w1_01 | Processing 2.1.2

Java

w1_01

```
/*
 * Creative Coding
 * Week 1, 01 - Draw your name!
 * by Indae Hwang and Jon McCormack
 * Copyright (c) 2014 Monash University

 * This program allows you to draw using the mouse.
 * Press 's' to save your drawing as an image to the file "yourName.jpg"
 * Press 'r' to erase your drawing and start with a blank screen
 *
 */


// setup function -- called once when the program begins
void setup() {

  // set the display window to size 500 x 500 pixels
  size(500, 500);

  // set the background colour to white
  background(255);

  // set the rectangle mode to draw from the centre with a specified radius
  rectMode(RADIUS);
}


// draw function -- called continuously while the program is running
void draw() {

  /* draw a rectangle at your mouse point while you are pressing
   the left mouse button */

  if (mousePressed) {
    // draw a rectangle with a small random variation in size
    stroke(170); // set the stroke colour to a light grey
    fill(0, 150); // set the fill colour to black with transparency
    rect(mouseX, mouseY, random(6), random(6));
  }


  // save your drawing when you press keyboard 's'
  if (keyPressed == true && key=='s') {
    saveFrame("yourName.jpg");
  }

  // erase your drawing when you press keyboard 'r'
  if (keyPressed == true && key == 'r') {
    background(255);
  }
}
```

1

w1_01

creative coding | MONASH University

# COMMENTS

creative coding | MONASH University

```
/*
 * Creative Coding
 * Week 1, 01 (2 June 2014)- Draw your name!
 * by Indae Hwang and Jon McCormack
 * Copyright (c) 2014 Monash University

 * This program allows you to draw using the mouse.
 * Press 's' to save your drawing as an image to the file "yourName.jpg"
 * Press 'r' to erase your drawing and start with a blank screen
 *
 */
```

# COMMENTS

creative coding | MONASH University

```
/* comments like this can
   span multiple lines…   */

/*** this is ok as well ***/

/* oops! forgot the * ERROR! MISSING /

// this comment runs until the end of the line

// good for short comments

int a; // or after some code
```

# SETUP & DRAW FUNCTIONS

creative coding | MONASH University

```
void setup() {
  // set the display window to size 500 x 500 pixels
  size(500, 500);

  // set the background colour to white
  background(255);

  // set the rectangle mode to draw from the centre with a specified radius
  rectMode(RADIUS);
}
```

# SETUP & DRAW FUNCTIONS

creative coding | MONASH University

```
void draw() {


  /* draw a rectangle at your mouse point while you are pressing
   the left mouse button */

  if (mousePressed) {
    // draw a rectangle with a small random variation in size
    stroke(170); // set the stroke colour to a light grey
    fill(0, 150); // set the fill colour to black with transparency
    rect(mouseX, mouseY, random(6), random(6));
  }
  …
}
```

# SETUP & DRAW FUNCTIONS

creative coding | MONASH University

```
void setup() {
  …
}
```

called once to setup the program

```
void draw() {
  …
}
```

called repeatedly in a loop

to exit a sketch press the ‘esc’ key

# FUNCTIONS

creative coding | MONASH University

functions perform a task

Processing has many useful built-in functions, e.g.

```
size(x,y)
```

```
background(grey)
```

```
random(n)
```

but you can also create your own...

