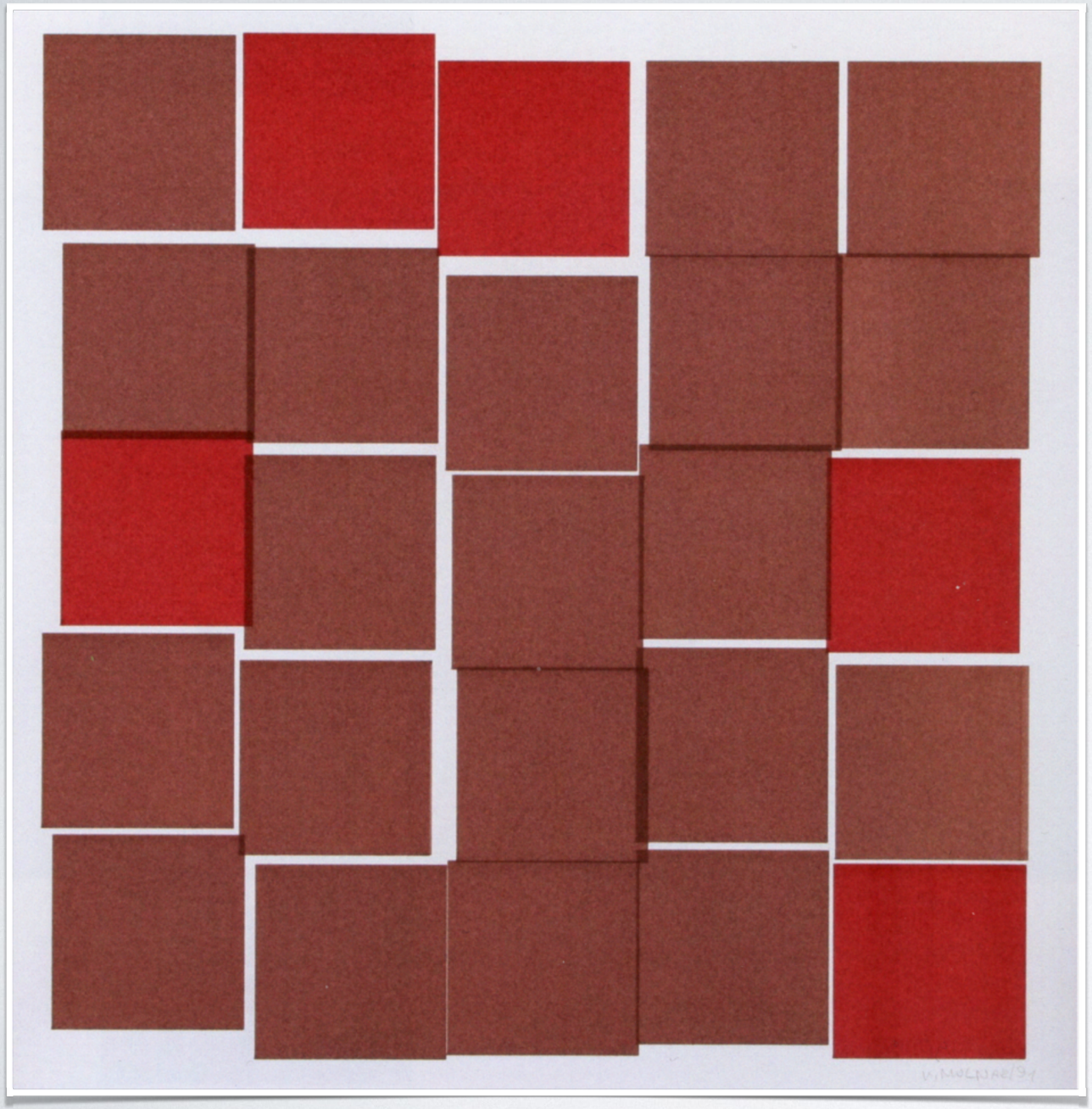
creative coding | MONASH University

02

LOOPING & REPEATING



Vera Molnar *25 Squares*, 1991

W2_01 : 9 SQUARES

creative coding | MONASH University

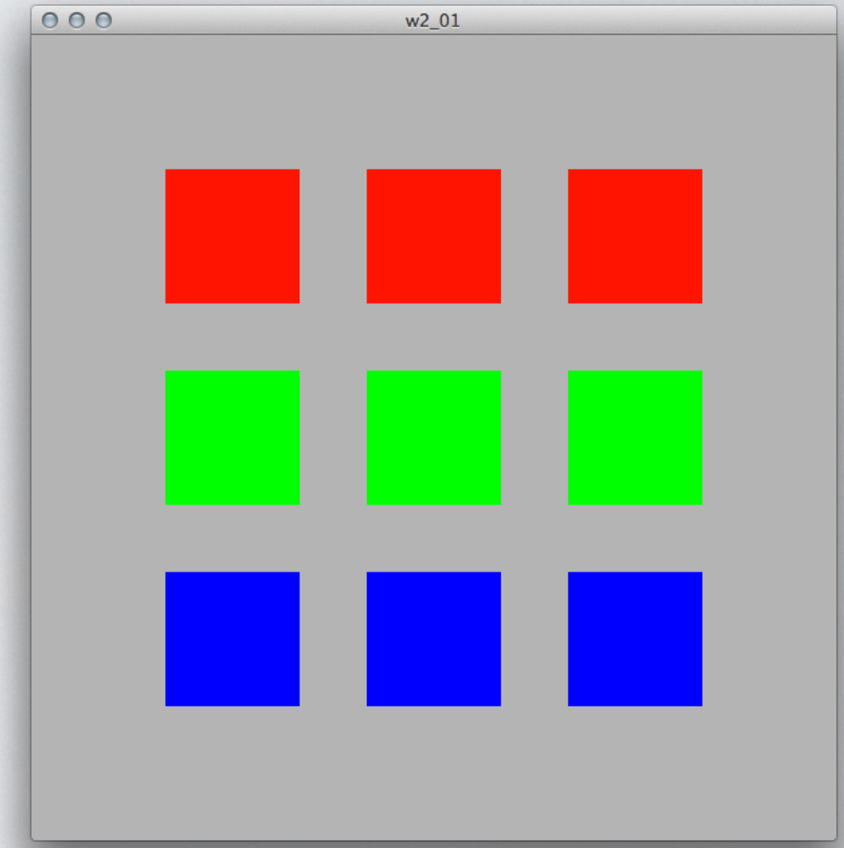
w2_01 | Processing 2.1.2

Java

w2_01

```
/*
 * Creative Coding
 * Week 2, 01 - Nine Squares
 * by Indae Hwang
 * Copyright (c) 2014 Monash University

 * This program draws 3 rows of 3 squares in the display window
 * Each row is coloured red, green, and blue.
 * Each rectangle is draw individually, meaning there are 9 rect calls.
 *
 */

void setup() {
  size(600, 600);
  background(180);
  noLoop();  // only execute the draw function once
  rectMode(CENTER); // set the rectangle drawing mode to specify the rectangle's centre
  noStroke();
}


void draw() {

  // row 1: red
  fill(255, 0, 0);
  rect(150, 150, 100, 100);
  rect(300, 150, 100, 100);
  rect(450, 150, 100, 100);

  // row 2: green
  fill(0, 255, 0);
  rect(150, 300, 100, 100);
  rect(300, 300, 100, 100);
  rect(450, 300, 100, 100);

  // row 3: blue
  fill(0, 0, 255);
  rect(150, 450, 100, 100);
  rect(300, 450, 100, 100);
  rect(450, 450, 100, 100);

} // end draw
```

