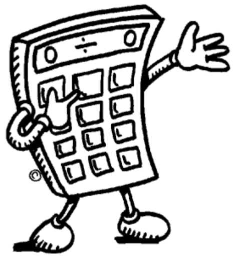
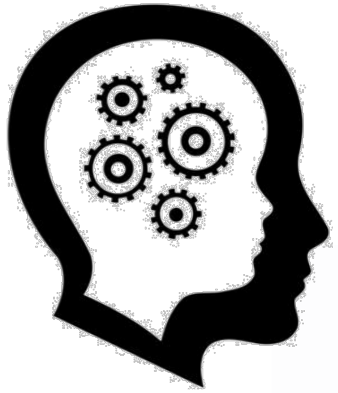
Getting Started with Processing

#2016ABbday

What is computer science?

What is computer science?

Solving problems!

Computational Thinking

GETTING STARTED WITH PROCESSING

My Only Rule:

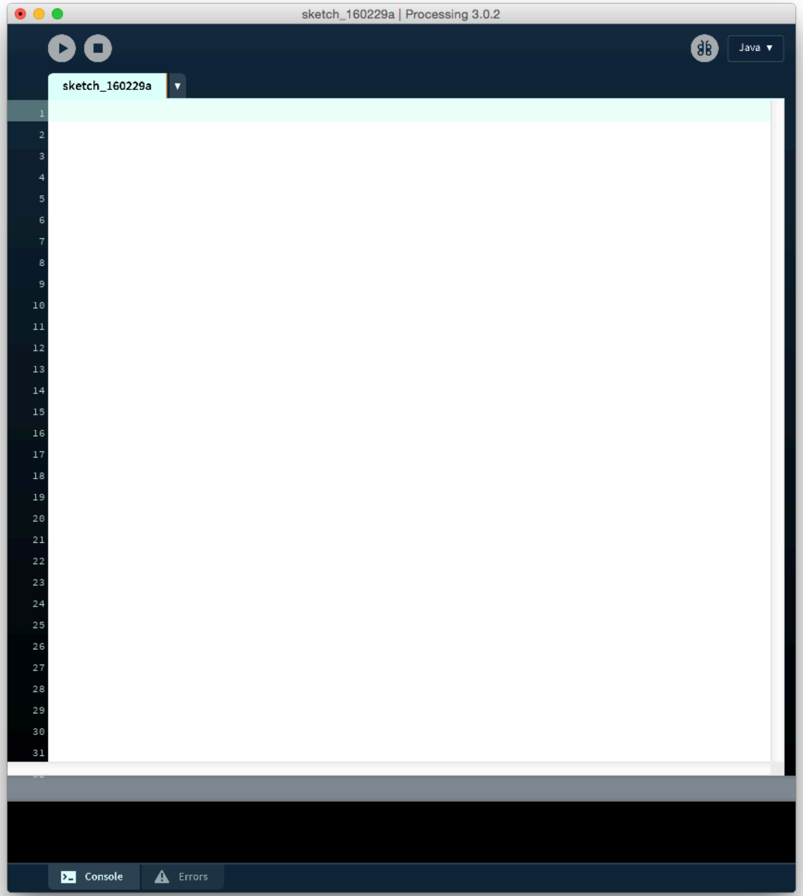
When I interrupt to teach the next thing,
everyone should get really, really quiet. ;)

Resources Available on Event Page!

https://www.eventbrite.com/e/anita-borg-birthday-celebration-tickets-21382259915

Wifi:
Shopify Guests
welcome2shopify

Open the Processing App

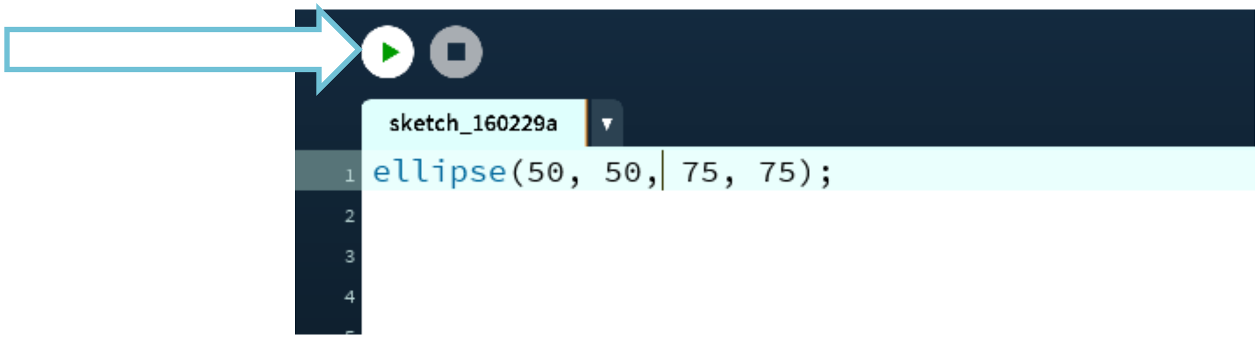


Forgot to download? Go to https://processing.org/download/?processing

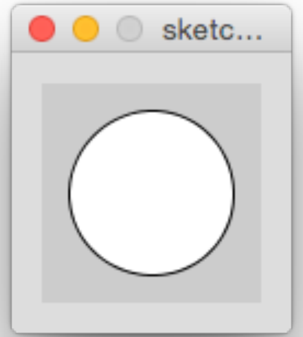
Type this in the new window:

```
ellipse(50, 50, 75, 75);
```

Run Your Program!



```
ellipse(50, 50, 75, 75);
```



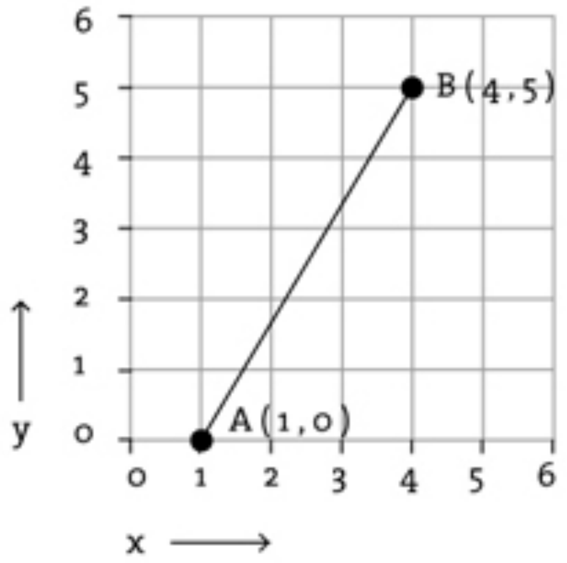
Play around with the numbers in the brackets.
What happens?

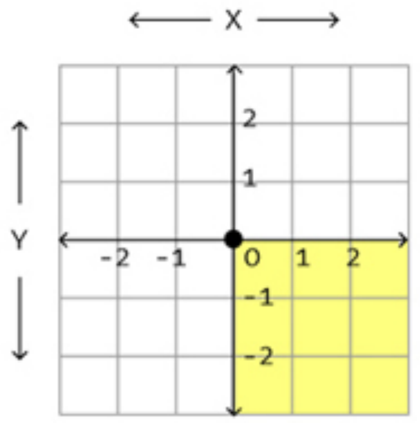
Can you draw a square?

A line?