# FUNCTIONS

creative coding | MONASH University

## calling a function

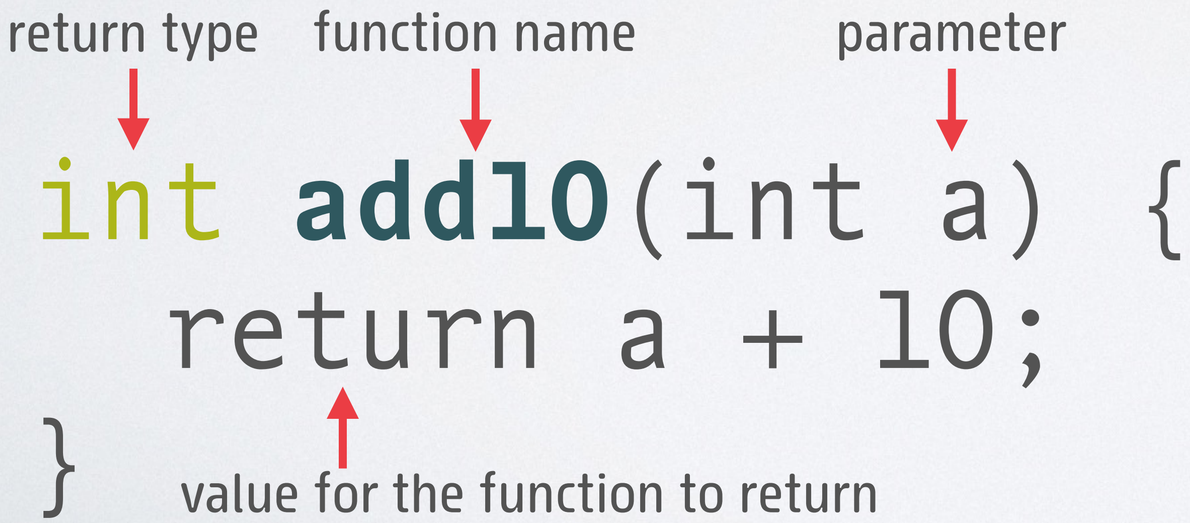
```
size(500, 500);
```

function name

arguments

---

## defining a function



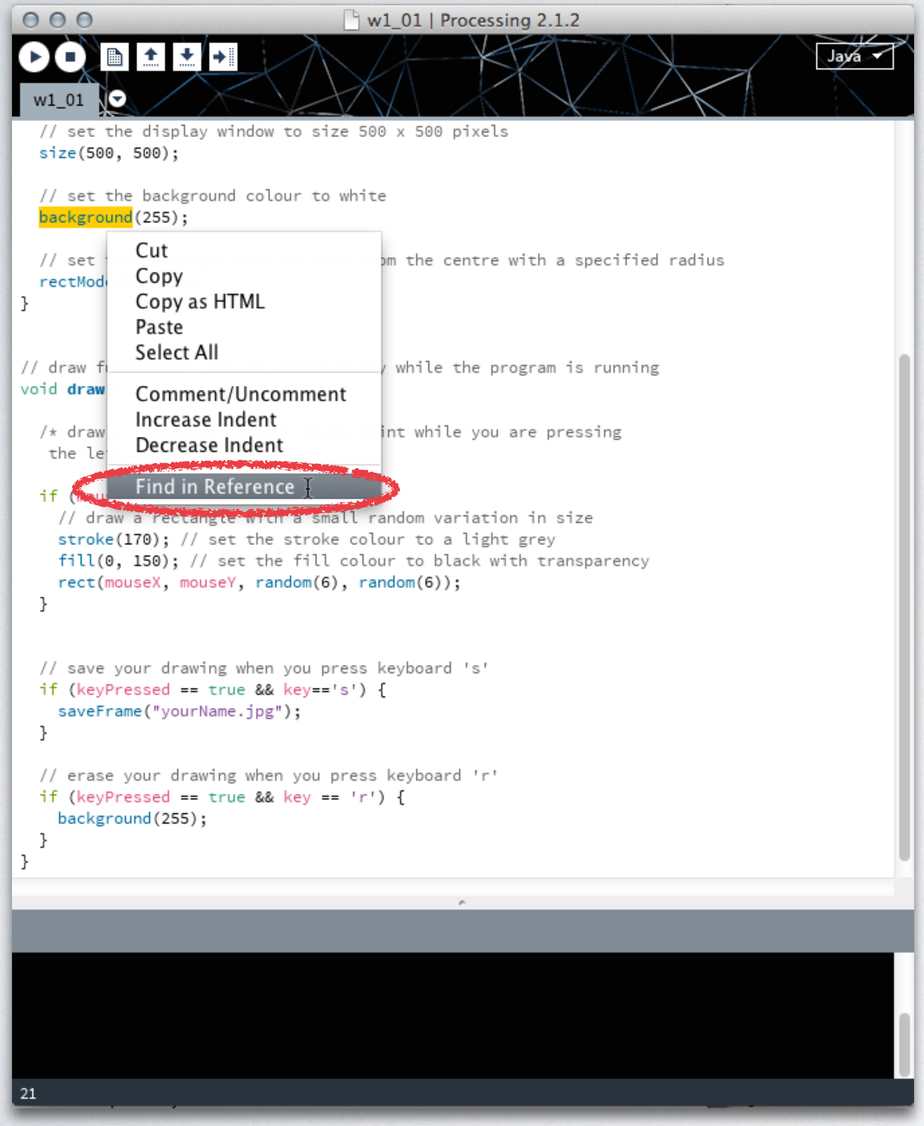
