CS1103 Workshop 2

1. VARIABLE
2. CONDITION

OPERATORS, SYSTEM VARIABLES, GLOBAL AND LOCAL VARIABLES, IF, IF-ELSE, ELSE-IF, CONDITIONS

CS1103 Workshop 2

1. VARIABLE
2. CONDITION

OPERATORS, SYSTEM VARIABLES, GLOBAL AND LOCAL VARIABLES, IF, IF-ELSE, ELSE-IF, CONDITIONS

RECAP

```
var x = 10;
```

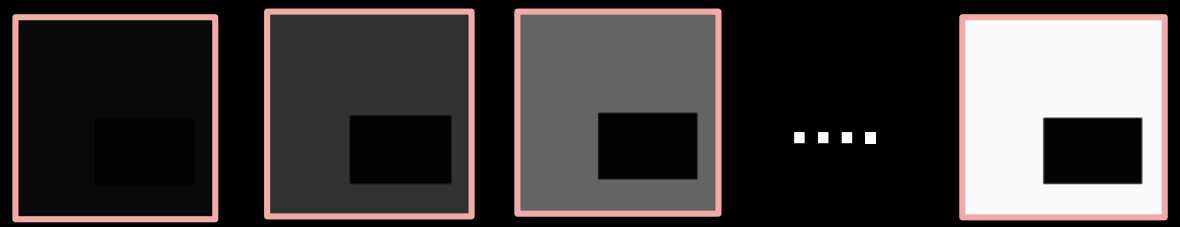
variable

```
function setup() {
  createCanvas(300, 300);
  fill(0); //set filling color to black
}
```

```
function draw() {
  background(x);
  rect(120, 150, 150, 100);
  x = x + 1;
}
```

setup() – Set up the basic building blocks for a program
Example: Canvas

draw() – sketching on canvas continuously over frames



comments:

comments, like *//set filling color to black*, **is ignored and not executed by the computer. We often add comments to remind ourselves or inform others about what the code does.**

VARIABLE

- Variables store information in our programs and provides a way of labeling data with a descriptive name
- It is helpful to think of variables as containers that hold information
- The sole purpose is to label and store data in the computer memory. This data can then be used throughout your program

VARIABLE

```
var x = 10;

function setup() {
  createCanvas(300, 300);
  fill(0); //set filling color to black
}

function draw() {
  background(x);
  rect(120, 150, 150, 100);
  x = x + 1;
}
```

x | 10

The value 10 is stored inside computer memory and labeled as **x**

We call **x** as a variable

Grammatically, we state "**var**" before **x** to make **x** a variable

VARIABLE

```
var x = 10;

function setup() {
  createCanvas(300, 300);
  fill(0); //set filling color to black
}

function draw() {
  background(x);
  rect(120, 150, 150, 100);
  x = x + 1;
}
```

x 10

Paint the background of canvas with a color value.

The value of **x**, 10, is used.

VARIABLE

```
var x = 10;

function setup() {
  createCanvas(300, 300);
  fill(0); //set filling color to black
}

function draw() {
  background(x);
  rect(120, 150, 150, 100);
  x = x + 1;
}
```

The value **x** is increased by 1

VARIABLE

```
var x = 10;

function setup() {
  createCanvas(300, 300);
  fill(0); //set filling color to black
}

function draw() {
  background(x);
  rect(120, 150, 150, 100);
  x = x + 1;
}
```

x 11

In the next frame …

the background value of canvas is set to 11

And the value of x will keep on increasing in the subsequence animations

EXERCISES

- Job – 1a
  1. Create a ball on the canvas
  2. Let the ball move horizontally from left to right

- Job – 1b
  1. Create a ball on the canvas
  2. Let the ball move vertically from bottom to top

- Job – 1c
  1. Create a ball on the canvas
  2. Let the ball move diagonally to the bottom right

```
function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(100);
  fill(250, 255, 0);
  ellipse(25, 150, 50, 50);
}
```

VARIABLE

- Variable is named as a sequence of characters that consist of letters, digits, underscores
- A variable must start with a letter or an underscore
- It cannot be started with a digit
  Also, it cannot be a reserved word, e.g., *var*, *function*, *rect*, *size, …*
- Which of the following is not correct
  - `cs1103`
  - `cs_1103`
  - `_cs1103`
  - `1103_cs`

NAMING VARIABLE (GUIDELINE)

- Always use meaningful and descriptive name for variable

- Use lowercase
  - ✓`radius`
  - ✓`area`

- If the name consists of several words, concatenate all in one, use lowercase for the first word, and capitalize the first letter of each subsequent word in the name
  - ✓`mediaComputing`
  - ✓`maxValue`

NUMERIC OPERATOR

Operators that apply on two numbers and give a result.

Operator	Meaning	Example	Result
+	Addition	`5+2`	7
-	Subtraction	`5-2`	3
*	Multiplication	`5*2`	10
/	Division	`5/2`	2.5
%	Remainder	`5%2`	1

MORE ON OPERATOR

Operators that assign a value to the left-side operand.