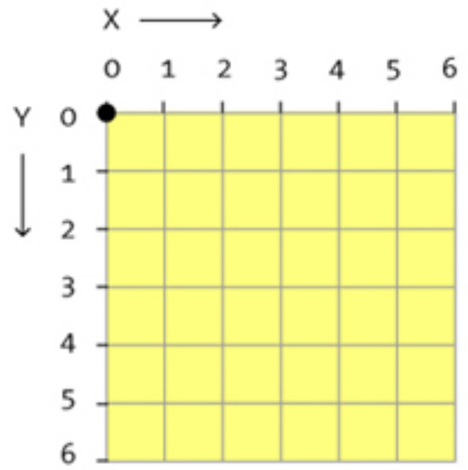
Super challenge: A pentagon?

Hint: Use the Processing reference at http://processing.org/reference

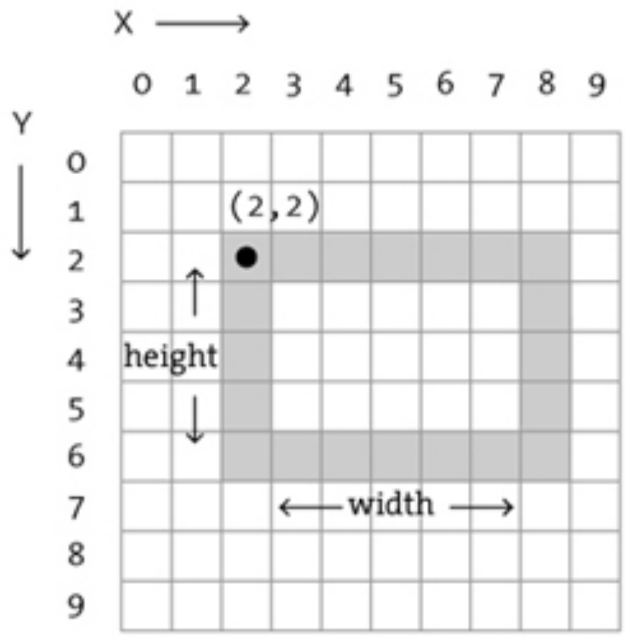

B(4,5)
A(1,0)
y
x
0
1
2
3
4
5
6



Eighth Grade



Computer



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

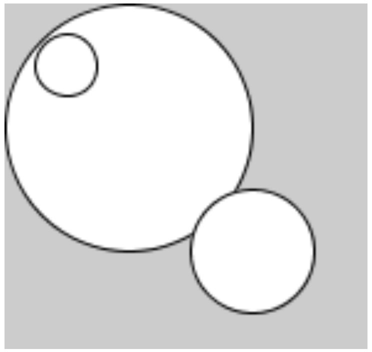
Example:

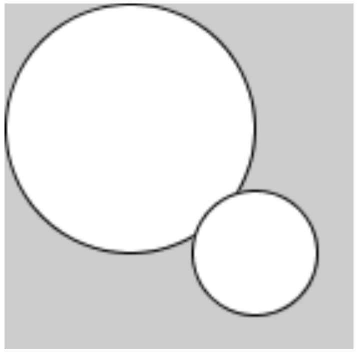
```
rect(2,2,7,5);
```

Challenge Question!

What picture will the following code make?

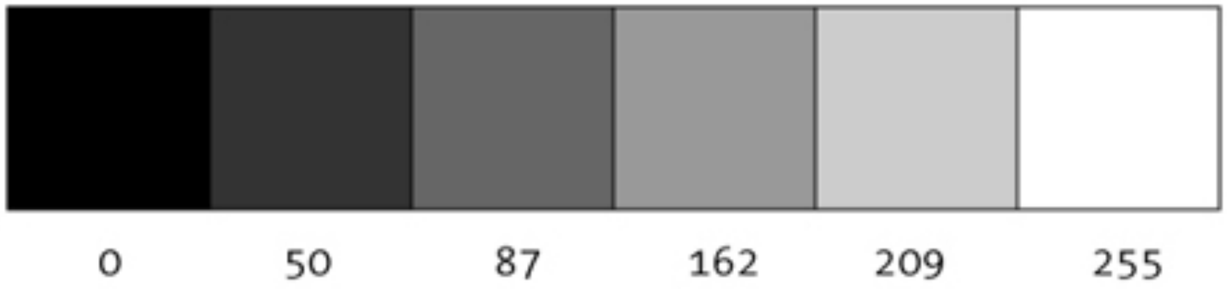
```
size(130,130);
ellipseMode(CENTER);
ellipse(25, 25, 25, 25);
ellipse(50, 50, 100, 100);
ellipse(100, 100, 50, 50);
```

