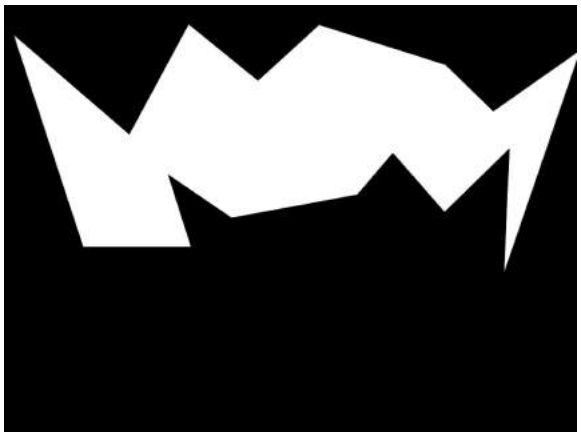
Sometimes it helps to draw out what you are trying to make, and map out the coordinates manually. Otherwise, if you want to freestyle - get the shapes onto the canvas and start playing around with the parameters.

Make your own wacky shape!

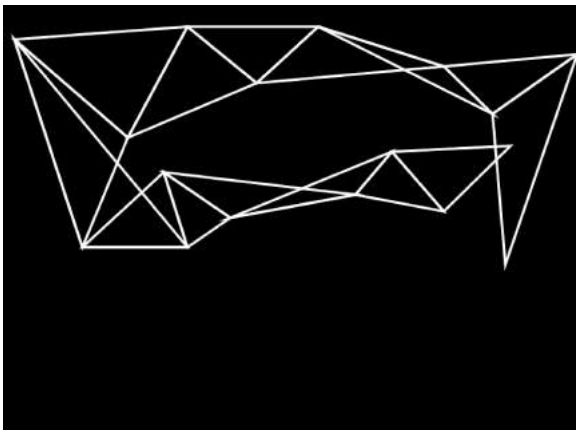
Let's dive a bit deeper!
Standard shapes are alright, but what if you want to make a crazy shape?!
Introducing `beginShape();` and `endShape();` ...

Sydney Ball
Australia 1933–2017

Zanzan 1968
synthetic polymer paint on canvas
188.5 x 158.0 cm
Collection unknown



```
beginShape();
  vertex(x1,y1);
  vertex(x2,y2);
  vertex(x3,y3);
endShape(CLOSE);
```



```
beginShape(TRIANGLE_STRIP);
  vertex(x1,y1);
  vertex(x2,y2);
  vertex(x3,y3);
endShape(CLOSE);
```

```
beginShape();
  vertex(x1,y1);
  vertex(x2,y2);
  vertex(x3,y3);
  beginContour();
    vertex(x1,y1);
    vertex(x2,y2);
    vertex(x3,y3);
  endContour();
endShape(CLOSE);
```

Try drawing a shape inspired by Sydney Ball's ZanZan.
If you're feeling adventurous - have a go at creating a negative shape within your shape with the `beginContour();` and `endContour();` functions.

NB: The vertex of shapes goes clockwise, the vertex of a contour goes counter-clockwise.

Swervy curves

This is *The Field* selection *Krimhilde* however, we will use a similar example of Christmann's work for our exercises.

Gunter Christmann
Germany born 1936 – Australia 2013, Australia from 1959

Krimhilde
1967
synthetic polymer paint polyvinyl acetate on canvas
Collection unknown

It is believed that the canvas for *Krimhilde* was reused by Gunter Christmann, and now exists on the back of another painting.