Operator	example	Meaning	Equivalent
=	`x = 5`	“Store” value 5 into the variable x	
	`x = x + 10`	Add value 10 to the existing value of x	
+=	`x += 10`	Add value 10 to the existing value of x	x = x + 10
++	`x++`	Increase the value of x by 1	x = x + 1
--	`x--`	Decrease the value of x by 1	x = x - 1

- Job – 2

Trace the changes of variable values in the program.
What will be the final output?

```
1  function setup() {
2    createCanvas(200, 50);
3    background(230);
4
5    var a, b, c, d;
6    a = -10;
7    a = a + 1;
8    b = 30;
9    b = a;
10   b++;
11   a -= 10;
12   c = -b;
13   d = c % 3;
14   d + 1;
15
16   text(a, 0, 20);
17   text(b, 50, 20);
18   text(c, 100, 20);
19   text(d, 150, 20);
20 }
```

Values of variables:

	a	b	c	d
after line 6.	a = -10; -10	--	--	--
after line 7.	a = a + 1;			
after line 8.	b = 30;			
after line 9.	b = a;			
after line 10.	b++;			
after line 11.	a -= 10;			
after line 12.	c = -b;			
after line 13.	d = c % 3;			
after line 14.	d + 1;			

i.e. Undefined

[Final output of the program]
The contents painted by line 16-19 are:

SYSTEM VARIABLE

- User variable
  - ✓A variable defined by programmer
- System variable
  - ✓A variable defined by system (or P5.JS)
  - ✓Note: **they are available after setup() runs**

Refer to
https://p5js.org/reference/

Example	Meaning
width	The width of canvas
height	The height of canvas
mouseX	Cursor position (x)
mouseY	Cursor position (y)
frameCount	The number of frames that have been displayed since the program starts

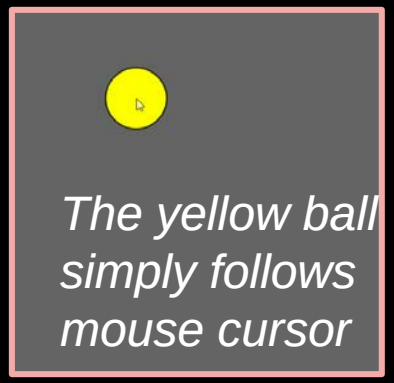
EXAMPLE

```
function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(100);
  fill(250, 255, 0);
  ellipse(width/2, height/2, 50, 50);
}
```

The values of **width** and **height** are set to 300

EXAMPLE

```
function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(100);
  fill(255, 255, 0);
  ellipse(mouseX, mouseY, 50, 50);
}
```

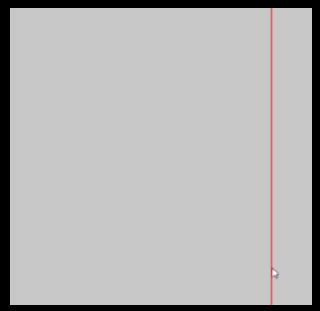


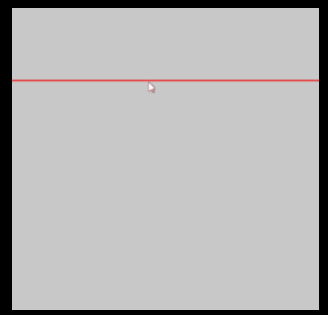
The yellow ball simply follows mouse cursor

Recall:
mouse position that is given as the values of the system variables mouseX and mouseY.

EXERCISES

- Job – 3a
  - ✓ Write a program that displays a vertical line in red color following cursor movement
- Job – 3b
  - ✓ Write a program that displays a horizontal line in red color following cursor movement





```
function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(200);
  stroke(255, 0, 0);
  line(width/2, 0, width/2, height);
}
```

GLOBAL & LOCAL VARIABLES

```
var a = 150;
var b = 150;

function setup() {
  createCanvas(300, 300);
  frameRate(10);
}

function draw() {
  var x = 150;
  var y = 150;

  background(200);
  noFill();
  rect(100, 100, 100, 100);

  //flying ball
  fill(255, 255, 0);
  x += random(-20, 20);
  y += random(-20, 20);
  ellipse(x, y, 50, 50);

  //flying box
  fill(255, 153, 0);
  a += random(-20, 20);
  b += random(-20, 20);
  rect(a-25, b-25, 50, 50);
}
```