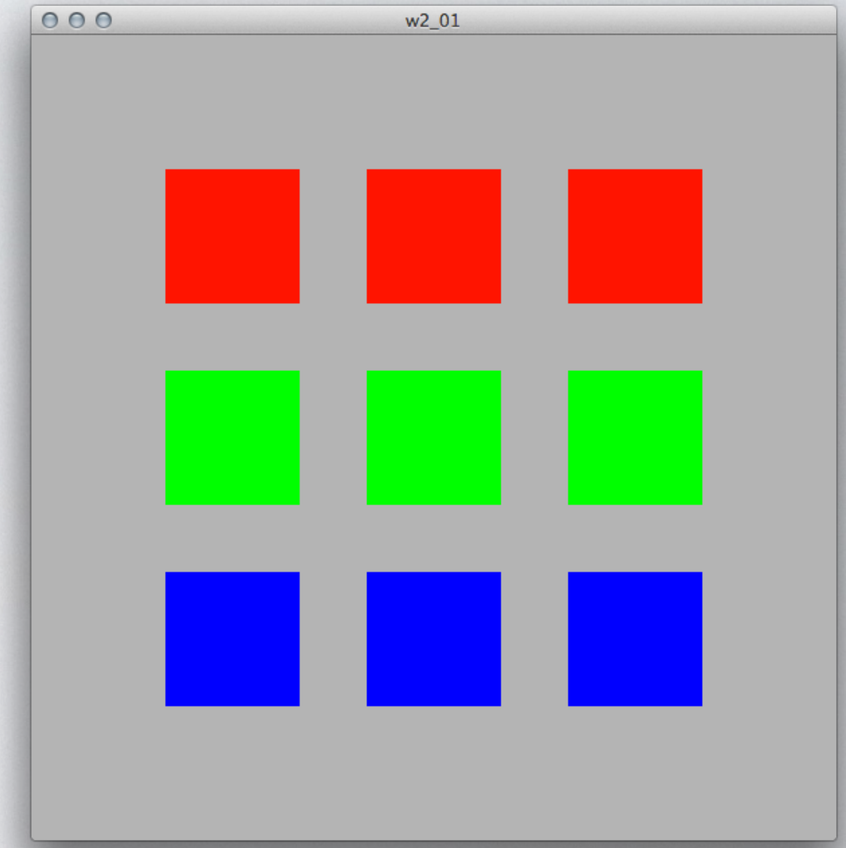
1



W2_01 : 9 SQUARES

creative coding | MONASH University

```
void draw() {
  // row 1: red
  fill(255, 0, 0);
  rect(150, 150, 100, 100);
  rect(300, 150, 100, 100);
  rect(450, 150, 100, 100);

  // row 2: green
  fill(0, 255, 0);
  rect(150, 300, 100, 100);
  rect(300, 300, 100, 100);
  rect(450, 300, 100, 100);

  // row 3: blue
  fill(0, 0, 255);
  rect(150, 450, 100, 100);
  rect(300, 450, 100, 100);
  rect(450, 450, 100, 100);
}
```



W2_02 : 9 SQUARES, PART 2

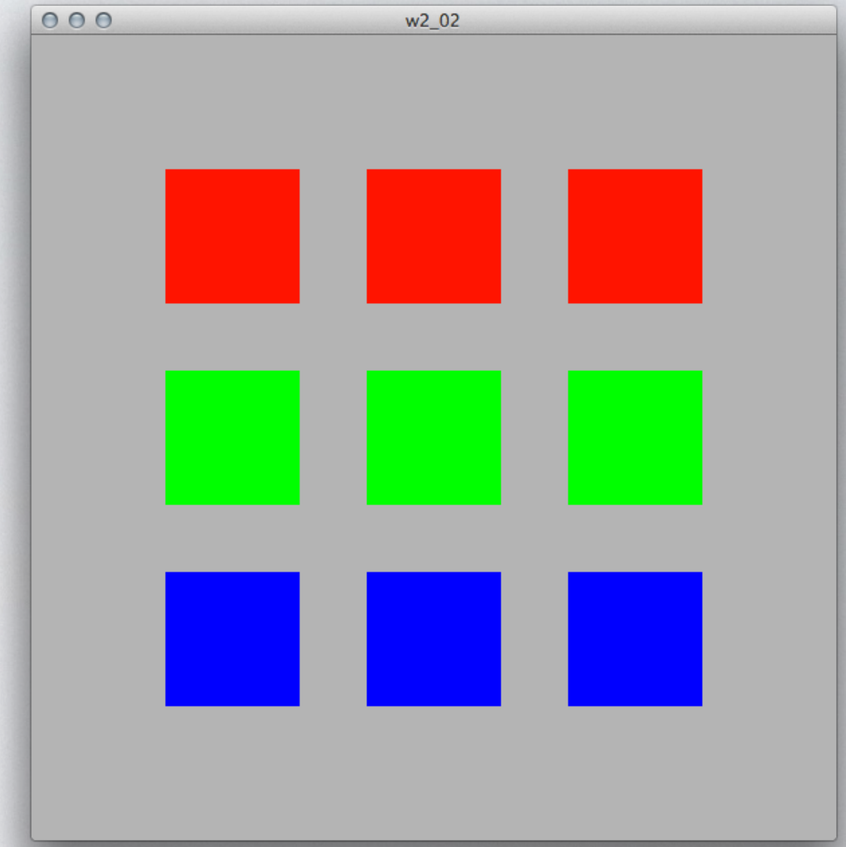
creative coding | MONASH University

w2_02 | Processing 2.1.2

Java

w2_02

```
void draw() {

  // the first case: 3 for loops, 1 for each row
  if (caseNum == 1) {

    // row 1
    fill(255, 0, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 150, 100, 100);
    }

    // row 2
    fill(0, 255, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 300, 100, 100);
    }

    // row 3
    fill(0, 0, 255);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 450, 100, 100);
    }

  } else if (caseNum == 2) {
    // second case: nested for loops

    for (int i=0; i<num; i++) { // col
      for (int j=0; j<num; j++) { // row

        // select the fill colour based on row
        switch(j) {
        case 0:
          fill(255, 0, 0);
          break;
        case 1:
          fill(0, 255, 0);
          break;
        case 2:
          fill(0, 0, 255);
          break;
        }
        // draw the rectangle
        rect(150 + 150*i, 150+ 150*j, 100, 100);
      } // end for (j)
    } // end for (i)
  } // end if
} //end of draw
```



W2_02 : 9 SQUARES, PART 2

creative coding | MONASH University

```
// change the value of caseNum from 1 to 2 to execute different parts of the code
caseNum = 1;
```