Gunter Christmann
Germany born 1936 – Australia 2013, Australia from 1959

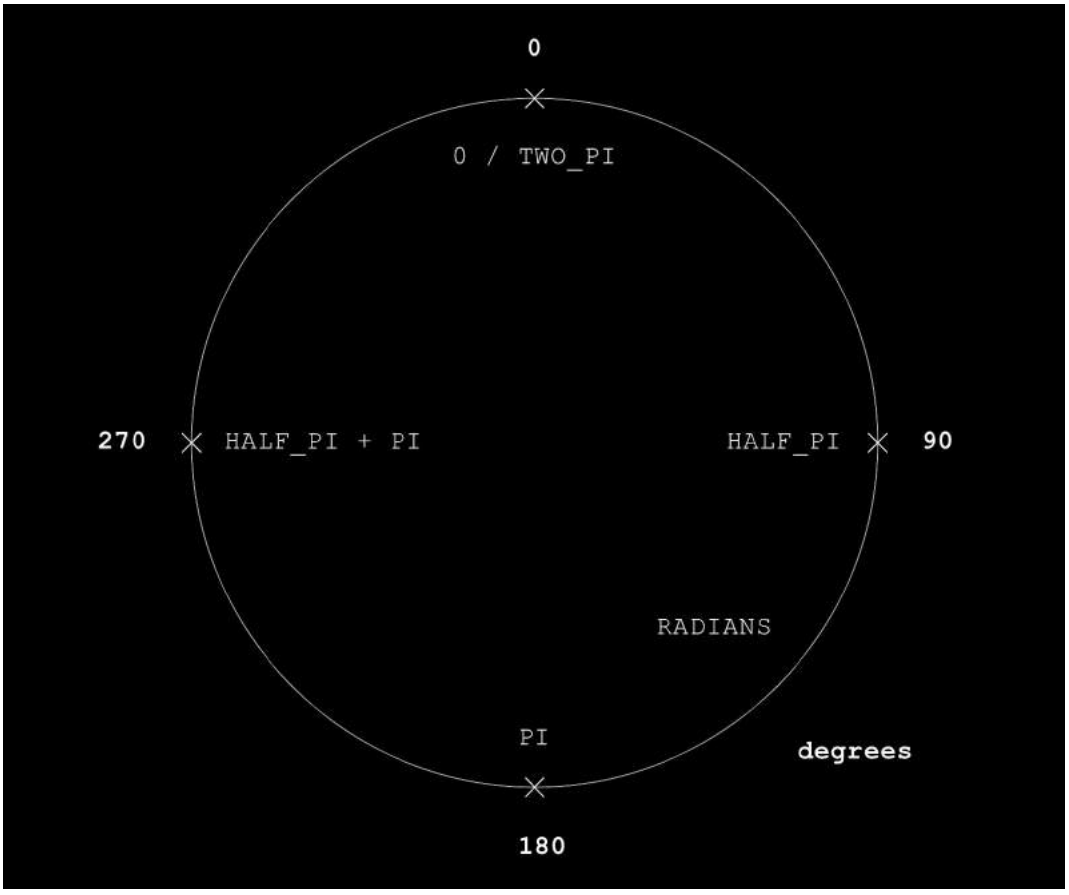
Folklore 1967/95
acrylic on canvas triptych: each panel 61 x 61cm
Private Collection Melbourne

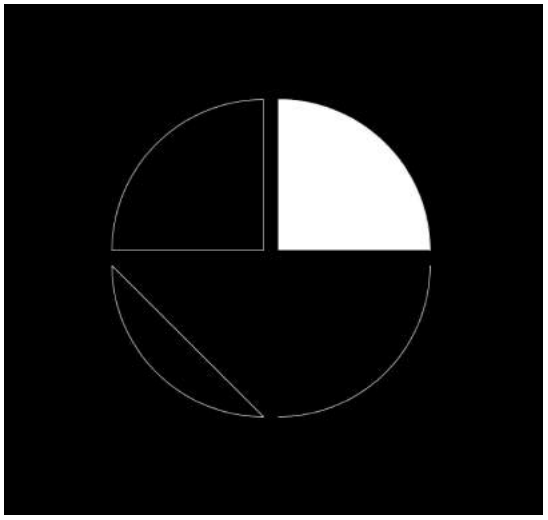
Now let's look at one of Gunter Christmann's *Folklore* pieces.

For the top ellipse, you already have the syntax to draw a black ellipse and a black rectangle on top to create the shape. However, for the lower ellipse, you'd have to draw a black ellipse followed by a black rectangle overlapping. This would then be followed by a blue rectangle to cover the excess of the black ellipse.

Alternatively, you can draw arcs.