Global variable

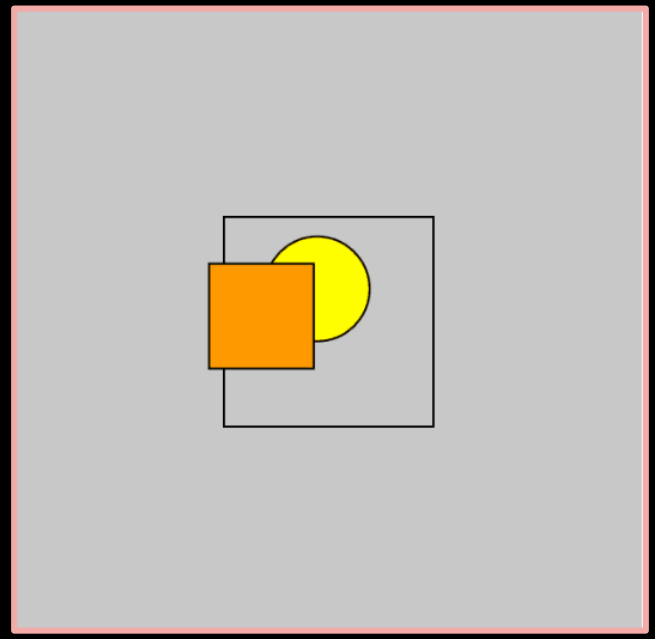
Local variable

Global variable is declared outside of functions. Its value takes effect throughout the program.

Local variable (short life-time) is declared inside a function. Its value takes effect only during the current execution of that function.

```
var a = 150;
var b = 150;

function setup() {
  createCanvas(300, 300);
  frameRate(10);
}

function draw() {
  var x = 150;
  var y = 150;

  background(200);
  noFill();
  rect(100, 100, 100, 100);

  //flying ball
  fill(255, 255, 0);
  x += random(-20, 20);
  y += random(-20, 20);
  ellipse(x, y, 50, 50);

  //flying box
  fill(255, 153, 0);
  a += random(-20, 20);
  b += random(-20, 20);
  rect(a-25, b-25, 50, 50);
}
```

GLOBAL & LOCAL VARIABLES



Why the yellow ball can never move out of the small window?

CS1103 Workshop 2

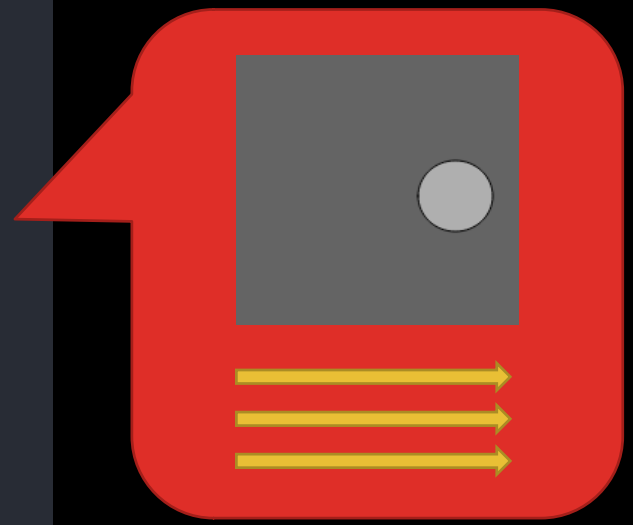
1. VARIABLE
2. CONDITION

OPERATORS, SYSTEM VARIABLES, GLOBAL AND LOCAL VARIABLES, IF, IF-ELSE, ELSE-IF, CONDITIONS

```
var x = 0;
var y = 100;

function setup() {
  createCanvas(200, 200);
  frameRate(30);  // 30 frames per second
}

function draw() {
  background(100);
  stroke(0);
  fill(175);
  ellipse(x, y, 50, 50);

  // condition
  if (x > width)  {
     x = 0;
  }
  x = x + 10;  // step forward by 10
}
```

IF STATEMENT

The ***if-then statement*** is a control flow that tells a program to execute certain portion of code if the condition is true.



Example – moving ball (comes back from the left)

```
var x = 0;
var y = 100;

function setup() {
  createCanvas(200, 200);
  frameRate(30);  // 30 frames per second
}

function draw() {
  background(100);
  stroke(0);
  fill(175);
  ellipse(x, y, 50, 50);

  // condition
  if (x > width)  {
     x = 0;
  }
  x = x + 10;  // step forward by 10
}
```