# FUNCTIONS

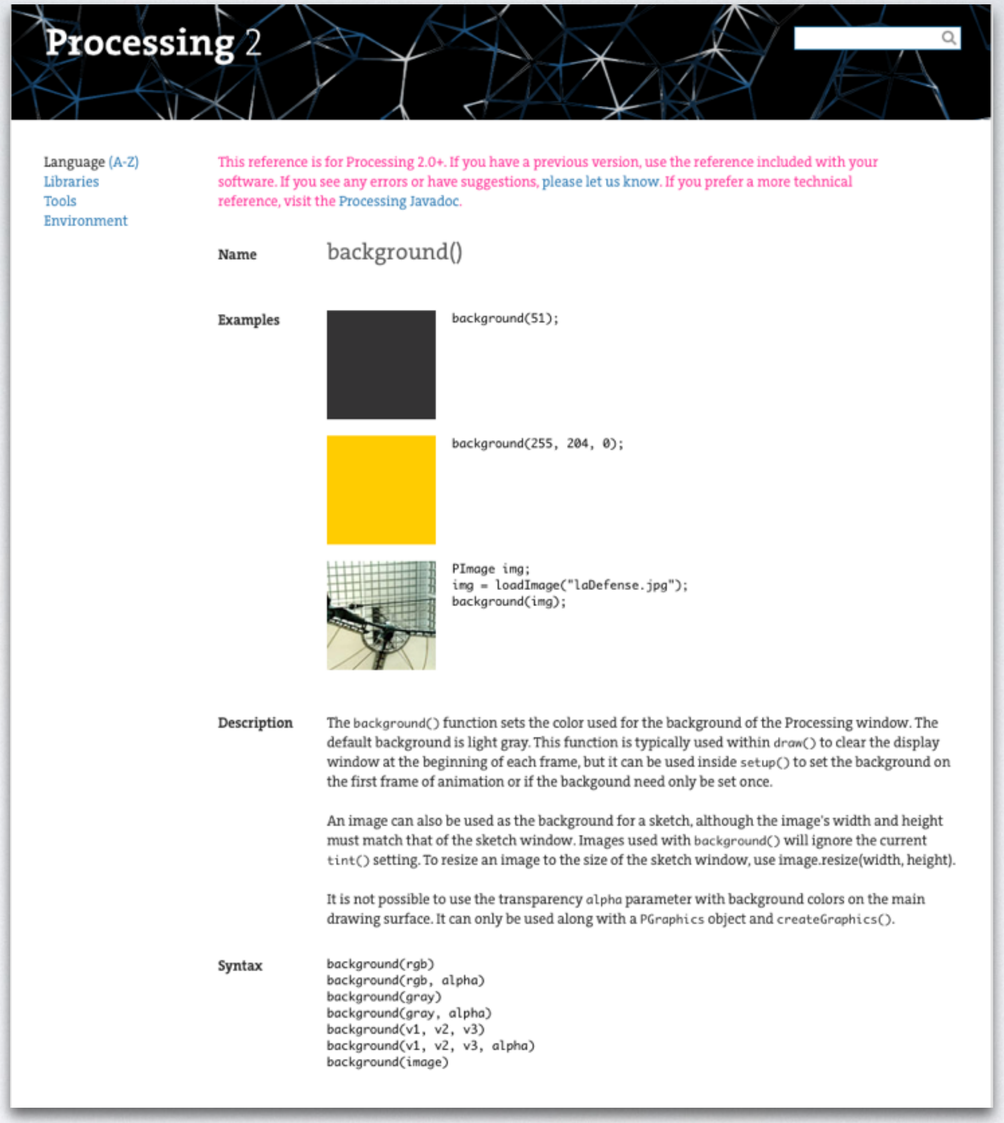
creative coding | MONASH University

```
size(x,y);
background(grey);
rectMode(mode);
stroke(grey);
fill(grey);
rect(x,y,w,h);
random(n);
saveFrame(“filename.type”);
```