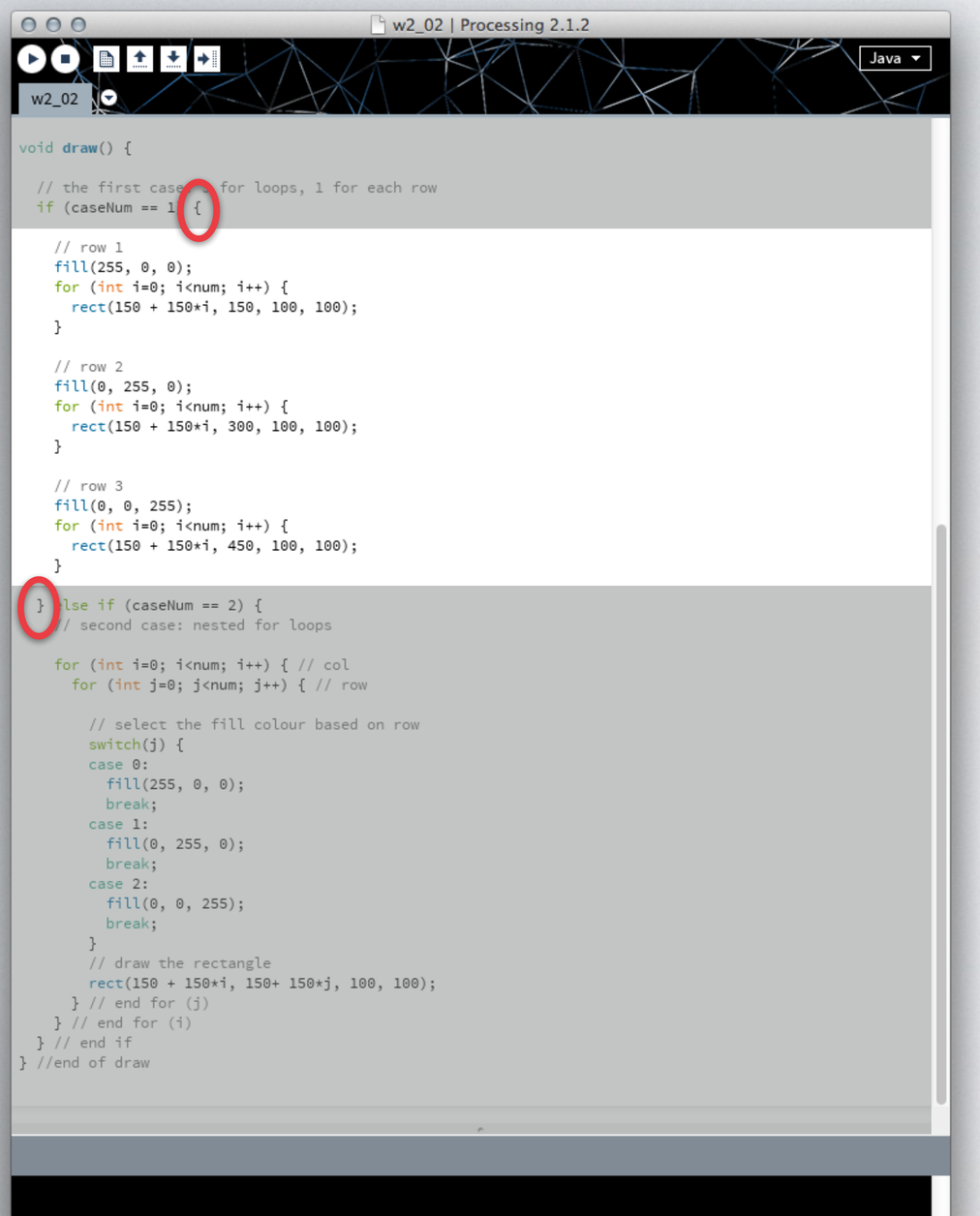
W2_02 : 9 SQUARES, PART 2

creative coding | MONASH University

```
// change the value of caseNum from 1 to 2 to execute different parts of the code
caseNum = 2;
```