Radians/Degrees cheatsheet



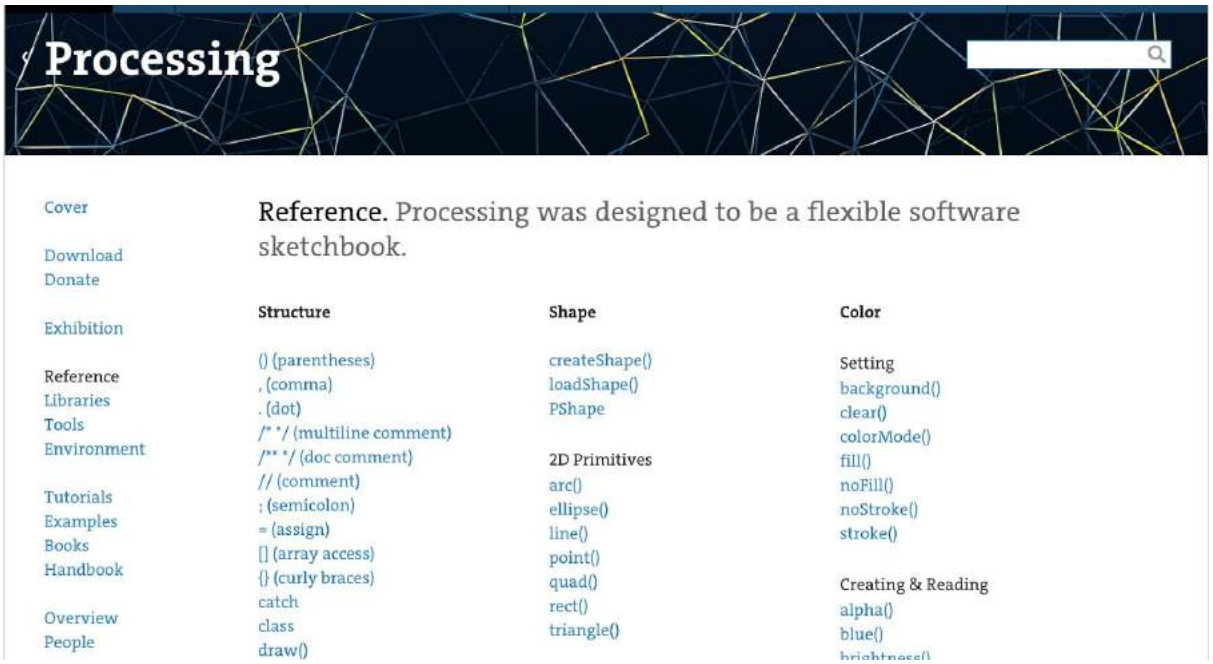


```
arc(x, y, width, height, start,
stop, mode);

...start,stop,mode);
// Bottom right:
...0, HALF_PI, OPEN);
// Bottom left:
...HALF_PI, PI, CHORD);
// Top left
...PI, HALF_PI+PI, PIE);
// Top right:
...HALF_PI+PI, TWO_PI, CLOSE);
```

Have a look at Christmann's triptych and have a go at one of the images.
Tip: If you want to use degrees instead of radians, use the `radians();` function. Ie. Instead of using `HALF_PI`, we can use `radians(180);` This converts a degrees value into a radians value(which is required for `arc()`)

Using the Processing reference



https://processing.org/reference

You may have noticed that with every shape, there are so many different parameters that you can access in the default Processing library. It's almost impossible to learn EVERY function and EVERY parameter for those functions so a part of any programmer's day is checking the reference docs! Processing is made specifically for visual artists and designers, so don't stress, most of the function references are very accessible. Let's practise.

`ellipseMode();`
You may have noticed that when you draw a rectangle - the `x` and `y` values are by default, the top left corner of the rectangle.

> **Can you get the ellipse x and y parameters to tell the computer to draw from the top left corner?**

`beginShape(mode);` / `endShape(mode);`
We've explored `beginShape(TRIANGLE_STRIP);` and `endShape(CLOSE);`

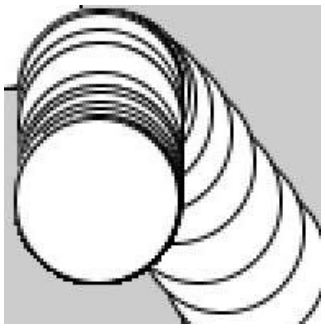
> **Try a different parameter for each function and see how things change.**
> Hint, some changes are more obvious with only a fill() or stroke().

These are our base layers.
Now let's add something less static.

Up to this point we have been “hard coding” values into our program. This means when we draw a `rect(10,10,100,100);`, our program is drawing a rectangle on top of itself again and again and AGAIN. Boring!

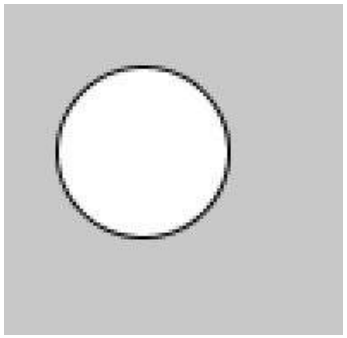
By replacing the value with a variable and incrementing/decrementing that variable each time our program calls the function `draw()`, we can start to add movement to our image.