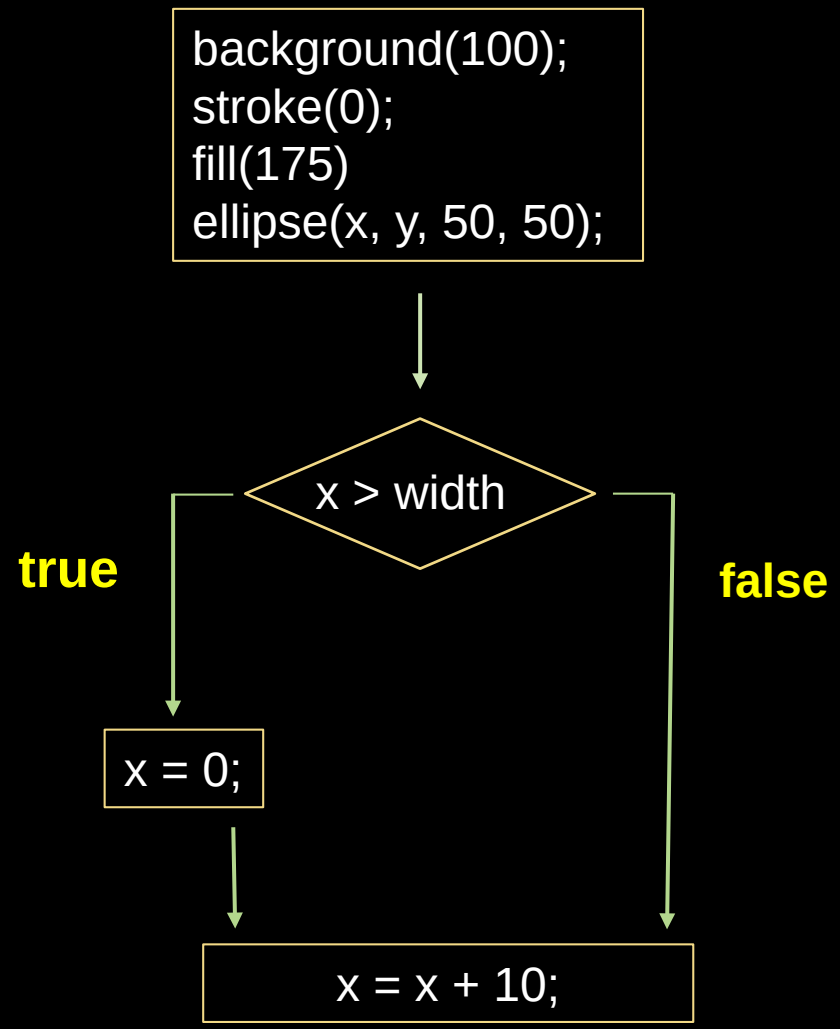
IF STATEMENT

Often in a program we need to compare two values, such as whether the variable x is greater than width

The result of comparison is a Boolean value called true or false

```
var x = 0;
var y = 100;

function setup() {
  createCanvas(200, 200);
  frameRate(30);  // 30 frames per second
}

function draw() {
  background(100);
  stroke(0);
  fill(175);
  ellipse(x, y, 50, 50);

  // condition
  if (x > width)  {
     x = 0;
  }
  x = x + 10;  // step forward by 10
}
```

IF STATEMENT



OPERATORS

Two types of operators to make comparison:

Relational and Equality operators	Meaning
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
!=	Not equal to

Logical operator	Meaning
!	not
&&	and
||	or

For example,

when x is -1 or 301, the result of **x<0 ||x>300** is true

when the mouse is in the left half of the canvas, **mouseX>=0 && mouseX<width/2** is true

IF STATEMENT

```
var x = 0;
var y = 100;

function setup() {
  createCanvas(200, 200);
  frameRate(30);  // 30 frames per second
}

function draw() {
  background(100);
  stroke(0);
  fill(175);
  ellipse(x, y, 50, 50);

  // condition
  if (x > width) {
    x = 0;
  }
  x = x + 10;  // step forward by 10
}
```

Example – moving ball (comes back from the left)

Note:

- We may add braces { } here to enclose the statements under the if-condition
- The braces { } are **mandatory** if there are 2 or more statements under the if-condition

e.g.

```
if (x > width) {
   x = 0;
   y += 25; //The ball goes down
}
```

IF STATEMENT

Parenthesis required

```
if (x > width)
    x = 0;
```

Braces required
for multiple actions
under an if clause

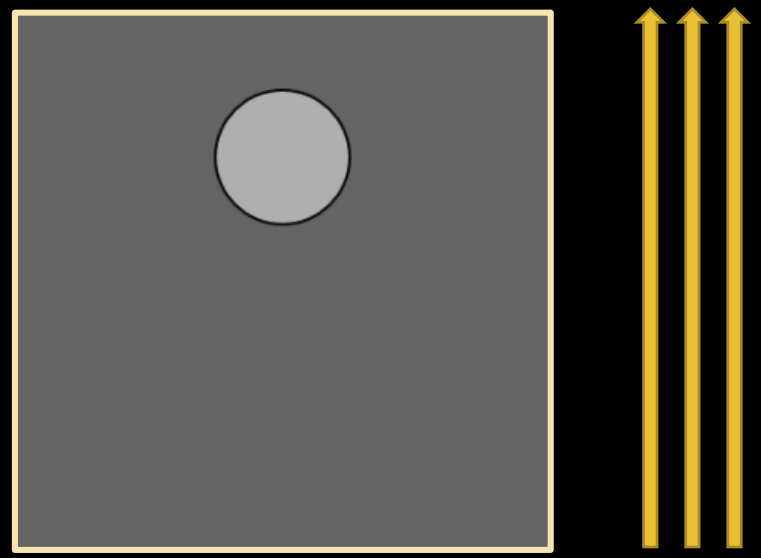
```
if (x > width) {
    x = 0;
    y += 25;
}
```

Adding semicolon at
the end of an if clause
is a common mistake

```
if (x > width);
    x = 0;
```