# GETTING HELP

creative coding | MONASH University



# COLOUR IN PROCESSING

creative coding | MONASH University

```
fill(grey);
```

grey level from 0 to 255

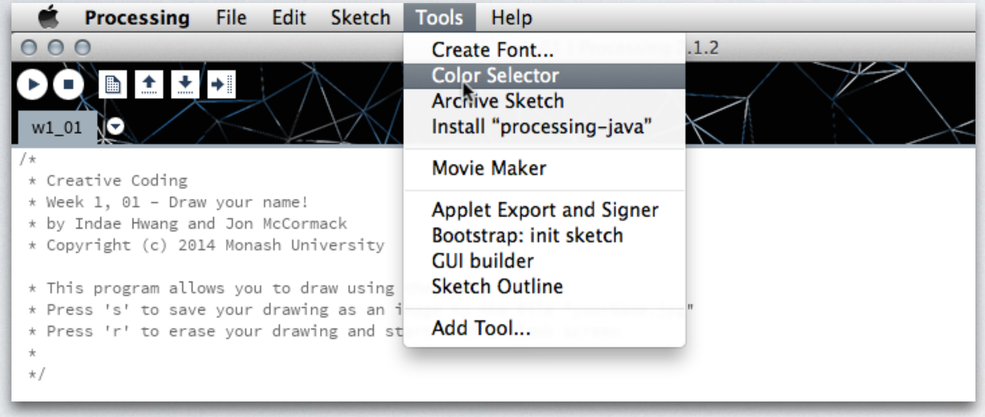

```
fill(grey,alpha);
```

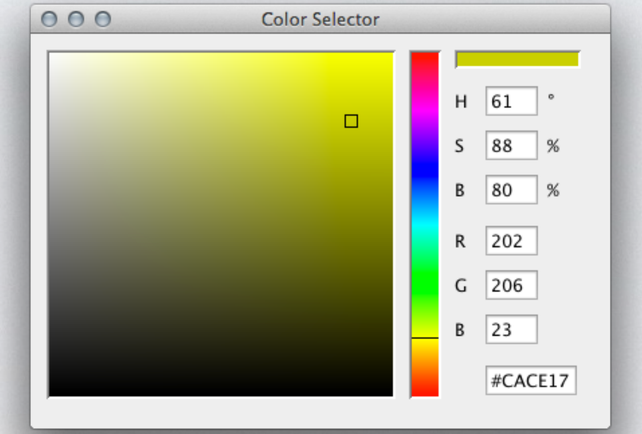
alpha level from 0 to 255

# COLOUR IN PROCESSING

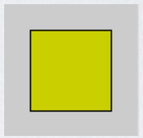
creative coding | MONASH University

## `fill(r,g,b);`



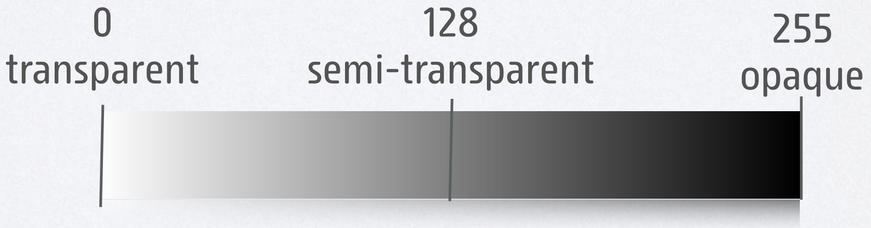




```
fill(202,206,23);
rect(20,20,60,60);
```