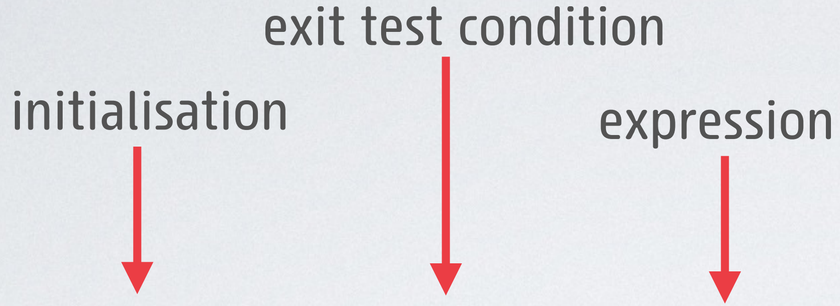
W2_02 : 9 SQUARES, PART 2

creative coding | MONASH University




```
void draw() {

  // the first case: 3 for loops, 1 for each row
  if (caseNum == 1) {

    // row 1
    fill(255, 0, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 150, 100, 100);
    }

    // row 2
    fill(0, 255, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 300, 100, 100);
    }

    // row 3
    fill(0, 0, 255);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 450, 100, 100);
    }

  } else if (caseNum == 2) {
    // second case: nested for loops

    for (int i=0; i<num; i++) { // col
      for (int j=0; j<num; j++) { // row

        // select the fill colour based on row
        switch(j) {
        case 0:
          fill(255, 0, 0);
          break;
        case 1:
          fill(0, 255, 0);
          break;
        case 2:
          fill(0, 0, 255);
          break;
        }
        // draw the rectangle
        rect(150 + 150*i, 150+ 150*j, 100, 100);
      } // end for (j)
    } // end for (i)
  } // end if
} //end of draw
```

FOR LOOPS

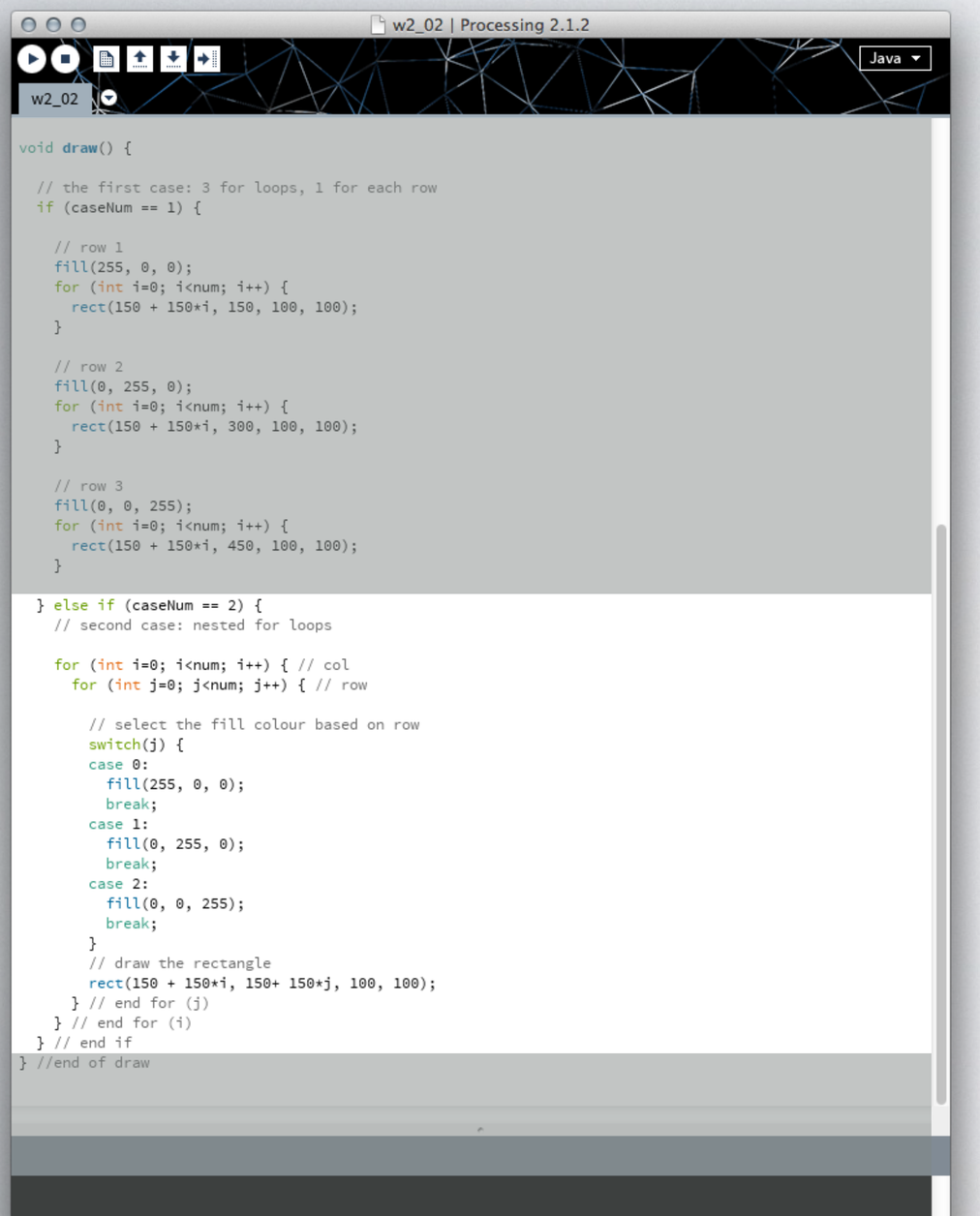



```
for (int i=0; i<num; i++) {
    rect(150 + 150*i, 150, 100, 100);
}
```

i	rect (150 + 150*i, 150, 100, 100)
0	rect(150,150,100,100)
1	rect(300,150,100,100)
2	rect(450,150,100,100)