Note:
Braces are recommended always,
even if there is only one action.
(Also, it makes the Atom editor works\ better ☺)

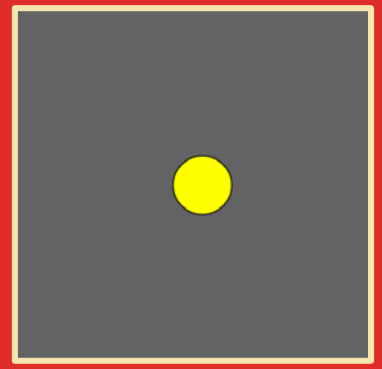
- Job – 4
  - ✓ Modify the previous program such that the ball moves from bottom to top of canvas **repeatedly**



```
var x = 0;
var step = 1;

function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(100);
  stroke(0);
  fill(255, 255, 0);
  ellipse(x, height/2, 50, 50);

  // condition
  if (x < 0 || x > width) {
      step = step * -1;
  }

  // move forward if step = 1;
  // move backward if step = -1;
  x = x + step;
}
```

USING LOGICAL OPERATOR

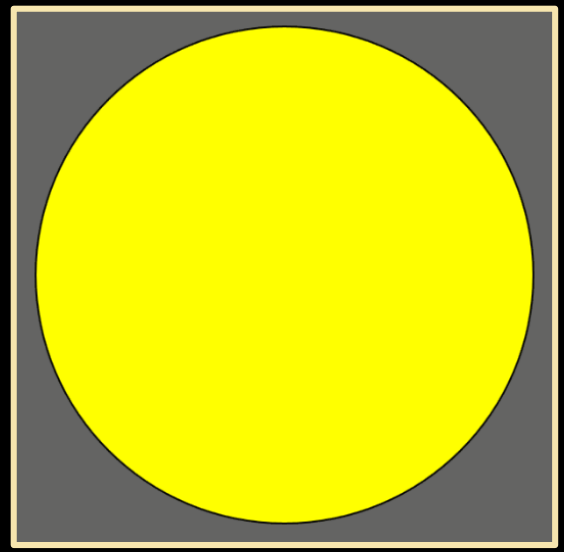
Logical operator	Meaning
&&	and
||	or



ball bounces at left/right boundaries

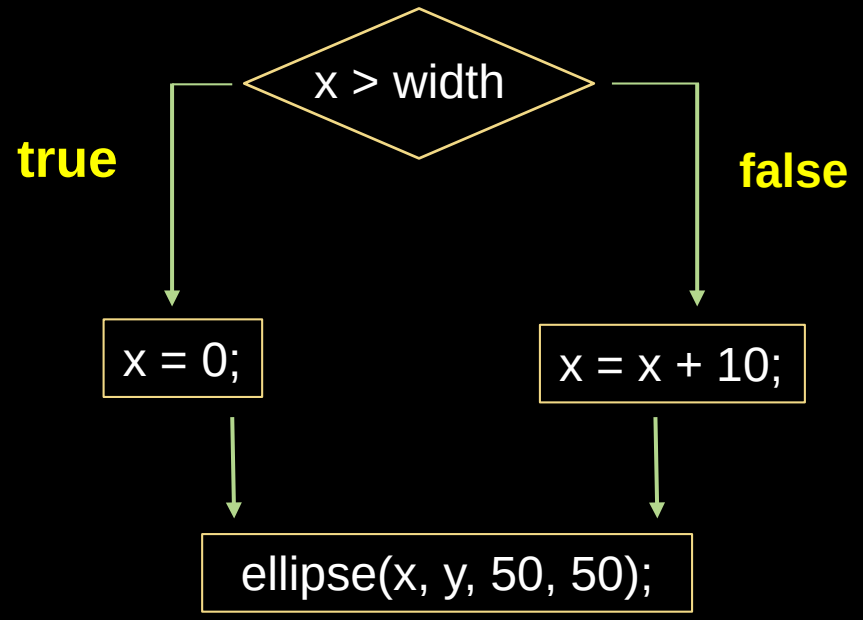
- Job – 5
  - Modify the program such that the balls grow from the center of canvas
  - The ball shrinks when hitting the wall and grows again when the size shrinks to zero

```
var radius = 25;

function setup() {
  createCanvas(300, 300);
}

function draw() {
  background(100);
  stroke(0);
  fill(255, 255, 0);
  ellipse(width/2, height/2, radius*2, radius*2);
}
```



IF-ELSE STATEMENT

```
if (x > width) {
    x = 0;
} else {
    x = x + 10;
}

ellipse(x, y, 50, 50);
```



JOB 6

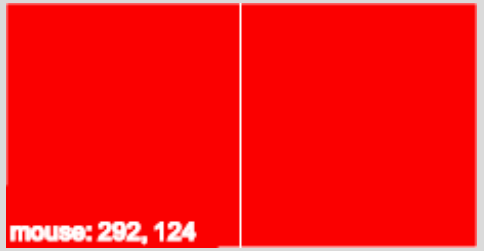
Modify the given program such that the canvas gradually turns red (r = r + 1) when mouse cursor is on the left hand side of canvas. Otherwise, the canvas gradually turns dark (r = r – 1).

