Conditional Events

Mouse events and Operators

Lecturers: Caio Fonseca

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
http://www.wit.ie/

Topics List

- Mouse Events
- Keyboard Events
- Recap: Arithmetic Operators
- Order of Evaluation

What is an Event?

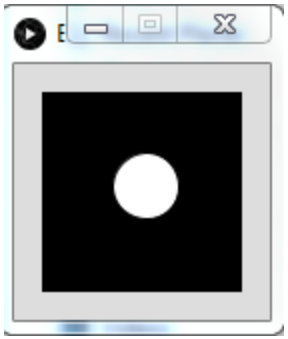
"An action such as a key being pressed,
the mouse moving, or
a new piece of data becoming available to read.
An event interrupts the normal
flow of a program to
run the code within an event block"
(Reas & Fry, 2014)

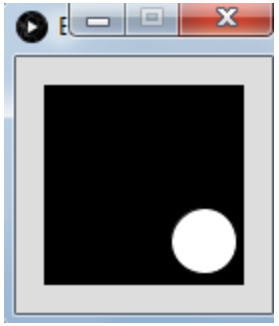
Mouse Events - Recap

- The processing variables mouseX and mouseY store the x-coordinate and y-coordinate of the cursor on the display window.
- When the program starts mouseX and mouseY are 0.
- If the cursor moves into the display window the values are set to the current position of the cursor.

```
void draw()
{
  println(mouseX + " : " + mouseY);
}
```

Mouse Events - Recap





Every time the cursor moves on the display window the circle moves as it coordinates change based on the location of the mouse.

Mouse Events

Mouse Variables	Description
mousePressed	***true*** if any mouse button is pressed, ***false*** otherwise. Note: this variable reverts to ***false*** as soon as the button is released.
mouseButton	Can have the value **LEFT**, **RIGHT** and **CENTER**, depending on the mouse button most recently pressed. Note: this variable retains its value until a different mouse button is pressed.

Mouse Events

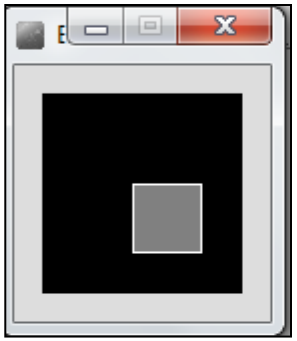
- Mouse and keyboard events only work when a program has draw().

- Without draw(), the code is only run once and then stops listening for events.

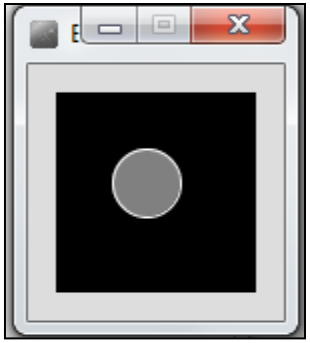
Source: https://processing.org/reference/

Mouse Pressed Example 3.5

- Functionality:
  - If the mouse is pressed, draw a gray square with a white outline.



  - Otherwise draw a gray circle with a white outline.



Mouse Pressed Example 3.5

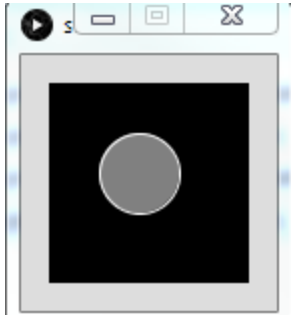
sketch_191126a | Processing 3.0b6

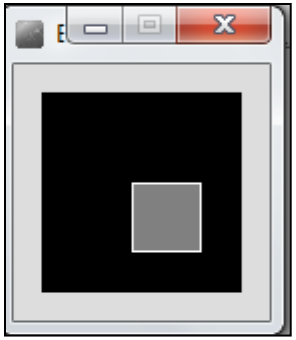
File Edit Sketch Debug Tools Help

Java ▾

sketch_191126a

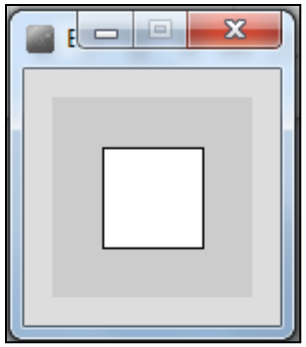
```
1
2  void setup()
3  {
4    size(100,100);
5  }
6
7  void draw()
8  {
9
10   background(0);
11   stroke(255);
12   fill(128);
13
14     if (mousePressed == true)
15     {
16
17           rect(45, 45, 40, 40);
18     }
19
20     else
21     {
22
23           ellipse(45, 45, 40, 40);
24     }
25
26 }
```

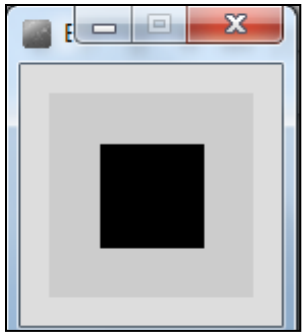




Mouse Pressed Example 3.6

- Functionality:
  - If the mouse is pressed, set the fill to white and draw a square.
  - Otherwise set the fill to black and draw a square.