Before we dive into what a “variable” can be - let’s try this:



```
void draw(){
  ellipse(mouseX, mouseY, 50, 50);
}
```

Now we have some crazy drawing tool that draws the current position of your mouse’s x-coordinate and y-coordinate. If we want it to appear like the ellipse is moving, we draw the background at the start of the `draw()` loop.



```
void draw(){
  background(200);
  ellipse(mouseX, mouseY, 50, 50);
}
```

`mouseX` and `mouseY` are examples of a system variable. This is a variable that Processing understands already. But of course, we can make our own variables!

DIY Variables

First off, we need to talk about data types. In order for the computer to understand what kind of value it's dealing with, you need to tell the computer what "type" of variable you're going to be giving it. The main ones you'll be dealing with at this early stage are:

float: Floating-point numbers, e.g. numbers that have a decimal point.
int: Integers, whole numbers aka numbers without a decimal point
char: Characters, typographic symbols such as A, d, and $. A char stores letters and symbols in the Unicode format(a computer standard for the consistent encoding, representation, and handling of text)

String: A string is a sequence of characters.
color - You guessed it, a datatype for handling colours. Note the lack of "u" in colour (this is a common type error when your code doesn't run).

You will notice that majority of the shape parameters accept the "float" data type. Most of the time, you will need to make sure when doing calculations that you aren't adding a String to a float for example - this will either not work or come up with something strange.

For example, this will not work:

```
float x = 2.342;
int y = 1;
int p = x + y;
println(p);

"Cannot convert from float to int".
```

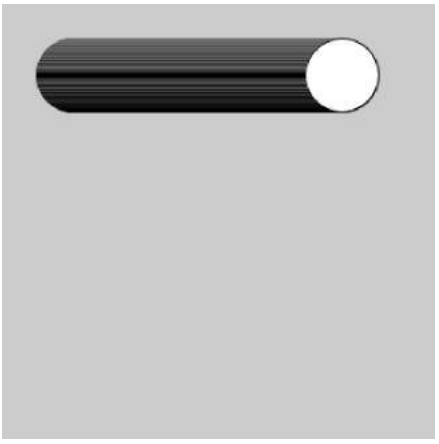
However, this will work…

```
float x = 2.342;
int y = 1;
float p = x + y;
println(p);
```

But enough theory, we'll learn by doing!

Global variables and local variables

Let's get our trusty circle back. We want what's called a "global" variable that will increase each time the function `draw()` is called. This will create what I term the "smudge" effect.

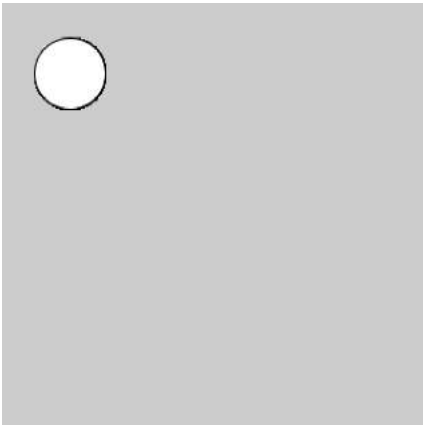


```
float x = 50.0;

void setup(){
  size(300,300);
}

void draw(){
  ellipse(x, 50, 50, 50);
  //Increment variable x by 1
  x++;
}
```

A local variable is contained within a function and is only available in this function. Therefore, in the below program, the function `setup();` does not know about `x`. Let's see what happens when you use a local variable...



```
void setup(){
  size(300,300);
}

void draw(){
  float x = 50.0;
  ellipse(x, 50, 50, 50);
  //Decrement variable x by 1
  x--;
}
```

Nothing! This is because everytime you call the function `draw();` it sets x to 50.0 and decrements it by 1. Repeat.

Local variables will be useful soon. However, for now, make sure you're declaring your variables at the top of your program, outside of any function.

Make a global variable called 's' and use it to increment or decrement the size of your ellipse!

Week 2 Exercise

Looking at The Field Revisited's missing works as your base layer, create a sketch of moving shapes inspired by one of the works.

Using at least 3 different 2D primitives, use incrementing and/or decrementing variables to make the shapes dance.

Have fun with it!!
Maybe you want to use random();
Or maybe instead of always incrementing by 1 - you want to increment by 17.32*y ?!

Punch in some maths like you used to punch in parameters. The world is your oyster.