W2_02 : 9 SQUARES, PART 2

creative coding | MONASH University




```
void draw() {

  // the first case: 3 for loops, 1 for each row
  if (caseNum == 1) {

    // row 1
    fill(255, 0, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 150, 100, 100);
    }

    // row 2
    fill(0, 255, 0);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 300, 100, 100);
    }

    // row 3
    fill(0, 0, 255);
    for (int i=0; i<num; i++) {
      rect(150 + 150*i, 450, 100, 100);
    }

  } else if (caseNum == 2) {
    // second case: nested for loops

    for (int i=0; i<num; i++) { // col
      for (int j=0; j<num; j++) { // row

        // select the fill colour based on row
        switch(j) {
        case 0:
          fill(255, 0, 0);
          break;
        case 1:
          fill(0, 255, 0);
          break;
        case 2:
          fill(0, 0, 255);
          break;
        }
        // draw the rectangle
        rect(150 + 150*i, 150+ 150*j, 100, 100);
      } // end for (j)
    } // end for (i)
  } // end if
} //end of draw
```

FOR LOOPS

creative coding | MONASH University

```
for (int i=0; i<num; i++) { // col
    for (int j=0; j<num; j++) { // row
      …
      // draw the rectangle
      rect(150 + 150*i, 150 + 150*j, 100, 100);
    } // end for (j)
} // end for (i)
```

FOR LOOPS

creative coding | MONASH University

i	j	rect(150+150*i, 150+150*j, 100, 100)
0	0	rect(150,150,100,100)
0	1	rect(150,300,100,100)
0	2	rect(150,450,100,100)
1	0	rect(300,150,100,100)
1	1	rect(300,300,100,100)
1	2	rect(300,450,100,100)
2	0	rect(450,150,100,100)
2	1	rect(450,300,100,100)
2	2	rect(450,450,100,100)