Mouse Pressed Example 3.6

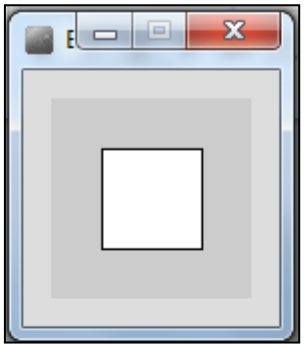
sketch_170609a | Processing 3.0b6

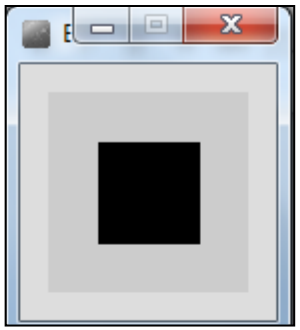
File Edit Sketch Debug Tools Help

Java ▾

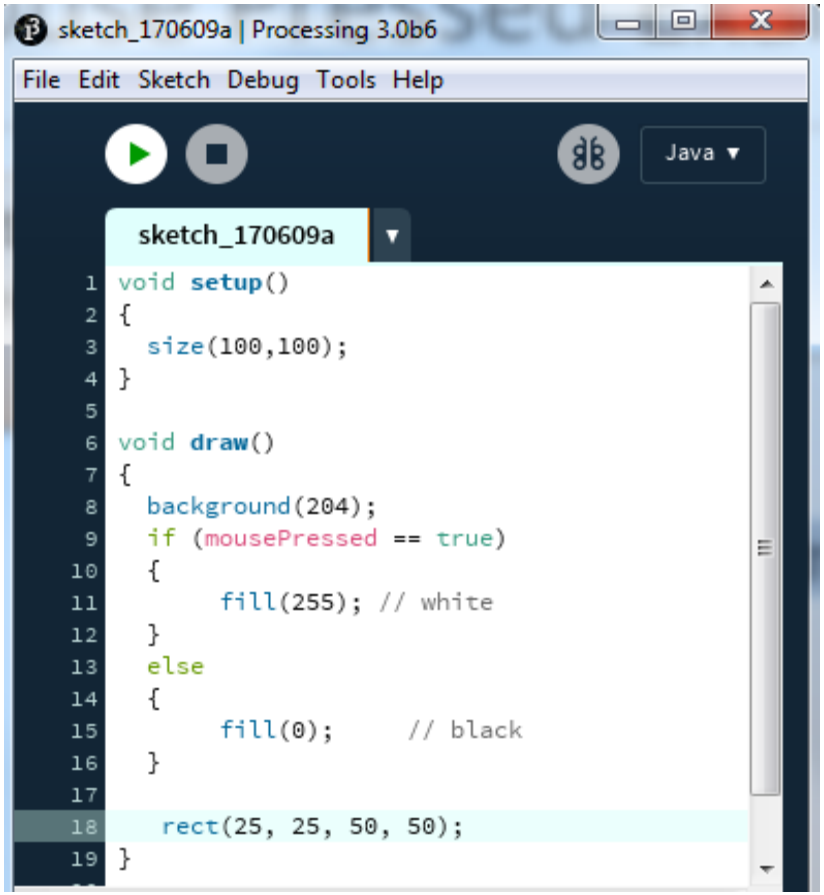
sketch_170609a

```
1  void setup()
2  {
3    size(100,100);
4  }
5
6  void draw()
7  {
8    background(204);
9
10   if (mousePressed == true)
11   {
12        fill(255); // white
13        rect(25, 25, 50, 50);
14   }
15   else
16   {
17        fill(0);      // black
18        rect(25, 25, 50, 50);
19   }
20
21 }
```





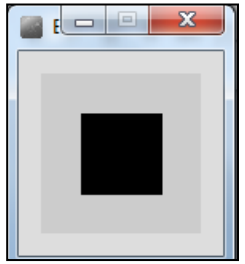
Mouse Pressed Example 3.6 - Improved

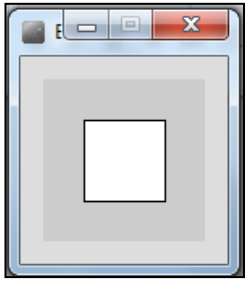


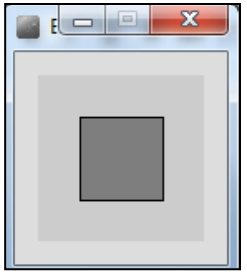
We removed the 2 statements to create the rectangle inside of the if statement. Instead we put it **ONCE** outside of the if statement. Why?

Mouse Pressed Example 3.7

- Functionality:
  - If the LEFT button on the mouse is pressed, set the fill to black and draw a square. As soon as the LEFT button is released, gray fill the square.
  - If the RIGHT button on the mouse is pressed, set the fill to white and draw a square. As soon as the RIGHT button is released, gray fill the square.
  - If no mouse button is pressed, set the fill to gray and draw a square.







