

INTRODUCTION

P5 & DATA ANALYSIS

CHALLENGE

PETRA ISENBERG

INFOVIS

CODING ENVIRONMENT

p5.js

[Download](#) * [Start](#) * [Reference](#) * [Libraries](#) * [Learn](#) * [Community](#)

Hello! p5.js is a JavaScript library that starts with the original goal of [Processing](#), to make coding accessible for artists, designers, educators, and beginners, and reinterprets this for today's web.

Using the original metaphor of a software sketchbook, p5.js has a full set of drawing functionality. However, you're not limited to your drawing canvas, you can think of your whole browser page as your sketch! For this, p5.js has add-on [libraries](#) that make it [easy to interact](#) with other HTML5 objects, including text, input, video, webcam, and sound.

p5.js is a new interpretation, not an emulation or port, and it is in active development. An official editing environment is coming soon, as well as many more features!

p5.js was created by [Lauren McCarthy](#) and is developed by a community of collaborators, with support from the [Processing Foundation](#) and [NYU ITP](#). © [Info](#).



Processing

[Cover](#)[Download](#)[Exhibition](#)[Reference](#)[Libraries](#)[Tools](#)[Environment](#)[Tutorials](#)[Examples](#)[Books](#)[Handbook](#)[Overview](#)[People](#)[Shop](#)[» Forum](#)[» GitHub](#)

Welcome to Processing 3! Dan explains the new features and changes; the links Dan mentions are on the [Vimeo page](#).

» [Download Processing](#)

» [Browse Tutorials](#)

» [Visit the Reference](#)

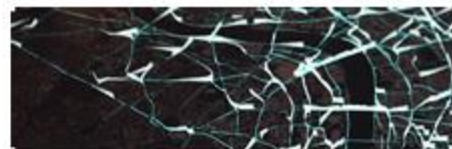
Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts. Since 2001, Processing has promoted software literacy within the visual arts and

» [Exhibition](#)



[Fluid Leaves](#)