W2_03 : N SQUARES

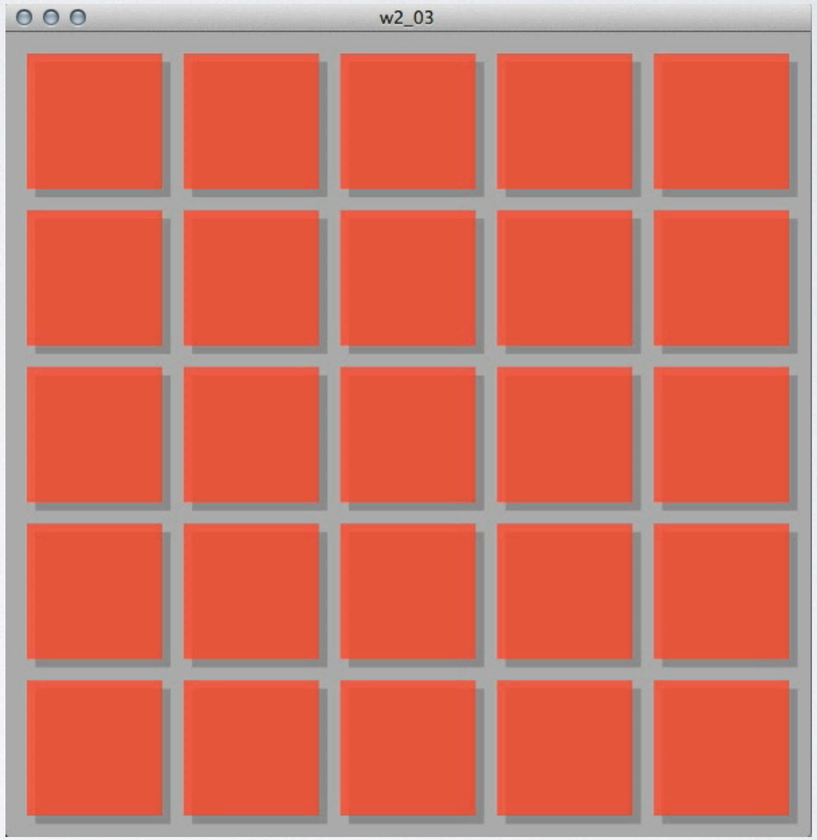
creative coding | MONASH University

w2_03 | Processing 2.1.2

w2_03

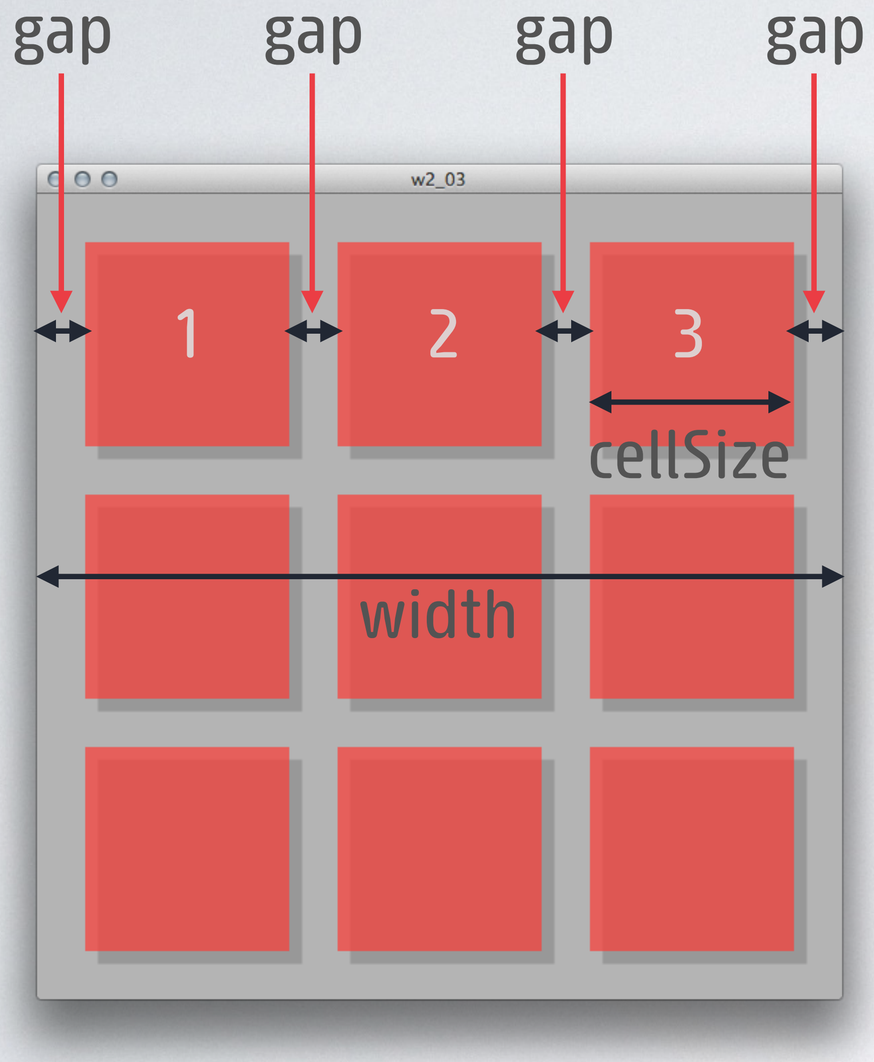
```
 * In the next iteration of the square drawing sketch, this version selects a random number of squares
 * and a random gap between them. From this it calculates the width of each square then draws the squares
 * using two nested for loops.
 *
 * A simple drop shadow is also drawn to give the illusion of depth.
 *
 */

void setup() {
  size(600, 600);
  rectMode(CORNER);
  noStroke();
  frameRate(1);  // set the frame rate to 1 draw() call per second
}


void draw() {

  background(180); // clear the screen to grey

  int num = (int) random(3, 12);   // select a random number of squares each frame
  int gap = (int) random(5, 50); // select a random gap between each square

  // calculate the size of each square for the given number of squares and gap between them
  float cellsize = ( width - (num + 1) * gap ) / (float)num;