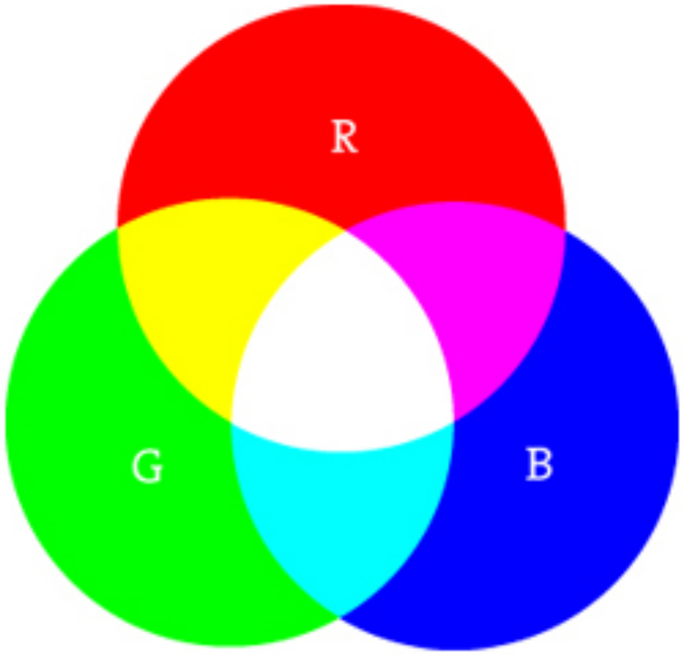


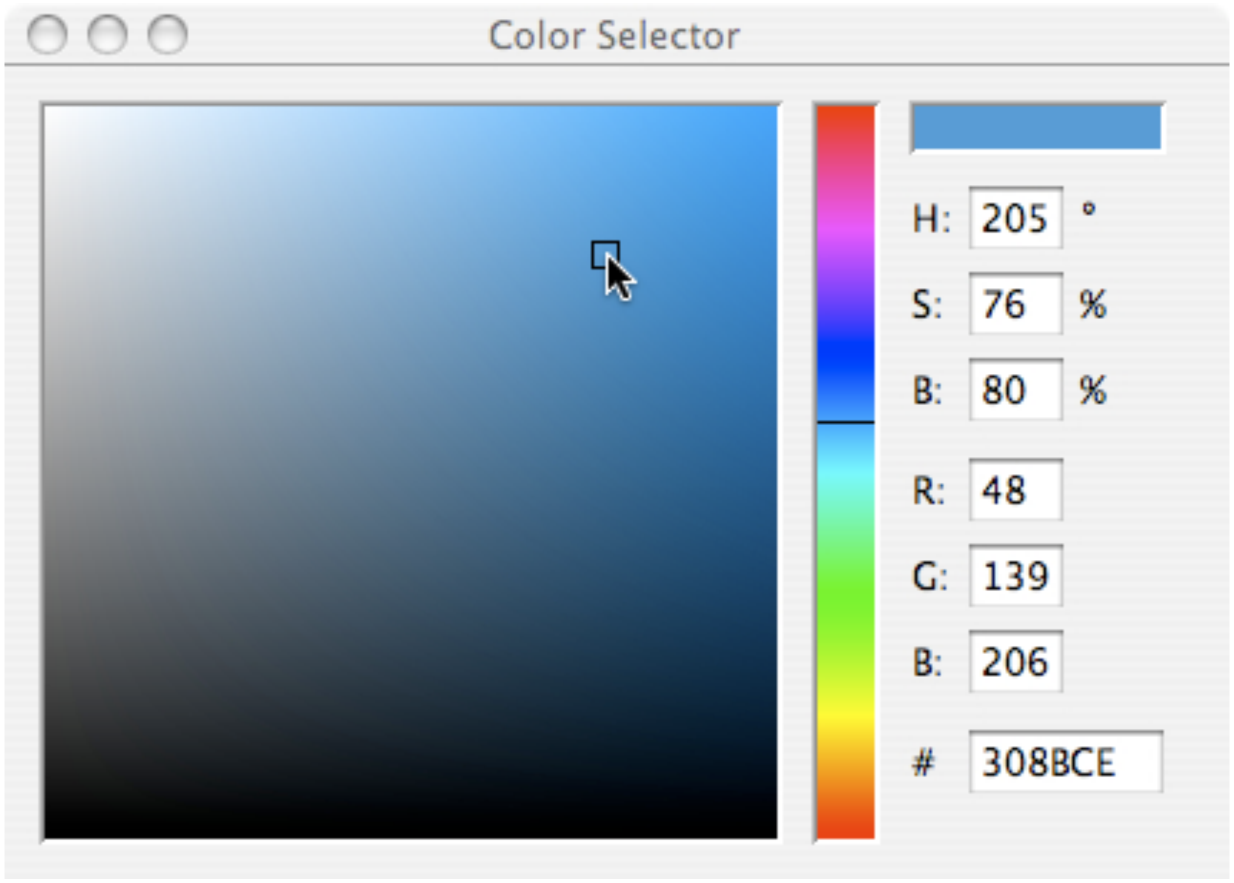


Neither

COLOUR


0
50
87
162
209
255


R
G
B



H: 205 °

S: 76 %

B: 80 %

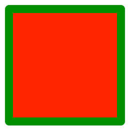
R: 48

G: 139

B: 206

308BCE

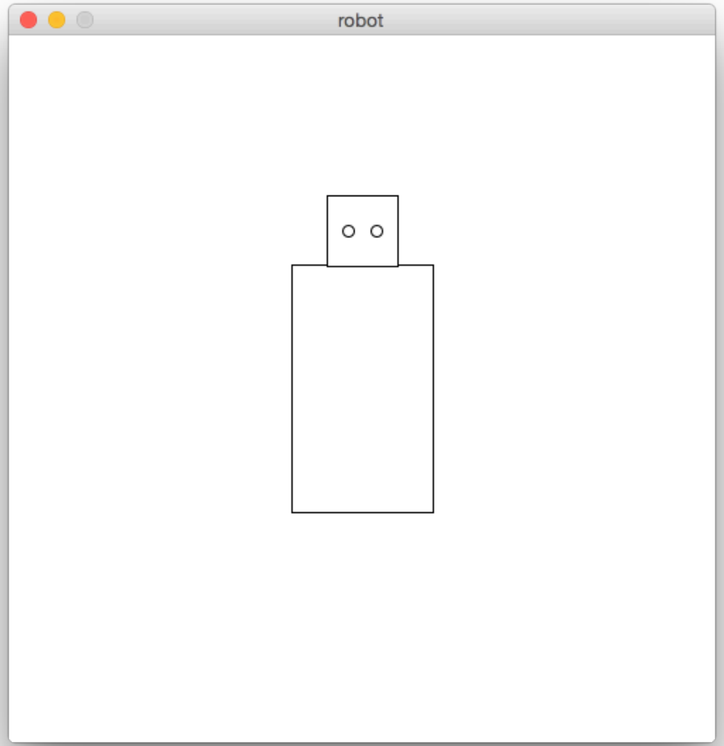
Can you modify your code to make a red square with a green outline?



Hint: Use the Processing reference at http://processing.org/reference

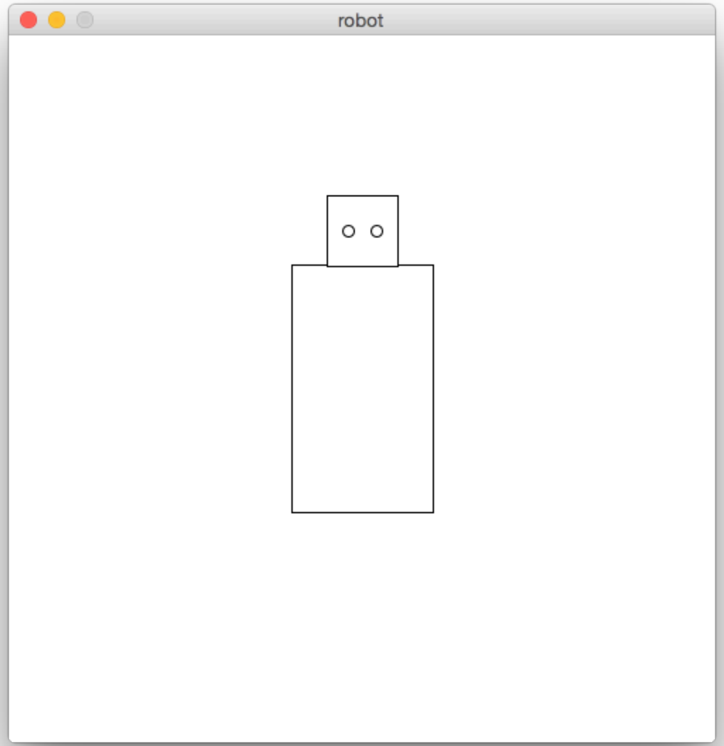
DRAWING MORE COMPLEX PICTURES

Let's Draw Something Harder



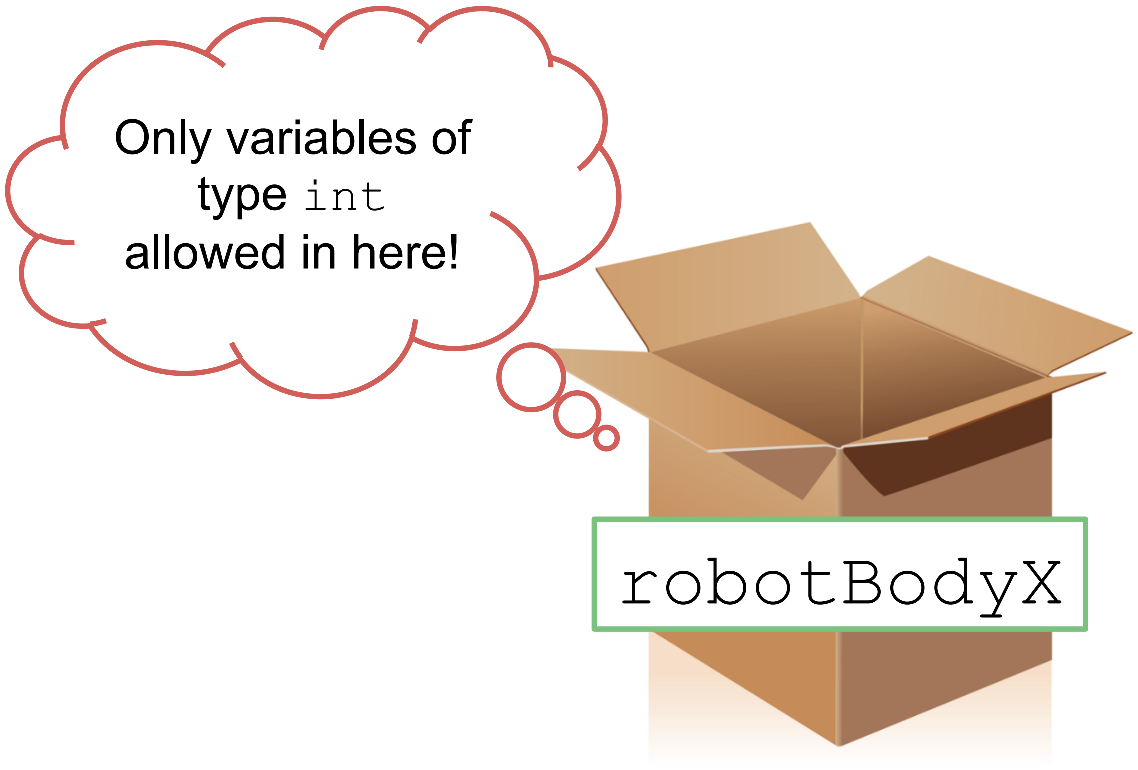
```
// Starter code

size(500,500);
background(255);

rectMode(CENTER);
rect(500/2, 500/2, 100, 175);
```