If the value of r is greater than 255, set the value to 255.

If the value of r is smaller than 0, set the value of r to 0.

Given program: A white line on red background

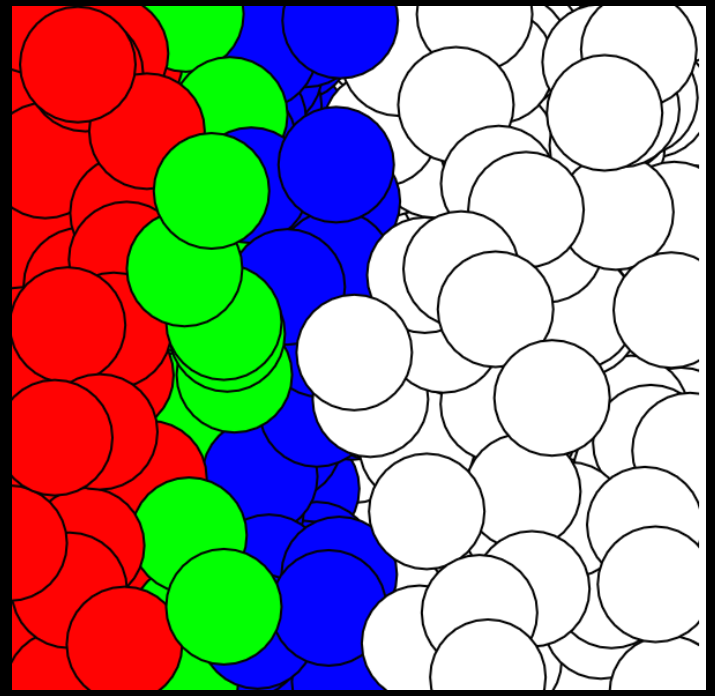
```
var r = 150;  // red

function setup() {
  createCanvas(400,200);
}

function draw() {
  background(r,0,0);
  stroke(255);
  line(width/2,0,width/2,height);

  //Show mouse coordinates
  fill(255); //choose text color
  text("mouse: " + mouseX + ", " + mouseY, 5, height-5);
}
```



Learn:
(1) Use double quotes for text literal: "...",
(2) use + to put together the output contents

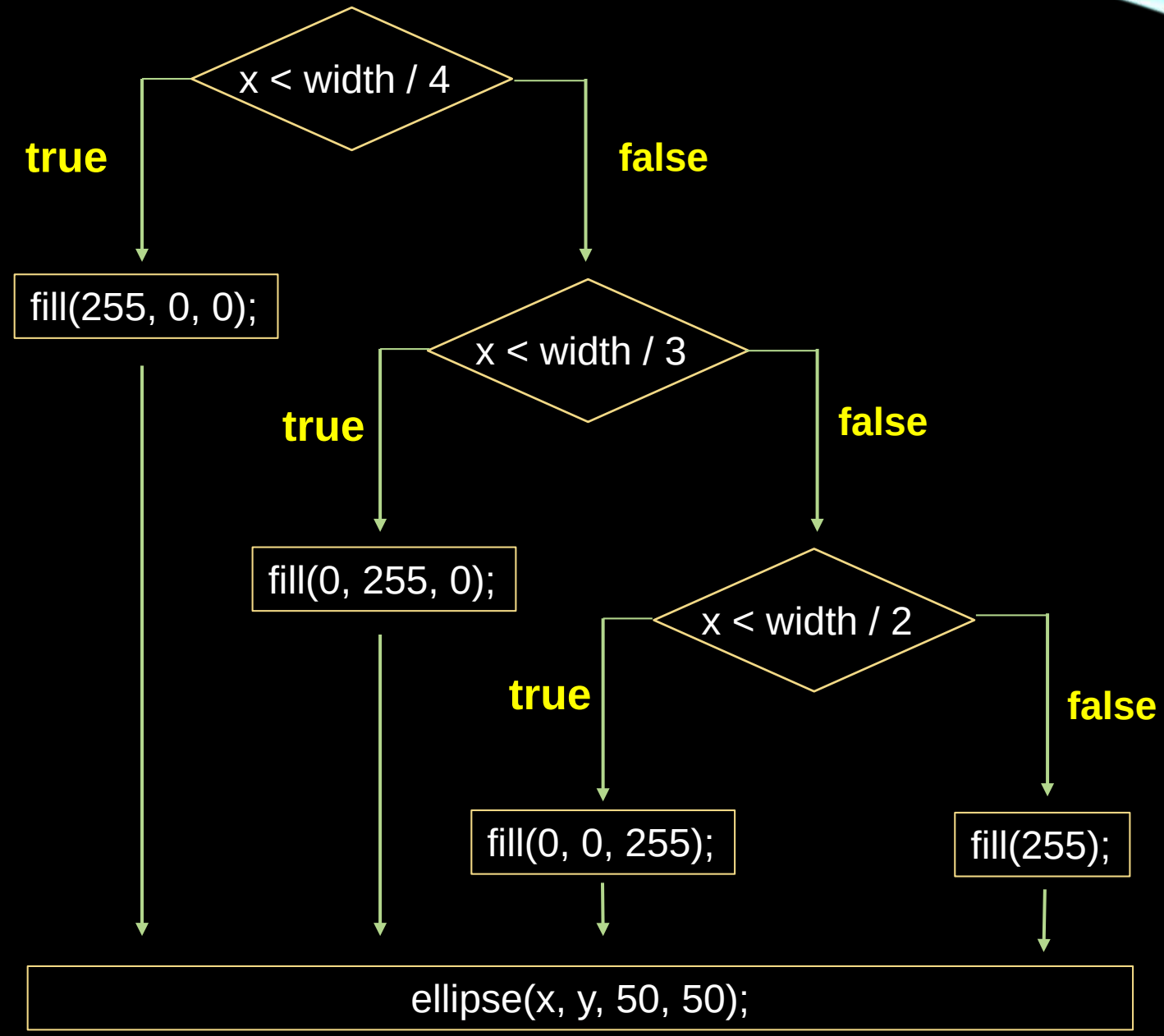
```
function draw() {
    var x = random(300);
    var y = random(300);

    if (x < width/4) {
      fill(255, 0, 0);
    } else if (x < width/3) {
      fill(0, 255, 0);
    } else if (x < width/2) {
      fill(0, 0, 255);
    } else {
      fill(255);
    }

    ellipse(x, y, 50, 50);
  }
```