  // print out the size of each square
  println("cellsize = " + cellsize);

  // calculate shadow offset
  float offsetX = cellsize/16.0;
  float offsetY = cellsize/16.0;


    for (int i=0; i<num; i++) {
      for (int j=0; j<num; j++) {

        fill(140, 180); // shadow
        rect(gap * (i+1) + cellsize * i + offsetX, gap * (j+1) + cellsize * j + offsetY, cellsize, cellsize);

        fill(247, 57, 57, 180); // rectangle
        rect(gap * (i+1) + cellsize * i, gap * (j+1) + cellsize * j, cellsize, cellsize);
      }
    }
} //end of draw
```

```
cellsize = 112.8
```



W2_03 : N SQUARES

creative coding | MONASH University

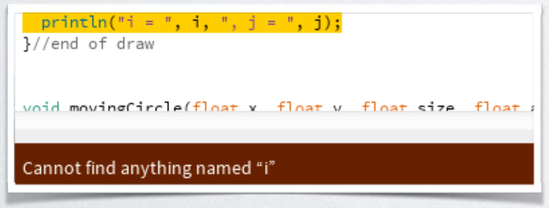



```
float cellsize = ( width - (num + 1) * gap ) / (float)num;
```

SCOPE

creative coding | MONASH University

```
for (int i=0; i<num; i++) { // col
    for (int j=0; j<num; j++) { // row
        …
        // draw the rectangle
        rect(150 + 150*i, 150 + 150*j, 100, 100);
    } // end for (j)
} // end for (i)
println(“i = “,i,”j = “, j);
```