Let’s Draw Something Harder



What if we needed to start drawing the robot somewhere other than the center?

PAINFUL

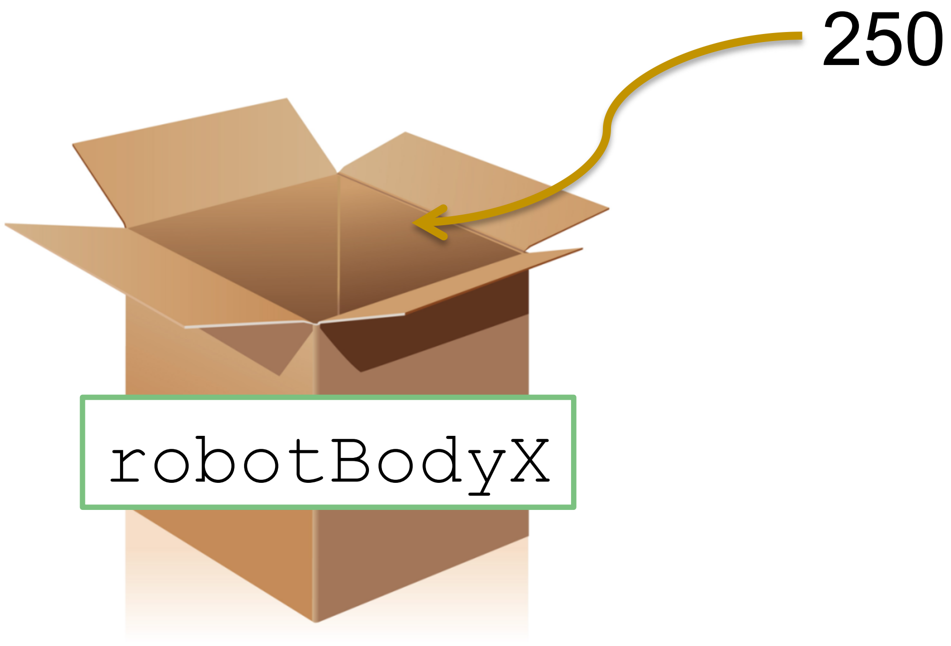


Variable type

```
int robotBodyX;
```

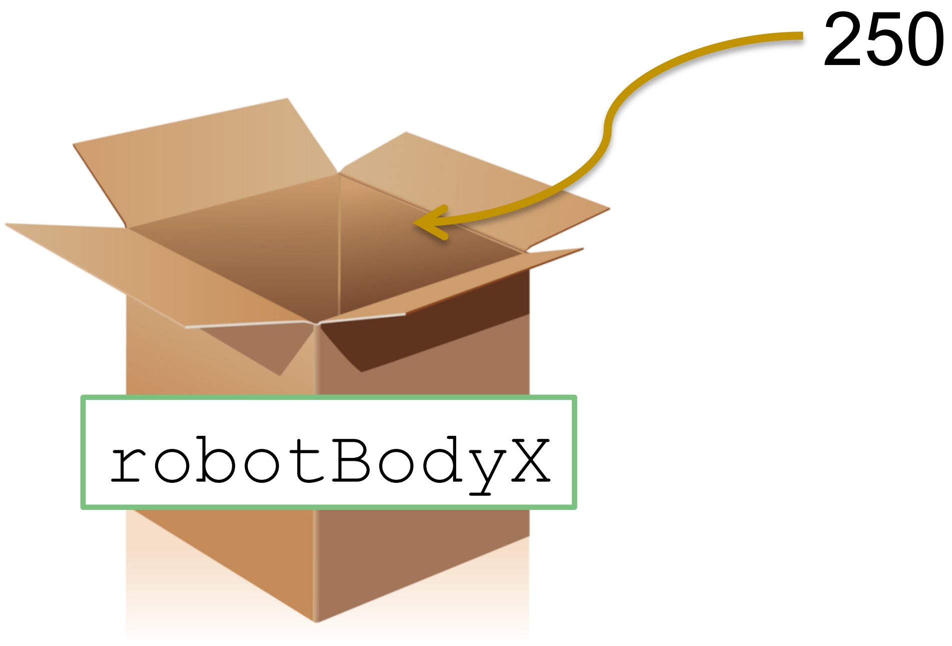
Variable declaration!

Data Type	Values
boolean	true/false
byte	generic 8 bits of data
char	character (‘a’, ‘b’, …)
color	a grayscale or RGB color
double	floating point with double precision
float	floating point (number with a decimal point)
int	integer (whole number)
long	really big integer



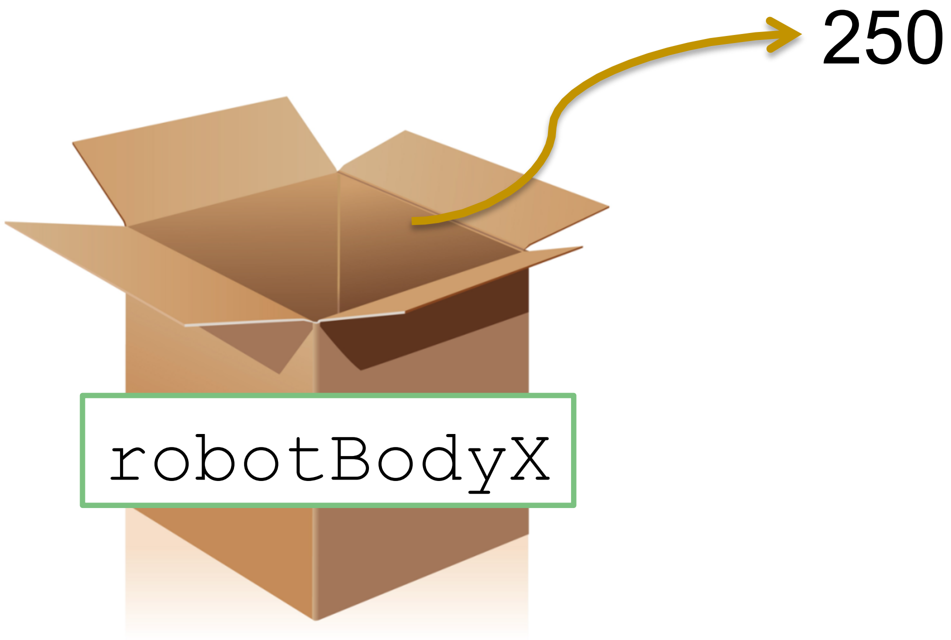
```
robotBodyX = 250;
```

Variable assignment!



```
int robotBodyX = 250;
```

Variable declaration AND assignment!



```
rect(robotBodyX, robotBodyY, 100, 175);
```