Mouse Pressed Example 3.7

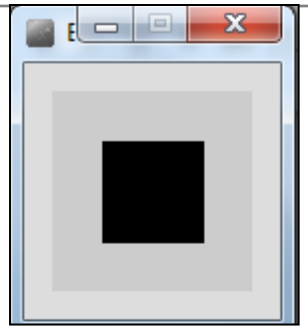
sketch_191126a | Processing 3.0b6

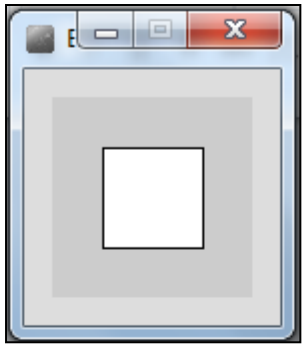
File Edit Sketch Debug Tools Help

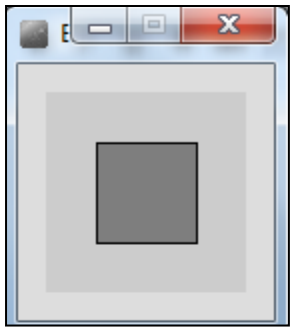
Java ▾

sketch_191126a

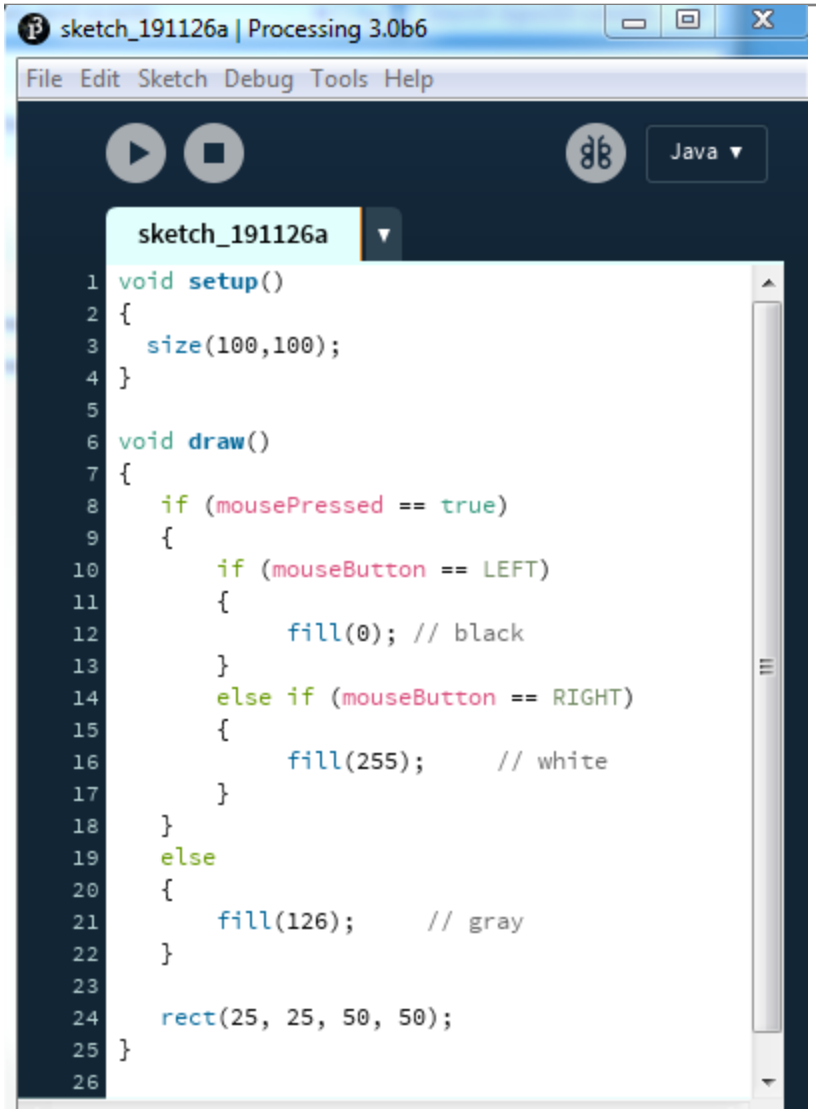
```
1  void setup()
2  {
3    size(100,100);
4  }
5
6  void draw()
7  {
8     if (mousePressed == true )
9     {
10        if (mouseButton == LEFT)
11        {
12             fill(0); // black
13             rect(25, 25, 50, 50);
14        }
15        else if (mouseButton == RIGHT)
16        {
17             fill(255);     // white
18             rect(25, 25, 50, 50);
19
20        }
21    }
22    else
23    {
24        fill(126);     // gray
25        rect(25, 25, 50, 50);
26    }
27 }
28
29
```







Mouse Pressed Example 3.7 - Improved




```
1  void setup()
2  {
3    size(100,100);
4  }
5
6  void draw()
7  {
8     if (mousePressed == true)
9     {
10        if (mouseButton == LEFT)
11        {
12             fill(0); // black
13        }
14        else if (mouseButton == RIGHT)
15        {
16             fill(255);      // white
17        }
18    }
19    else
20    {
21        fill(126);      // gray
22    }
23
24    rect(25, 25, 50, 50);
25 }
26
```

We removed the 3 statements to create the rectangle inside the if statement. Instead we put it **ONCE** outside of the if statement. Again why?