SCOPE

creative coding | MONASH University

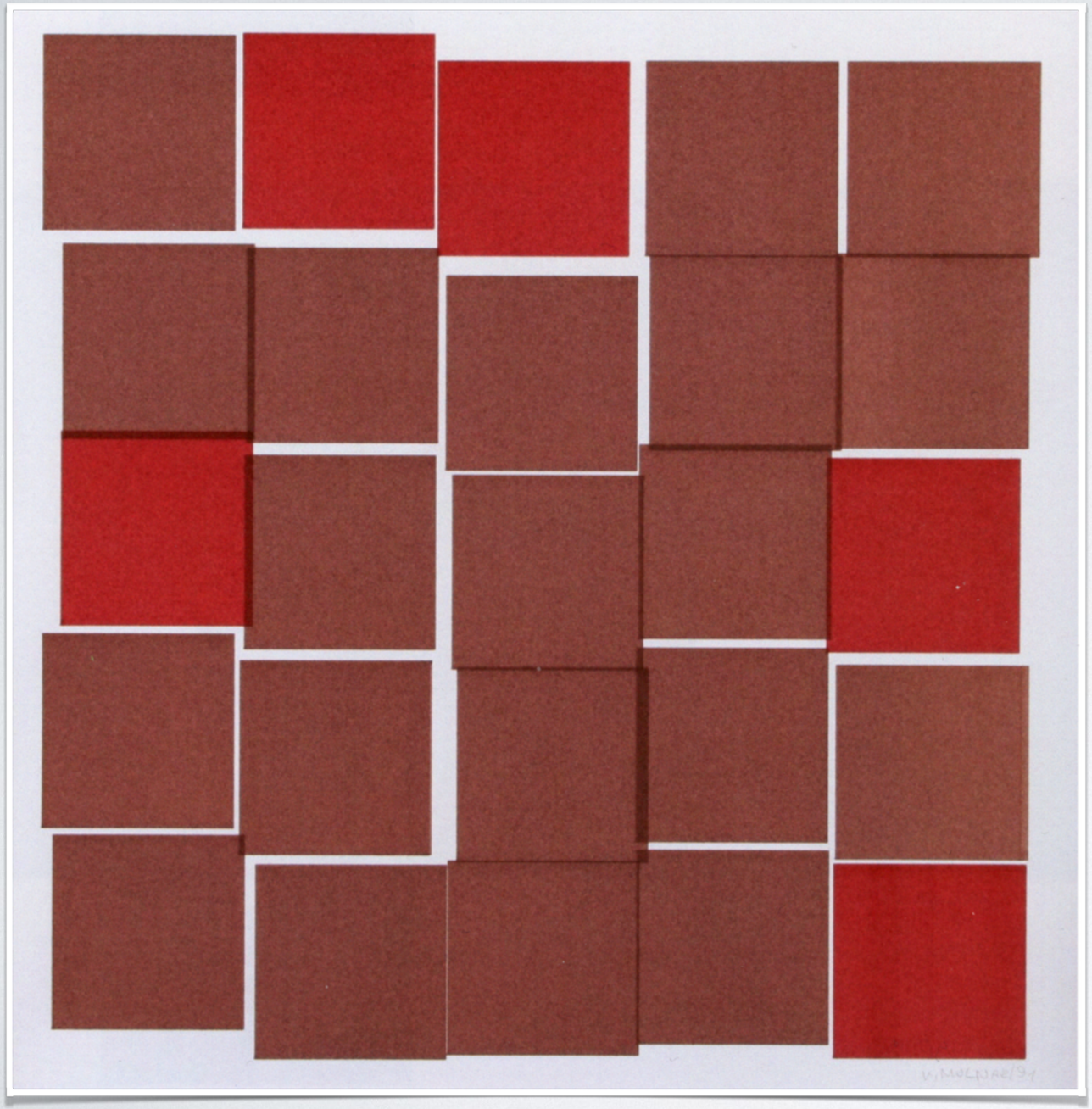
```
int i, j;
for (i=0; i<num; i++) { // col
    for (j=0; j<num; j++) { // row
      …
      // draw the rectangle
      rect(150 + 150*i, 150 + 150*j, 100, 100);
    } // end for (j)
} // end for (i)
println("i = ",i,"j = ", j);
```

i and j declared outside the code blocks of the for loops

WHILE

creative coding | MONASH University

```
int i = 0;
while (i<num) { // col
    int j = 0;
    while (j<num) { // row
      …
      // draw the rectangle
      rect(150 + 150*i, 150 + 150*j, 100, 100);
      ++j;
    } // end while (j)
    ++i;
} // end while (i)
```



Vera Molnar *25 Squares*, 1991

RANDOMSEED

creative coding | MONASH University

Call once from within setup():

```
randomSeed(hour() + minute() + second() + millis());
```

Call to get a random number between 0 and n:

```
random(n);
```

GETTING HELP

creative coding | MONASH University