Using the variable's value

```
// Variables!

int robotBodyX = width/2;
int robotBodyY = height/2;
int robotBodyWidth = 100;
int robotBodyHeight = 175;

int robotHeadWidth = 50;
int robotHeadHeight = 50;

// Drawing...

rectMode(CENTER);
rect(robotBodyX,
     robotBodyY,
     robotBodyWidth, robotBodyHeight);
rect(robotBodyX,
     robotBodyY - robotBodyHeight/2 - robotHeadWidth/2,
     robotHeadWidth, robotHeadHeight);

ellipseMode(CENTER);
ellipse(robotBodyX - 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
ellipse(robotBodyX + 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
```

```
// Variables!

int robotBodyX = width/2;
int robotBodyY = height/2;
int robotBodyWidth = 100;
int robotBodyHeight = 175;

int robotHeadWidth = 50;
int robotHeadHeight = 50;

// Drawing...

rectMode(CENTER);
rect(robotBodyX,
     robotBodyY,
     robotBodyWidth, robotBodyHeight);
rect(robotBodyX,
     robotBodyY - robotBodyHeight/2 - robotHeadWidth/2,
     robotHeadWidth, robotHeadHeight);

ellipseMode(CENTER);
ellipse(robotBodyX - 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
ellipse(robotBodyX + 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
```

Declare and assign variables describing robot's body

```
// Variables!

int robotBodyX = width/2;
int robotBodyY = height/2;
int robotBodyWidth = 100;
int robotBodyHeight = 175;

int robotHeadWidth = 50;
int robotHeadHeight = 50;

// Drawing...

rectMode(CENTER);
rect(robotBodyX,
     robotBodyY,
     robotBodyWidth, robotBodyHeight);
rect(robotBodyX,
     robotBodyY - robotBodyHeight/2 - robotHeadWidth/2,
     robotHeadWidth, robotHeadHeight);

ellipseMode(CENTER);
ellipse(robotBodyX - 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
ellipse(robotBodyX + 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
```

Use the values stored in the variables to draw the body

```
// Variables!

int robotBodyX = width/2;
int robotBodyY = height/2;
int robotBodyWidth = 100;
int robotBodyHeight = 175;

int robotHeadWidth = 50;
int robotHeadHeight = 50;

// Drawing...

rectMode(CENTER);
rect(robotBodyX,
     robotBodyY,
     robotBodyWidth, robotBodyHeight);
rect(robotBodyX,
     robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
     robotHeadWidth, robotHeadHeight);

ellipseMode(CENTER);
ellipse(robotBodyX - 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
ellipse(robotBodyX + 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
```

Position the robot's head relative to the robot's body

```
// Variables!

int robotBodyX = width/2;
int robotBodyY = height/2;
int robotBodyWidth = 100;
int robotBodyHeight = 175;

int robotHeadWidth = 50;
int robotHeadHeight = 50;

// Drawing...

rectMode(CENTER);
rect(robotBodyX,
     robotBodyY,
     robotBodyWidth, robotBodyHeight);
rect(robotBodyX,
     robotBodyY - robotBodyHeight/2 - robotHeadWidth/2,
     robotHeadWidth, robotHeadHeight);

ellipseMode(CENTER);
ellipse(robotBodyX - 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
ellipse(robotBodyX + 10,
        robotBodyY - robotBodyHeight/2 - robotHeadHeight/2,
        8, 8);
```

Try changing variable values to see what happens:

- body position
- head position
- body size
- head size

What are some advantages of using variables?

Makes the code easier to read.

You can adjust numbers in only one place.

Challenge Question!

What color will the circle be?

```
color blueColor = color(0,0,255);
color redColor = color(255,0,0);
blueColor = redColor;
redColor = blueColor;
fill(redColor);
ellipse(50,50,75,75);
```

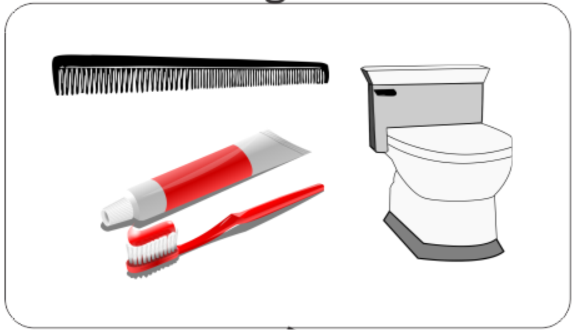
🔴 **red**

🔵 **blue**

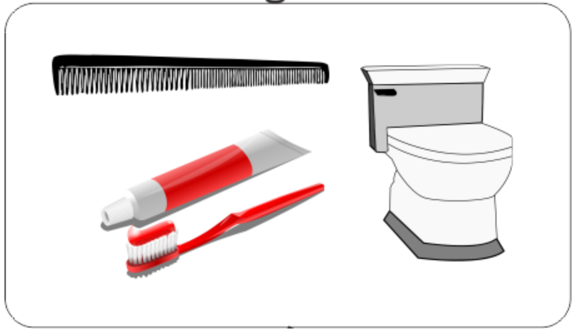
Neither
(syntax error)