Mouse Pressed Example 3.7 - Improved

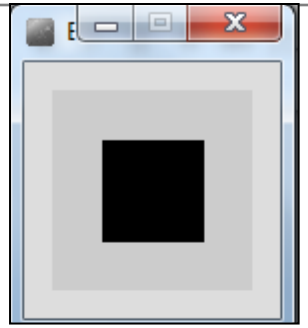
sketch_171023a | Processing 3.0b6

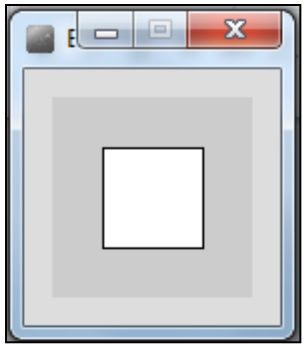
File Edit Sketch Debug Tools Help

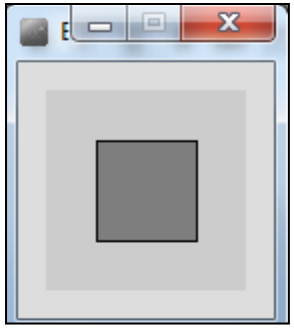
Java

sketch_171023a

```
1  void setup()
2  {
3    size(100,100);
4  }
5
6  void draw()
7  {
8     if (mousePressed == true && mouseButton == LEFT)
9     {
10
11              fill(0); // black
12              rect(25, 25, 50, 50);
13    }
14    else if (mousePressed == true && mouseButton == RIGHT)
15     {
16              fill(255);      // white
17              rect(25, 25, 50, 50);
18
19    }
20
21    else // or else if (mousePressed == false)
22    {
23        fill(126);      // gray
24        rect(25, 25, 50, 50);
25    }
26 }
```







Mouse Pressed Example 3.7 - Improved

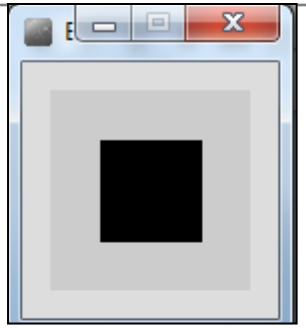
sketch_171023a | Processing 3.0b6

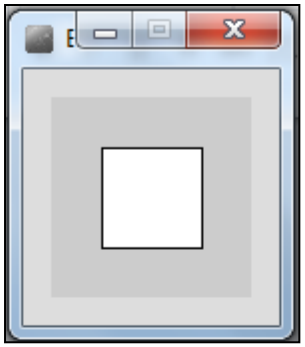
File Edit Sketch Debug Tools Help

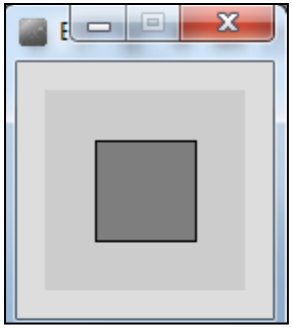
Java ▾

sketch_171023a

```
{
  size(100,100);
}

void draw()
{
   if (mousePressed == true && mouseButton == LEFT)
   {
            fill(0); // black
   }
   else if (mousePressed == true && mouseButton == RIGHT)
    {
            fill(255);     // white
   }

   else // or else if (mousePressed == false)
   {
       fill(126);     // gray
   }

 rect(25, 25, 50, 50);
}
```

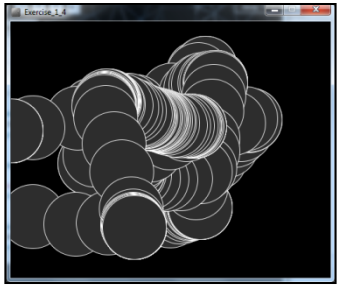


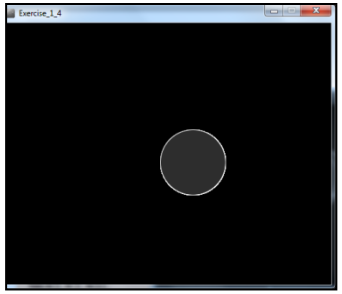




Mouse Pressed Example 3.8

- Functionality:
  - Draw a circle on the mouse (x,y) coordinates.
  - Each time you move the mouse, draw a new circle.
  - All the circles remain in the sketch until you press a mouse button.
  - When you press a mouse button, the sketch is cleared and a single circle is drawn at the mouse (x,y) coordinates.