ELSE IF



ELSE IF

```
if (x < width/4) {
    fill(255, 0, 0);
} else if (x < width/3) {
    fill(0, 255, 0)
} else if (x < width/2) {
    fill(0, 0, 255);
} else {
    fill(255);
}

ellipse(x, y, 50, 50);
```



```
if (x < width/2) {
    fill(255, 0, 0);
} else if (x < width/3) {
    fill(0, 255, 0)
} else if (x < width/4) {
    fill(0, 0, 255);
} else {
    fill(255);
}

ellipse(x, y, 50, 50);
```

What's wrong with this program?

No blue and green ellipse!

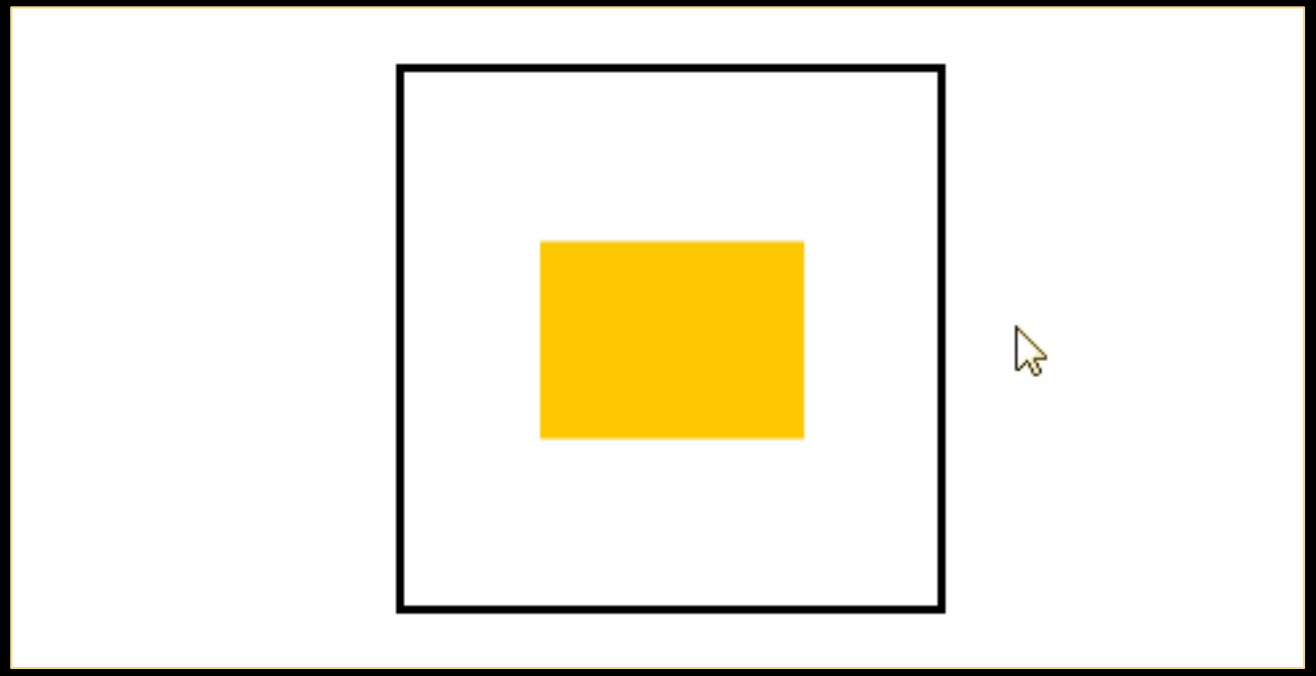
JOB 7

Create a program with followings:

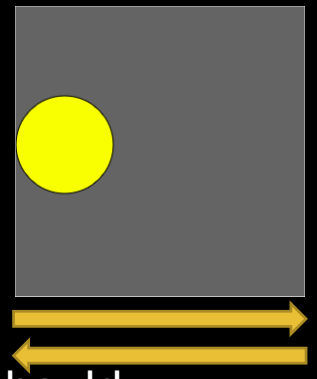
- ✓ Split the canvas equally into four zones
- ✓ A bouncing ball moves horizontally across the canvas
- ✓ The ball shows different colors when moving across different zones

JOB 8

Write a program such that the background of canvas turns dark when the mouse cursor is inside the yellow rectangle



- Job – 9 [Self Test Exercise]
  1. Create a ball and animate it like Job – 1.
  2. Add **var step = 3;** use it to control the movement step of the ball.
  3. Use 100 as the diameter of the ball
  4. Set the frame rate to 50 frames per second
  5. Make the ball move horizontally from left to right **and then right to left**. The direction should change whenever the rim of the ball touches (or has just gone beyond) the left or right boundary. (Hint: what should be the starting location of the ball? Make sure it is completely inside the boundary when it starts!)

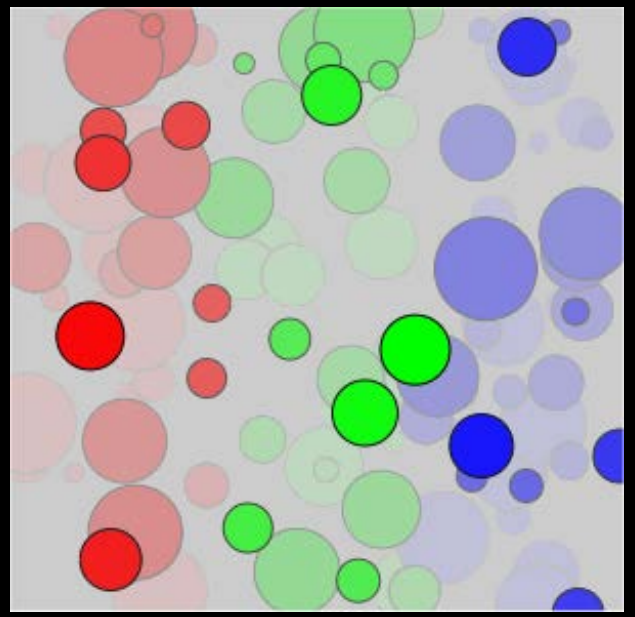


Suggested challenges for interested students

* When the ball bounces at the boundaries, reduce its size by 10% (That is, multiply the size by 0.9)
  Hint 1: use one more variable, d, to control the size. (Note: **do not use the name "size"**, it is a reserved word)
  Hint 2: you may need the braces { } in the if-then statement

* When the ball bounces at the left boundary, also add 1 to step so that the ball will move faster.

- Job – 10 [Self Test Exercise]
  1. In each frame, generate a random location for a ball.
  2. Depend on whether it is in the left / middle / right zone of the canvas, paint the ball in red, green, or blue respectively.
  3. Create a fading-out effect by giving an alpha value to background() command in draw().
  4. Control the diameters of the balls using a global variable var d = 10; add 1 to d every time after drawing a ball so that the painted balls will be larger and larger.
     Note: use d++; in this step.
  5. When d exceeds 50, reset it to 10.



Assignment of this lesson:

Create a program according to the requirement below.
Submit it on Canvas.
Due: Before the start of your next lesson.

The program on the right draws a picture of "Zoog".

Your task: Modify the program to

1. make the eyes of "Zoog" blink like the animation below. (2 marks)
2. change cheek color between red and pale-red. (2 marks)
3. add one or more creative features / animations. (1 mark)

Please refer to this animation for the desired effects.
https://courses.cs.cityu.edu.hk/cs1103/public/Workshop02_AsgAnimation/

```
//Given program:

function setup() {
  createCanvas(480, 360);
  background(225);

  ellipseMode(CENTER);
  rectMode(CENTER);

  // Body
  stroke(0);
  fill(150);
  rect(240, 190, 20, 100);

  // Head
  fill(255, 200, 200);
  ellipse(240, 160, 80, 80);
  noStroke();
  fill(255, 50, 50);
  ellipse(220, 170, 30, 20);
  ellipse(260, 170, 30, 20);

  // Eyes
  stroke(0);
  fill(255);
  ellipse(225, 150, 16, 16);
  ellipse(255, 150, 16, 16);
  fill(0);
  ellipse(225, 150, 3, 3);
  ellipse(255, 150, 3, 3);

  // Legs
  stroke(0);
  line(230, 240, 220, 250);
  line(250, 240, 260, 250);
}
```