PICTURES IN MOTION

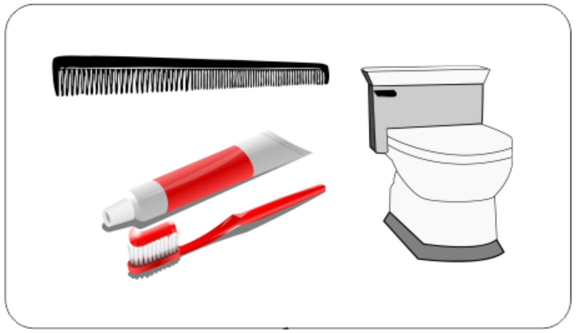
morningRoutine

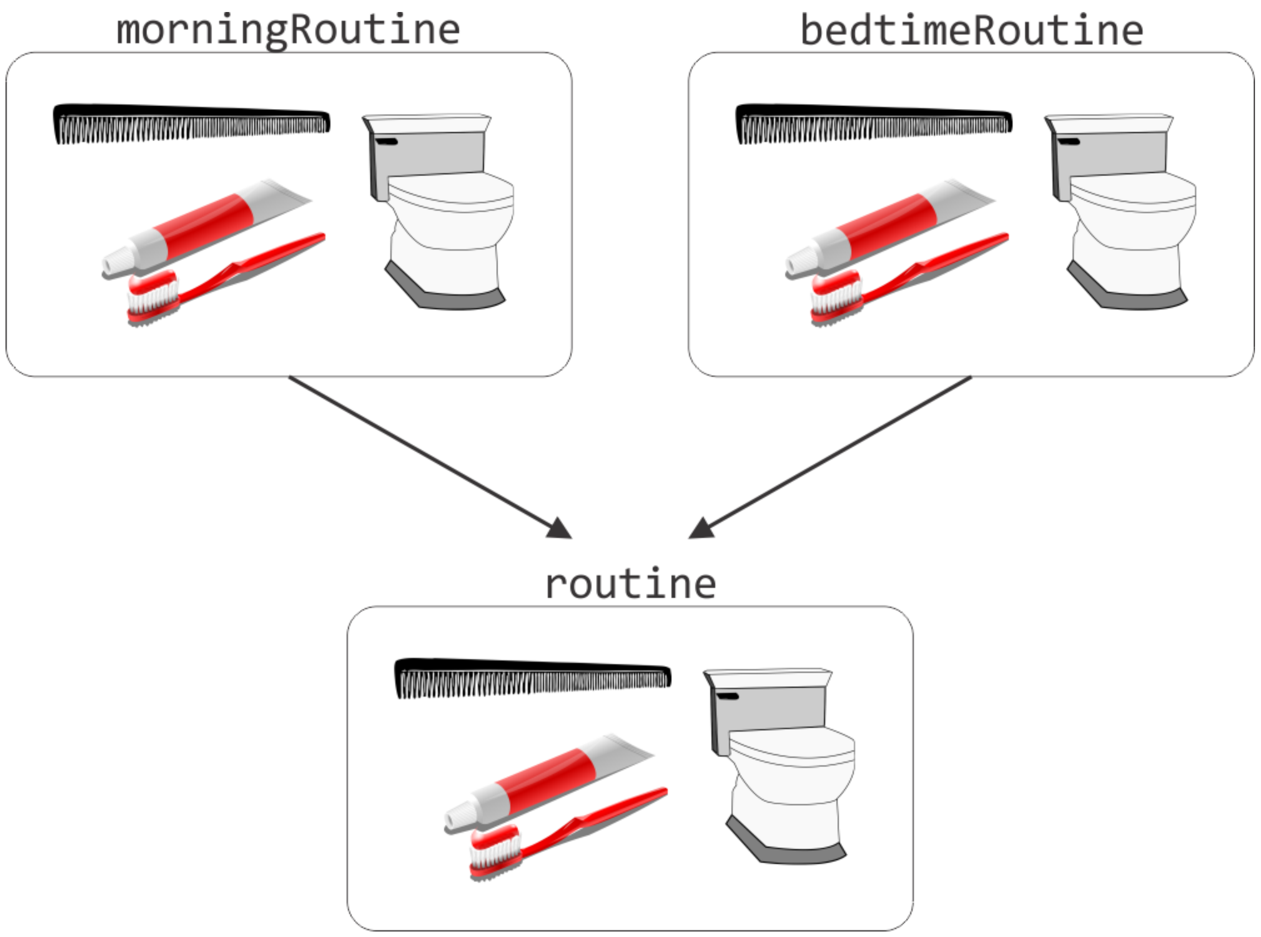


morningRoutine

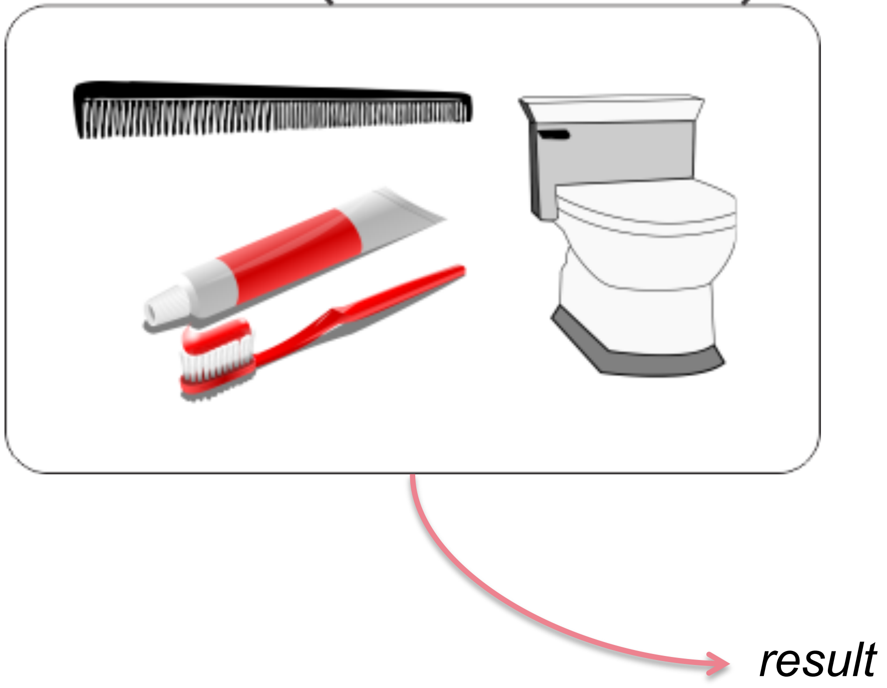


bedtimeRoutine




morningRoutine
bedtimeRoutine
routine

`routine(doThisFirst)`



Functions

fancy term for routine!

```
ellipse(…)
line(…)
background(…)
color(…)
noFill()
```

Functions

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, no result is returned)
}
```

Functions

function name

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, nothing returned)
}
```

Functions

parameter list

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, nothing returned)
}
```

Functions

parameter type

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, nothing returned)
}
```

Functions

parameter name

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, nothing returned)
}
```

Functions

```
void ellipse(float x, float y, float width, float height)
{
   // code in here that does something
   // (in this case, nothing returned)
}
```

function body

Functions

return type

```
void ellipse(float x, float y, float width, float height)
{
    // code in here that does something
    // (in this case, nothing returned)
}
```

Functions

return type

```
color color(int red, int green, int blue)
{
  // creates and returns color data type
}
```

Functions

empty parameter list

```
void noFill()
{
  // does some stuff to turn off fill
}
```

Active Mode in Processing

```
void setup()
{
  // Runs once at the beginning of program
}

void draw()
{
  // Runs once every frame
}
```

```
void setup()
{
  size(500,500);
  background(0);
}

void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

```
void setup()
{
  size(500,500);
  background(0);
}
```

Run the setup routine once when the program starts

```
void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

```
void setup()
{
  size(500,500);
  background(0);
}

void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

Set the window size and set the background to black exactly once

```
void setup()
{
  size(500,500);
  background(0);
}

void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

Run the draw routine once every frame of the animation