Mouse Pressed Example 3.8

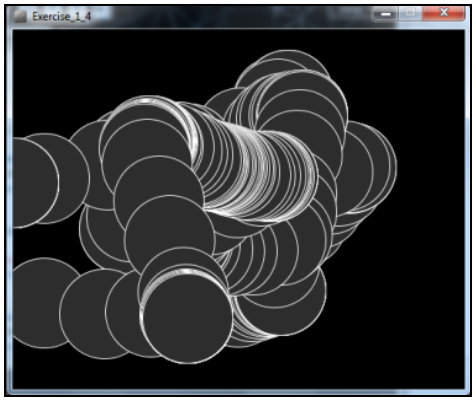
sketch_191126a | Processing 3.0b6

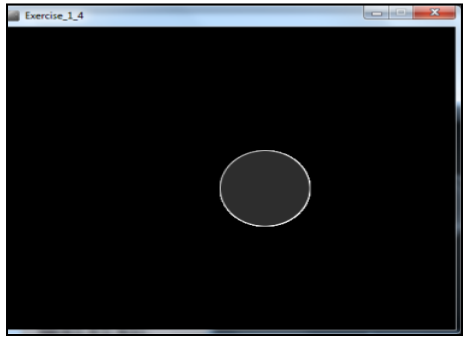
File Edit Sketch Debug Tools Help

Java ▾

sketch_191126a

```
1
2  void setup()
3  {
4    size(500,400);
5    background(0);
6  }
7
8  void draw()
9  {
10
11   if (mousePressed == true)
12   {
13     background(0);
14   }
15
16   stroke(255);
17   fill(45,45,45);
18   ellipse(mouseX, mouseY, 100, 100);
19
20 }
21
22
```





Q. What would happen if background was set to black in draw()??

https://processing.org/tutorials/interactivity/

Mouse Pressed Example 3.8

sketch_191126a | Processing 3.0b6

File Edit Sketch Debug Tools Help

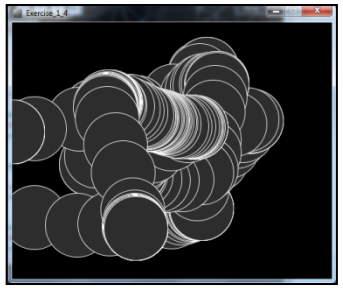
Java

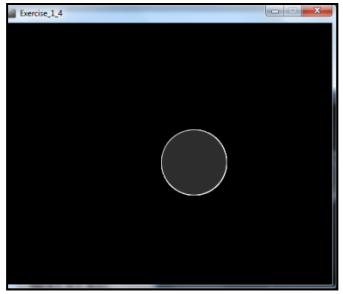
sketch_191126a

```
1
2  void setup()
3  {
4     size(500,400);
5     background(0);
6     stroke(255);
7     fill(45,45,45);
8     ellipse(mouseX, mouseY, 100, 100);
9
10 }
11
12 void draw()
13 {
14   if (mousePressed == true)
15   {
16     background(0);
17   }
18
19 }
20
21
```

Now we have moved the stroke and fill function calls to the setup() function.

Q: Does this change the functionality of our sketch?





https://processing.org/tutorials/interactivity/

Topics List

- Mouse Events
- Keyboard Events
- Recap: Arithmetic Operators
- Order of Evaluation

Keyboard Events

Keyboard Variables	Description
keyPressed	***true*** if any key is pressed, ***false*** otherwise. Note: the key variable remains true while the key is held down and becomes ***false*** as soon as the key is released.
key	Stores a single alphanumeric character. They key can be displayed with the text() function.

KeyPressed – Example 1

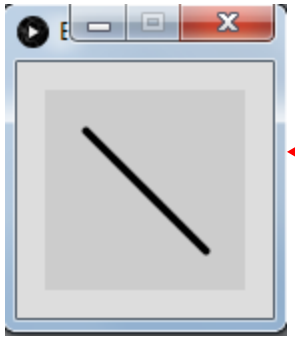
Ex_15 | Processing 3.0b6

File Edit Sketch Debug Tools Help

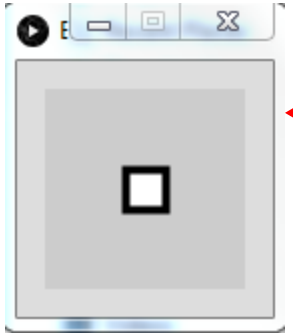
Java ▾

Ex_15

```
1  void setup()
2  {
3    size(100, 100);
4    strokeWeight(4);
5  }
6
7  void draw()
8  {
9    background(204);
10
11   if (keyPressed == true)  // If a key is pressed
12   {
13     line(20, 20, 80, 80);   // draw a line
14   }
15   else                     // Otherwise,
16   {
17     rect(40, 40, 20, 20);    // draw a rectangle
18   }
19 }
```