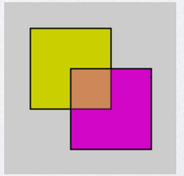


---

## `fill(r,g,b,alpha);` alpha level from 0 to 255




```
fill(202,206,23);
rect(20,20,60,60);
fill(206,23,197,100);
rect(50,50,60,60);
```



# ERRORS

creative coding | MONASH University

**syntax errors**
typos or spelling mistakes

**semantic errors**
do what I mean, not what I say

# ERRORS

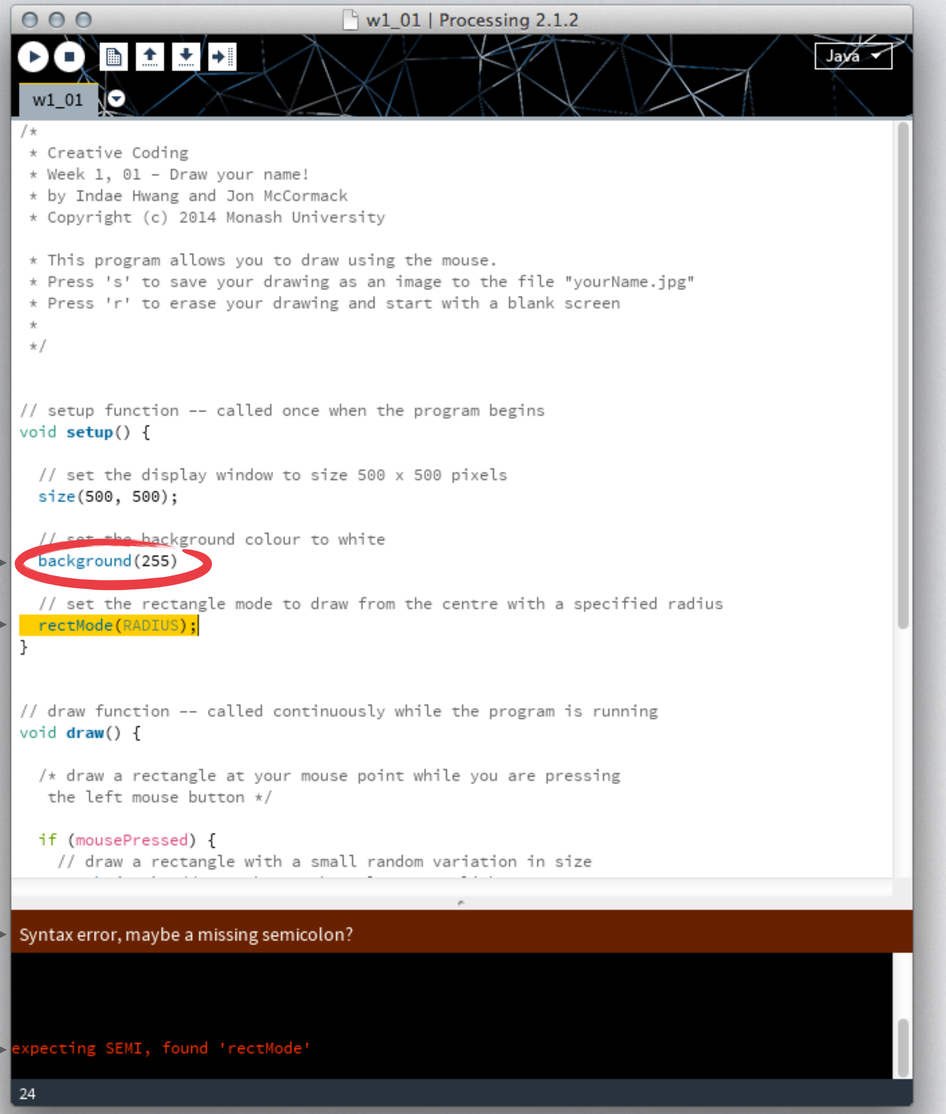
creative coding | MONASH University

missing ; →

Processing highlights where it thinks the error is located →

a more intelligible version →

Processing reports errors here →



# ERRORS

creative coding | MONASH University

**semantic errors**

do what I mean, not what I say