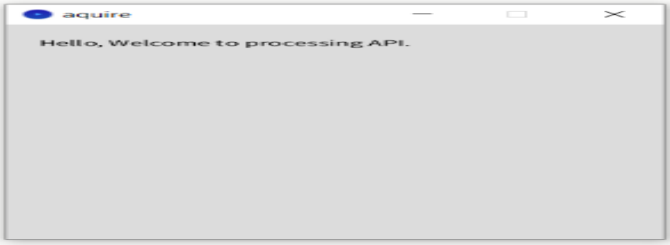
1. Aquire

//Reads the contents of a file and creates a String array of its individual lines.
//If the name of the file is used as the parameter, as in the above example,
//the file must be loaded in the sketch's "data" directory/folder.

//loadStrings(filename)

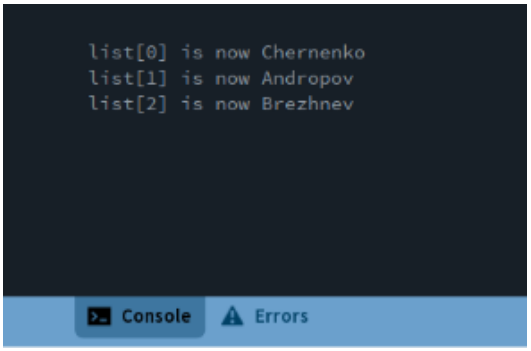
```
String[] lines;
void setup() {
 size(400, 400);
 lines = loadStrings("list.txt");
 println("There are " + lines.length + " lines");
}

void draw() {
 background(220);                                    // Clear the background

 // Display the lines on the canvas with black text color
 textSize(16);
 textAlign(LEFT);
 fill(0);  // Set text color to black

 for (int i = 0; i < lines.length; i++) {
  text(lines[i], 20, 40 + i * 20);                   // Adjust position as needed
 }
}
```



2. Parse

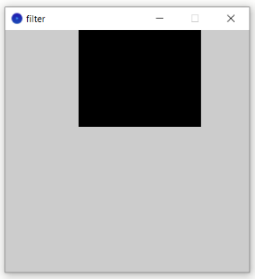
// split()

```
String men = "Chernenko,Andropov,Brezhnev";
String[] list = split(men, ',');

void setup() {
 for (int i = 0; i < list.length; i++) {
  println("list[" + i + "] is now " + list[i]);
 }
}
```



3. Filter

```
size(400, 400);
for (int i = 0; i < 160; i = i+1)
 {
  line(120, i, 320, i);
}
```



4. Mine

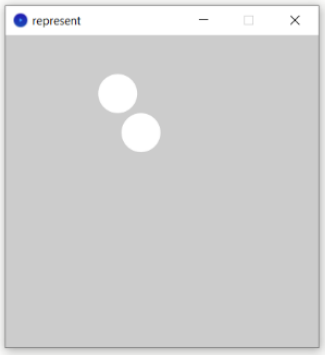
// max():-

```
int a, b;
float c;

void setup() {
  size(400, 200);
  a = max(5, 9);
  b = max(-4, -12);
  c = max(12.3, 230.24);
}

void draw() {
  background(220);
  textSize(20);
  textAlign(CENTER);

  fill(0);
  text( a, width/4, height/2 );
  text( b, width/2, height/2 );
  text( c, 3 * width/4, height/2);
}
```

5. Represent

//map():-

```
void setup() {
  size(400, 400);
  noStroke();
}

void draw() {
  background(204);
  float  x1 = map(mouseX, 0, width, 100, 150);
  ellipse(x1, 75, 50, 50);
  float x2 = map(mouseX, 0, width, 0, 200);
  ellipse(x2, 125, 50, 50);
}
```



6. Refine

//stokeweight

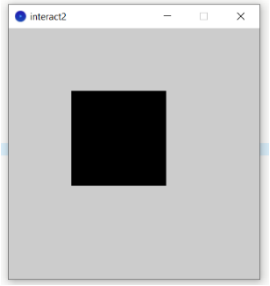
```
void setup() {
  size(400, 400);
}

void draw() {
  background(220);
  stroke(0);                          // Set stroke color to black
  strokeWeight(4);                    // Default
  line(80, 80, 320, 80);
  strokeWeight(16);                   // Thicker
  line(80, 160, 320, 160);
  strokeWeight(40); // Beastly
  line(80, 280, 320, 280);
}
```

7. Interact

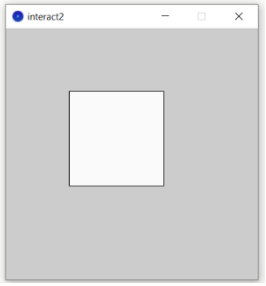
//mouseDragged()

```
// Drag (click and hold) your mouse across the
// image to change the value of the rectangle

int value = 0;

void setup()
{
 size(400,400);
}

void draw() {
  fill(value);
  rect(100,100, 150, 150);
}

void mouseDragged()
{
  value = value + 5;
  if (value > 255) {
    value = 0;
  }
}
```

Before



After mousedrag



Drawing Map with locations

```
PImage mapImage;
Table locationTable;
int rowCount;
void setup()
{
size(900, 700);
mapImage = loadImage("map.png");
// Make a data table from a file that contains
// the coordinates of each state.
locationTable = new Table("locations.tsv");
// The row count will be used a lot, so store it globally.
 rowCount = locationTable.getRowCount( );
}
void draw( ) {
 background(255);
 image(mapImage, 0, 0);
 // Drawing attributes for the ellipses.
// smooth( );
 fill(192, 0, 0);
 noStroke( );
```

// Loop through the rows of the locations file and draw the points.

```
 for (int row = 0; row < rowCount; row++) {
 float x = locationTable.getFloat(row, 1); // column 1
 float y = locationTable.getFloat(row, 2); // column 2
 ellipse(x, y, 4, 4);
 }
}
```