If ANY key is pressed



If NO key is pressed

KeyPressed – Example 2

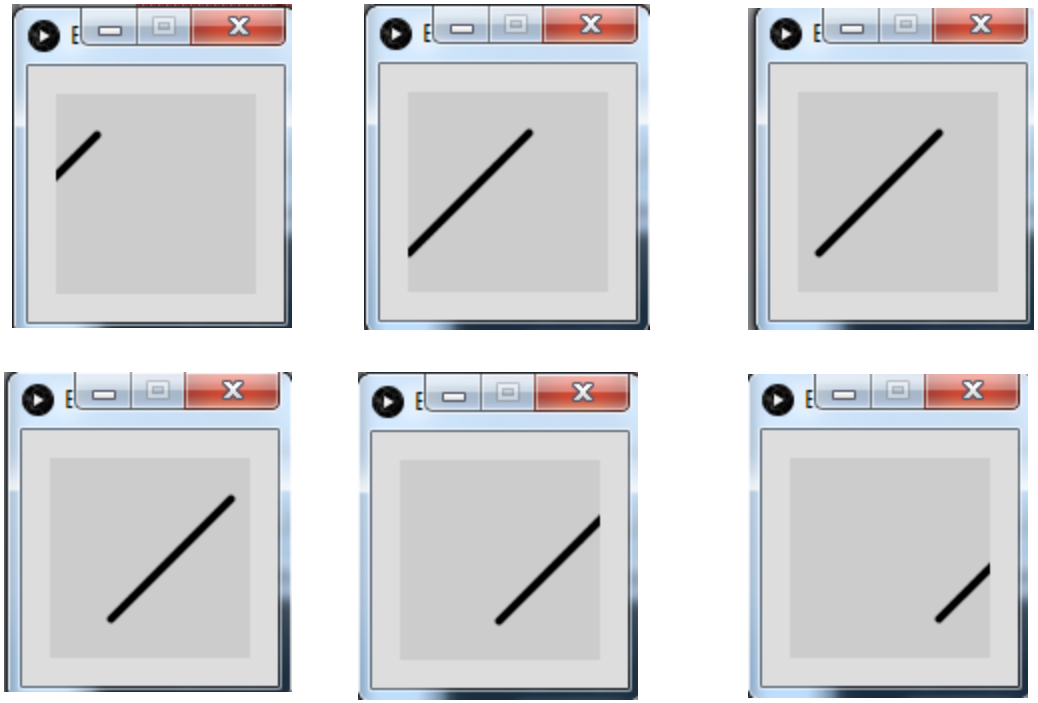
Ex_16 | Processing 3.0b6

File Edit Sketch Debug Tools Help

Java ▾

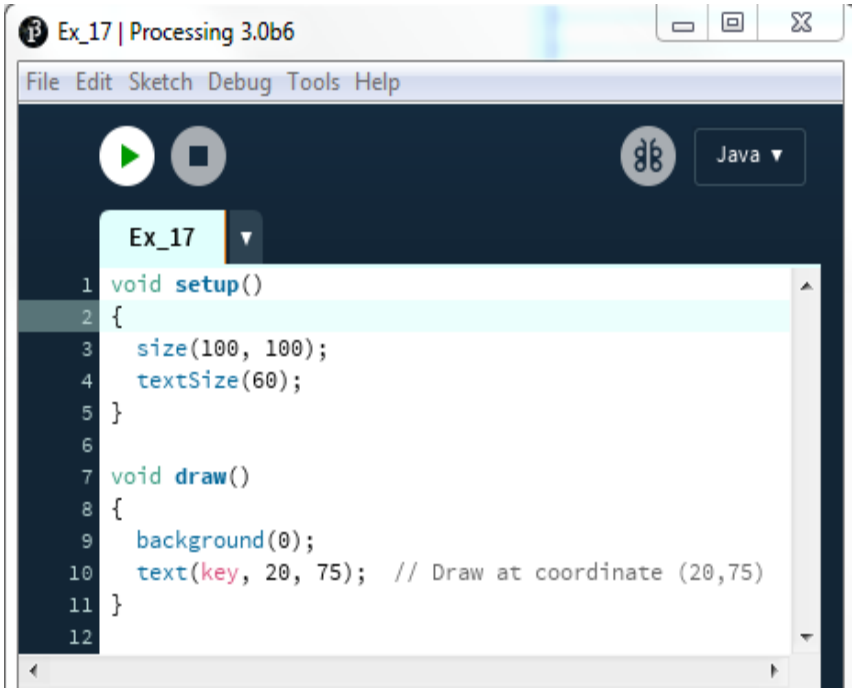
Ex_16

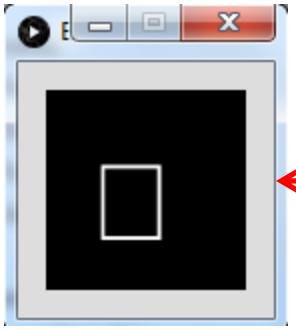
```
1  int x = 20;
2
3  void setup()
4  {
5    size(100, 100);
6    strokeWeight(4);
7  }
8
9  void draw()
10 {
11   background(204);
12   if (keyPressed == true) // If the key is pressed
13   {
14     x++;                  // add 1 to x
15   }
16   line(x, 20, x-60, 80);
17 }
18
```



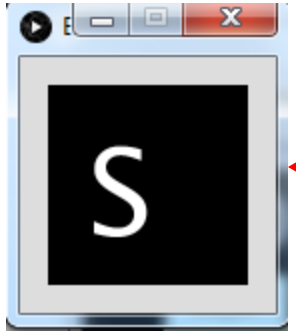
Every time ANY key is pressed the variable x is increased by 1 and the line draws at a new location on the screen.