by Reinoud van Laar

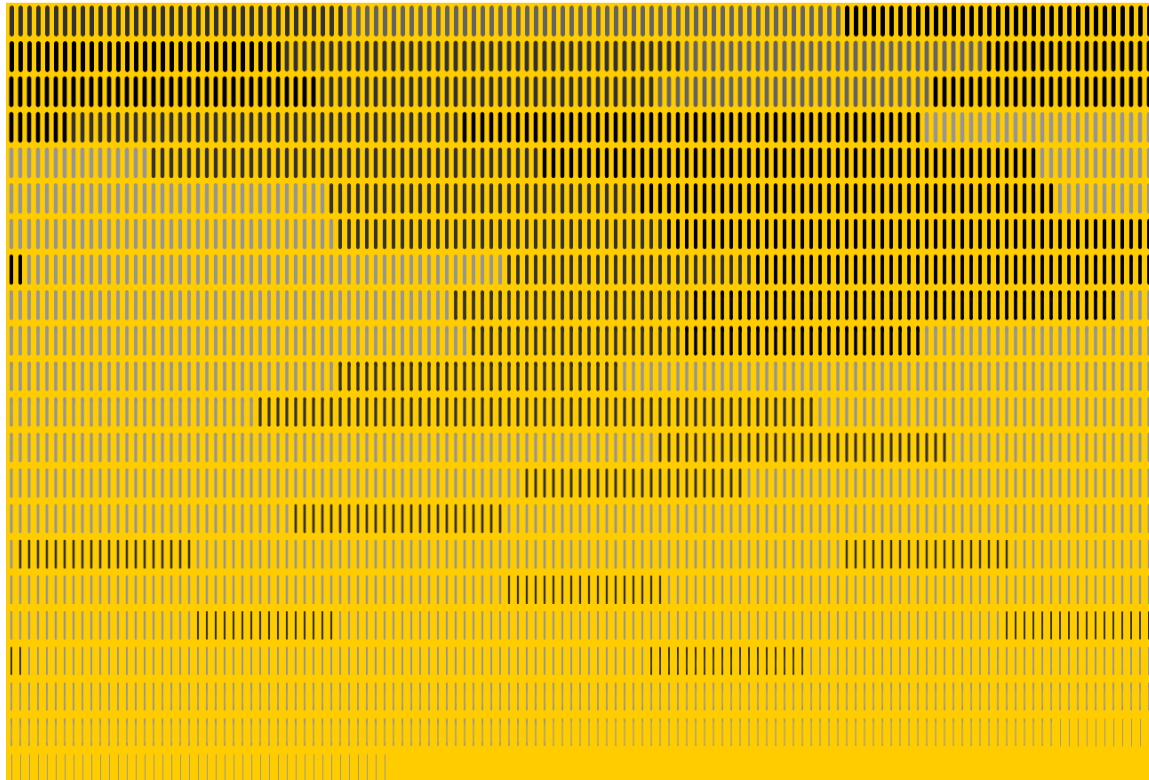


[cf.city flows](#)

by Till Nagel and Christopher Pietsch



WHAT WE WILL BE BUILDING TODAY

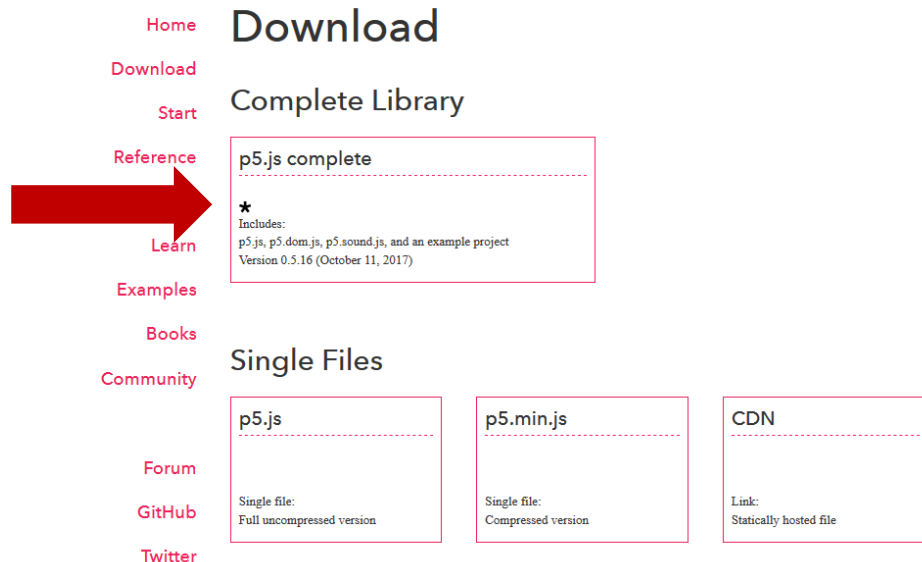


DOWNLOAD

Get your favorite
text editor

On windows, e.g.
Notepad++

p5.js



The screenshot shows the p5.js website's download page. A red arrow points to the 'Reference' link in the left sidebar. The main content area is titled 'Download' and includes a 'Complete Library' section with a box for 'p5.js complete' which lists included files and the version (0.5.16). Below this is a 'Single Files' section with three boxes: 'p5.js' (Full uncompressed version), 'p5.min.js' (Compressed version), and 'CDN' (Statically hosted file).