```
void setup()
{
  size(500,500);
  background(0);
}

void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

Draw new lines all the time, adding to existing ones

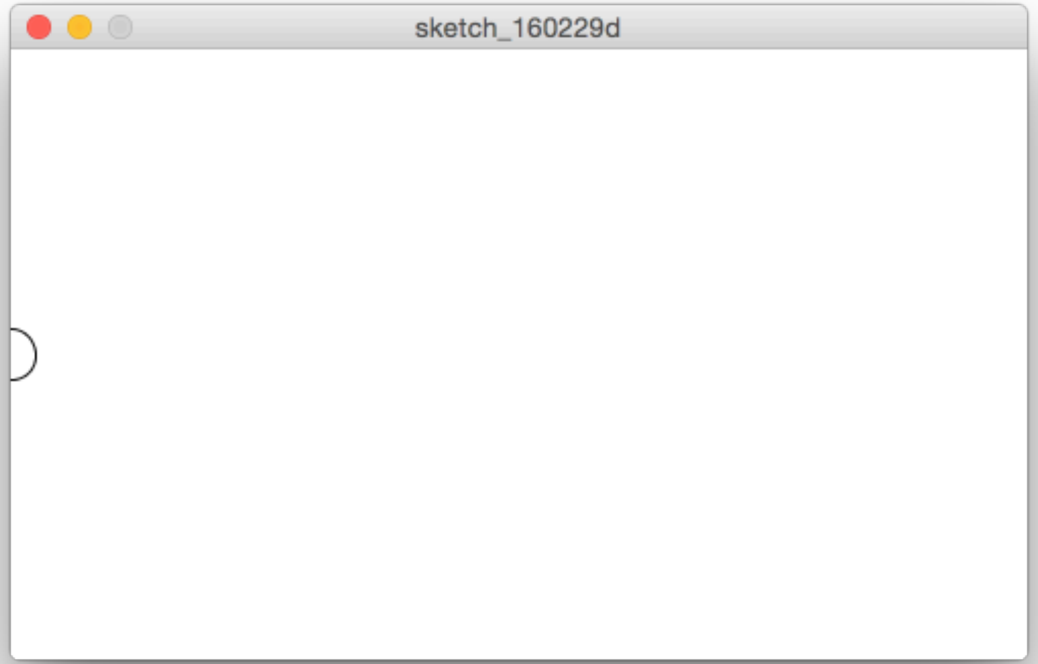
Try modifying your code to get a different effect (change colours, shapes, the order of the lines of code…).

```
void setup()
{
  size(500,500);
  background(0);
}

void draw()
{
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```

Challenge Question!

What is the result of modifying the code as follows?

```
void setup()
{
  size(500,500);
}

void draw()
{
  background(0);
  color lineColor = color(mouseX, mouseY, mouseX+mouseY);
  stroke(lineColor);
  line(0, 0, mouseX, mouseY);
}
```


sketch_160229d

Ball Animation Starter Code

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Ball Animation Starter Code

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Set up some useful variables

Ball Animation Starter Code

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Set the size of the window exactly once

Ball Animation Starter Code

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Clear the background every frame

Ball Animation Starter Code

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Draw a circle using its variables every frame

Challenge Project!

Make your ball move across the screen!

Hints:

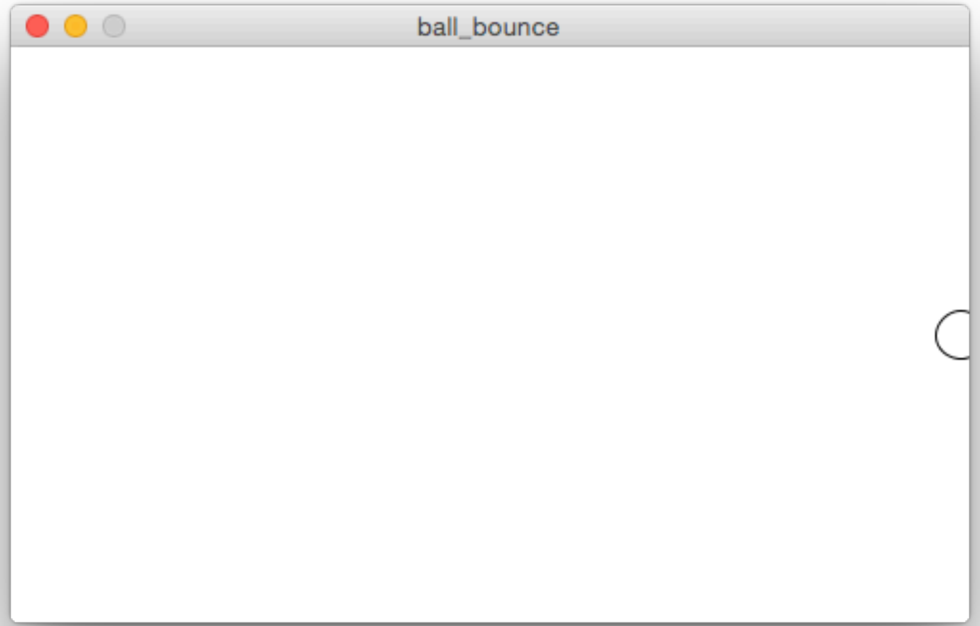
- Notice what the name *variable* implies about what can happen to its values.
- Think about what values change over time.
- Think about *when* a value needs to change, and by how much.

Challenge Question!

What would happen if we "commented out" the call to `background` in `draw`?

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

void setup()
{
  size(500,300);
}

void draw()
{
  //background(255);
  circleX += 5;
  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

MAKING DECISIONS



What if we wanted the ball to bounce back when it hit the right-hand side?

If Statements

Does the expression evaluate to true?

YES:
Run the code in the body.

NO:
Skip past the if statement.

if	It is raining
then	Wear a raincoat

if	Grade is at least 50
then	Pass the class

and

Everything has to be true

and

true and true = true

true and false and true = false

false and false = false

or

Only one thing has to be true

or

true or **true** = **true**

true or **false** or **true** = **true**

false or **false** = **false**

not

Flips a Boolean value

not

not **true** = **false**

not **false** = **true**

not

not (**a** and **b**) = (not **a**) or (not **b**)

not (**a** or **b**) = (not **a**) and (not **b**)

$\uparrow$

de Morgan's Law

```
if not married and not engaged and like
her:
    should put a ring on it
```

$$\downarrow$$

```
if not (married or engaged) and like her:
    should put a ring on it
```

if	I am buying a movie ticket *and* I am a student
then	I will get a discount on the price

if	My percentage is at least 77 *and* my percentage is at most 79
then	My grade is B+

if	The battery is dead *or* there is no gas
then	The car will not start

Logical Operators in Processing

and: `&&`
or: `||`
not: `!`

equals: `==`
less than: `<`
less than or equal: `<=`
greater than: `>`
greater than or equal: `>=`

Challenge Question!

What colour will the ellipse be?

```
fill(0,0,255);

boolean b = true;
if (true && (!b || false))
{
  fill(255,0,0);
}

ellipse(width/2, height/2, width, height);
```

If-Else Statements

Does the expression evaluate to true?

YES:
Run the code in the body.

NO:
Run the code in the else.

Else-If Statements

Does the expression evaluate to true?

YES:
Run the code in the body.

NO:
Check the next expression.

Does the expression evaluate to true?

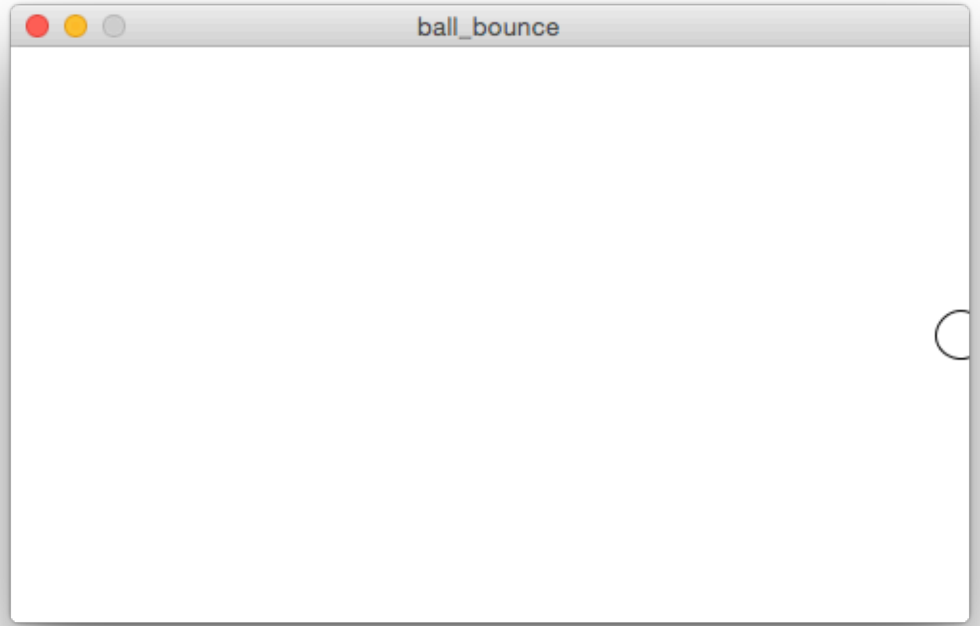
YES:
Run the code in the else-if body.

NO:
Run the code in the else, or done.

Challenge Question!