KeyPressed (key) – Example 3





Before any key is pressed



When S is pressed

When 8 is pressed

- textSize function takes a value to set the size of the text that will be displayed
- The text method displays the key that was pressed and the coordinates of where the text will be displayed.

KeyPressed (key) – Example 4

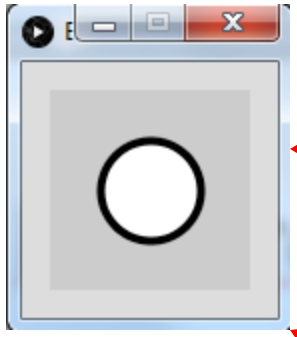
Ex_18 | Processing 3.0b6

File Edit Sketch Debug Tools Help

Java ▾

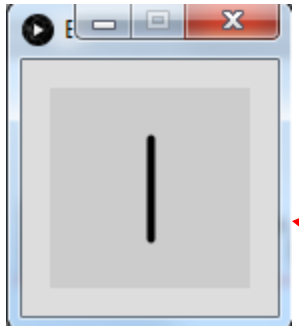
Ex_18

```
1  void setup()
2  {
3    size(100, 100);
4    strokeWeight(4);
5  }
6
7  void draw()
8  {
9    background(204);
10             // If the 'A' key is pressed draw a line
11   if ((keyPressed == true) && (key == 'A'))
12   {
13     line(50, 25, 50, 75);
14   }
15   else
16   {           // Otherwise, draw an ellipse
17     ellipse(50, 50, 50, 50);
18   }
19 }
```



Before any key is pressed

Also if any key except A is pressed



Only when A is pressed (not a)

Q. How would we change the code to get it to change to a line when we press ‘a’ aswell????

Topics List

- Mouse Events
- Keyboard Events
- Recap: Arithmetic Operators
- Order of Evaluation

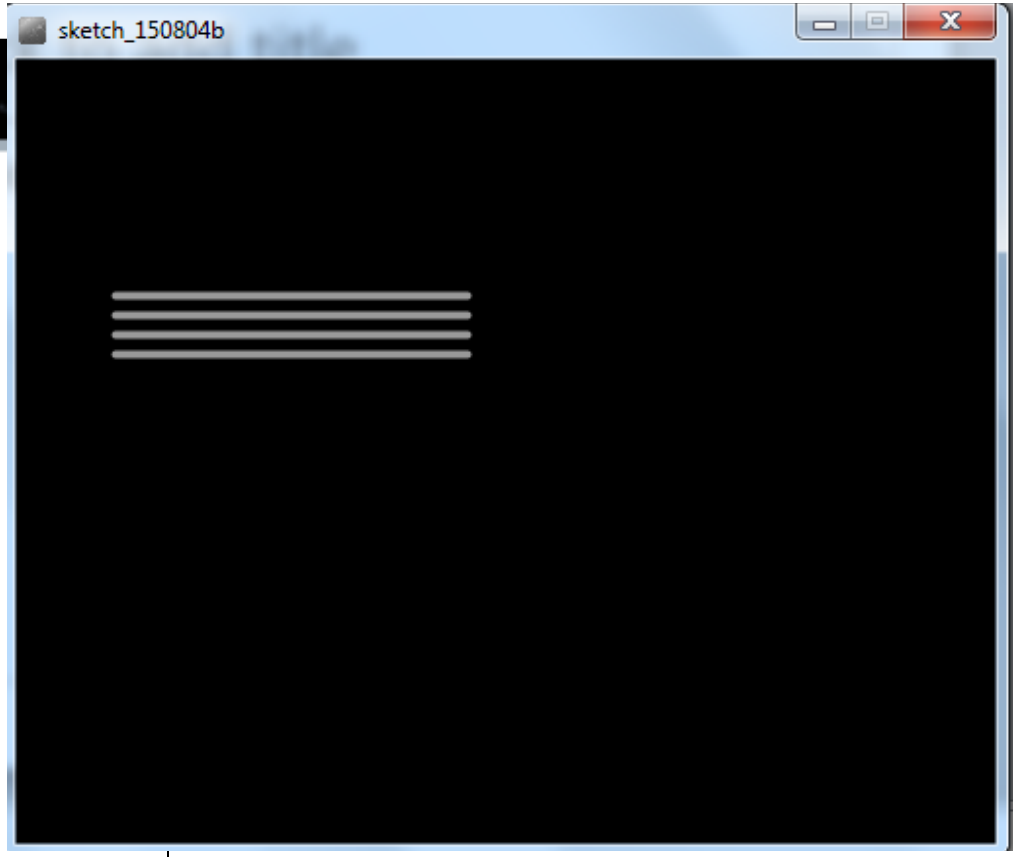
Recap: Arithmetic Operators

Arithmetic Operator	Explanation	Example(s)
+	Addition	6 + 2 amountOwed + 10
-	Subtraction	6 – 2 amountOwed – 10
*	Multiplication	6 * 2 amountOwed * 10
/	Division	6 / 2 amountOwed / 10