```
void setup()
{
  size(500,100);
  fill(255,0,0);
}

void draw()
{
  background(255);

  if (mouseX < width/3)
  {
    rect(0, 0, width/3, height);
  }
  else if (mouseX >= width/3 && mouseX < width*2/3)
  {
    rect(width/3, 0, width/3, height);
  }
  else
  {
    rect(width*2/3, 0, width/3, height);
  }
}
```

What does this code do?



What if we wanted the ball to bounce back when it hit the right-hand side?

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circleX < 0)
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circleX < 0)
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

If the circle's x-position gets too big…

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circleX < 0)
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

…set the speed to a negative number.

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circleX < 0)
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Otherwise, if the x-position is too small…

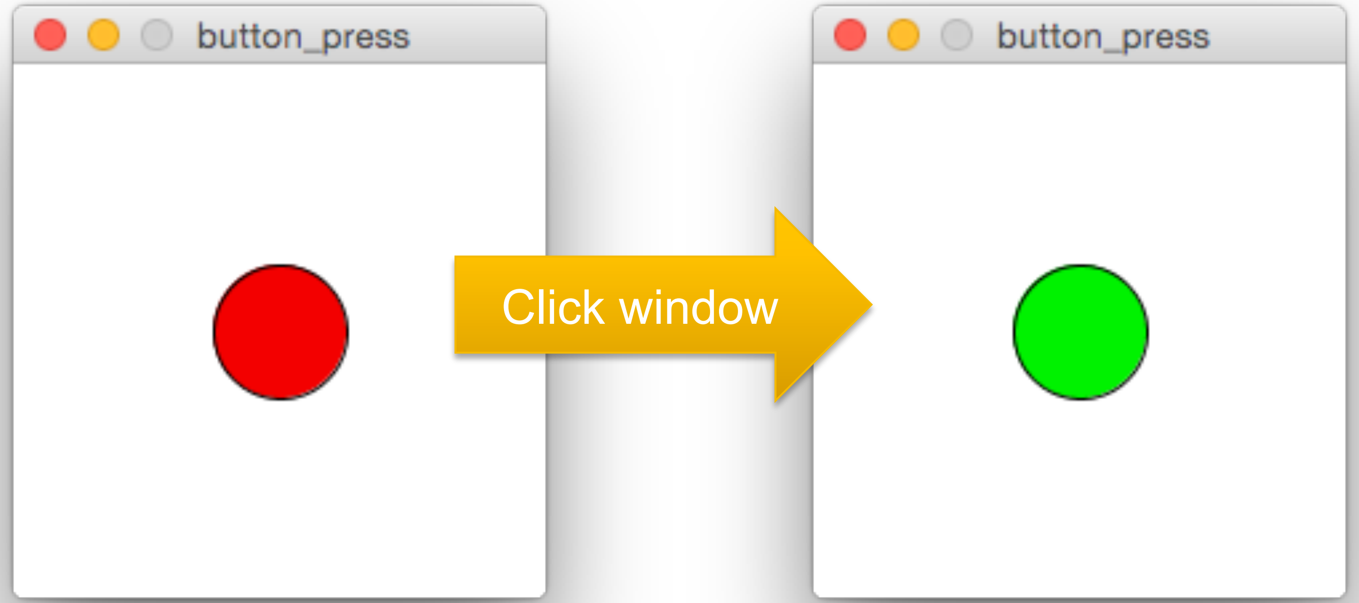
```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circl
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

…then the speed must have been negative, so switch back to positive

```
int circleX = 0;
int circleY = 150;
int circleWidth = 25;
int circleHeight = 25;

int speed = 5;

void setup()
{
  size(500,300);
}

void draw()
{
  background(255);
  circleX += speed;
  if (circleX > width)
  {
    speed = -5;
  }
  else if (circleX < 0)
  {
    speed = 5;
  }

  ellipse(circleX, circleY, circleWidth, circleHeight);
}
```

Otherwise, don't change the speed at all.

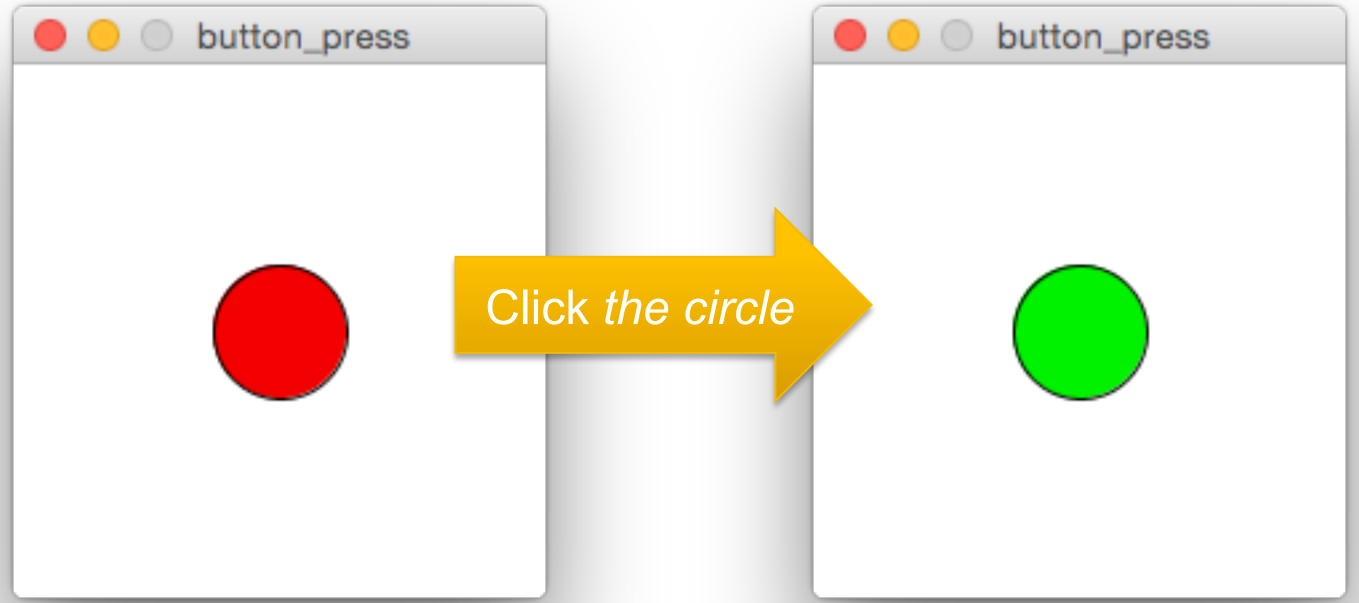
Challenge Project!



Hint: Look up the `mouseClicked` function.

Hint: Keep track of what colour the circle is currently in a variable.

Challenge Project Extension!



Hint: Use if-statements to decide whether the mouse is inside the circle whenever the mouse is clicked.