Recap: Arithmetic Operators: Example 1

sketch_150804b

```
size(500, 400);
background(0);
stroke(153);
strokeWeight(4);

int a = 50;
int b = 120;
int c = 180;

line(a, b, a+c, b);
line(a, b+10, a+c, b+10);
line(a, b+20, a+c, b+20);
line(a, b+30, a+c, b+30);
```

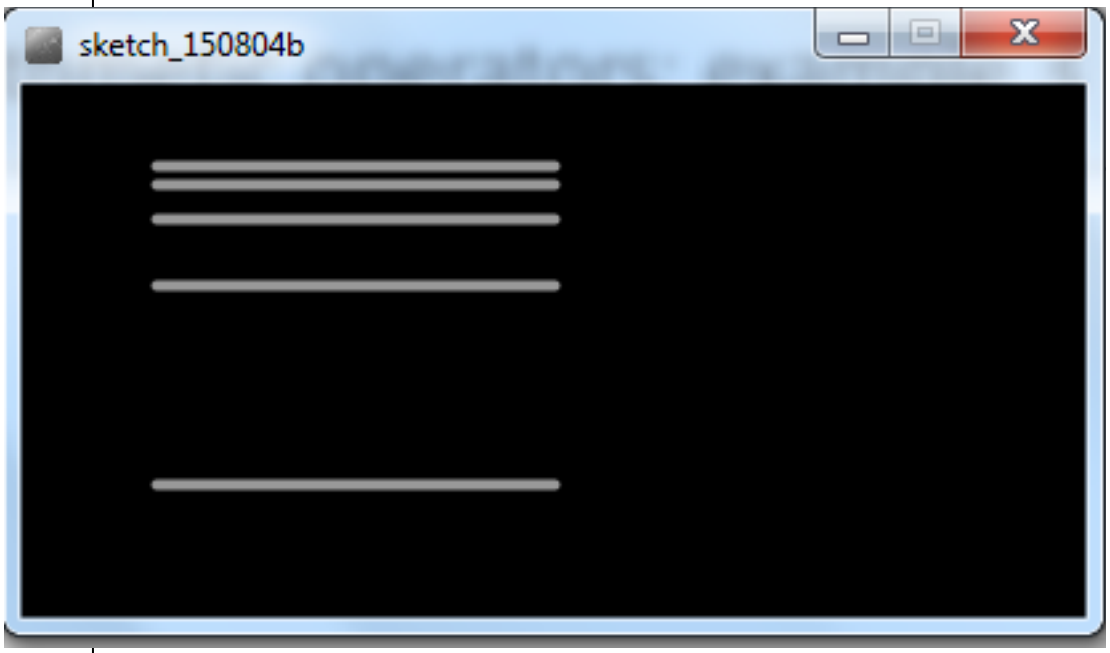


https://processing.org/examples/variables.html

Recap: Arithmetic Operators: Example 3

sketch_150804b

```
size(400, 200);
background(0);
stroke(153);
strokeWeight(4);

int a = 50;
int b = 1500;
int c = 4;

line(a, b/10, a*c, b/10);
line(a, b/20, a*c, b/20);
line(a, b/30, a*c, b/30);
line(a, b/40, a*c, b/40);
line(a, b/50, a*c, b/50);
```



Based on the Processing Example: Basics → Data → Variables

Arithmetic Operators

- If you want to keep track of how many times something happens, you are keeping a **running total** e.g.
  - The number of times you drew a line on the computer screen.
  - As each line is drawn, you add one to your counter variable.

Arithmetic Operators

```
int counter = 0;


void draw()
{
  line (mouseX, mouseY, 50,50);
  counter = counter + 1;
  println (counter);
}
```

Arithmetic Operators

- These examples are straightforward uses of the arithmetic operators.
- However, we typically want to do more complex calculations involving many arithmetic operators.
- To do this, we need to understand the **Order of Evaluation.**

Topics List

- Mouse Events
- Keyboard Events
- Recap: Arithmetic Operators
- Order of Evaluation

Order of Evaluation

- Brackets ()
- Multiplication (*)
- Division (/)
- Addition (+)
- Subtraction (-)

BoMDAS

Beware My Dear Aunt Sally

Order of Evaluation - Quiz

What are the results of these calculations?

Q1: 3+6*5-2

Q2: 3+6*(5-2)

Q3: (3+6)*5-2

Questions?

Produced by:

Dr. Siobhán Drohan
Mairead Meagher
Sinéad Walsh

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
http://www.wit.ie/