Home Download
Download Complete Library
Start
Reference **p5.js complete**
Learn
Examples
Books
Community
Forum
GitHub
Twitter







Includes:
p5.js, p5.dom.js, p5.sound.js, and an example project
Version 0.5.16 (October 11, 2017)

Single Files

p5.js Single file: Full uncompressed version	p5.min.js Single file: Compressed version	CDN Link: Statically hosted file
---	--	---

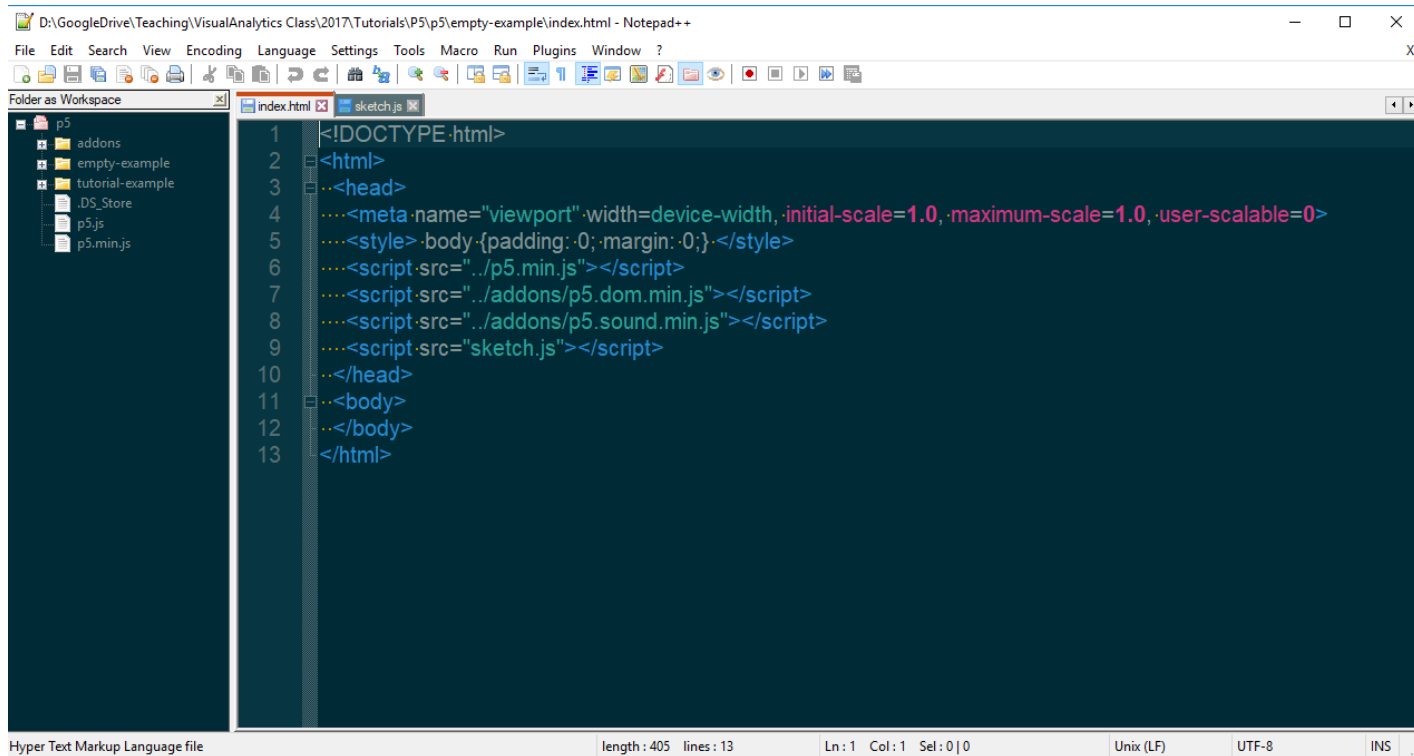
P5 COMPLETE

- Extract into a folder
- Copy the empty example
- Rename the empty example to something useful, e.g. “tutorial-example”

 addons	19/11/2017 22:21	File folder	
 empty-example	19/11/2017 22:21	File folder	
 tutorial-example	19/11/2017 22:22	File folder	
 .DS_Store	19/11/2017 22:21	DS_STORE File	7 KB
 p5.js	19/11/2017 22:21	JavaScript File	2.500 KB
 p5.min.js	19/11/2017 22:21	JavaScript File	1.159 KB

OPTIONAL - NOTEPAD++

- File -> Open folder as workspace

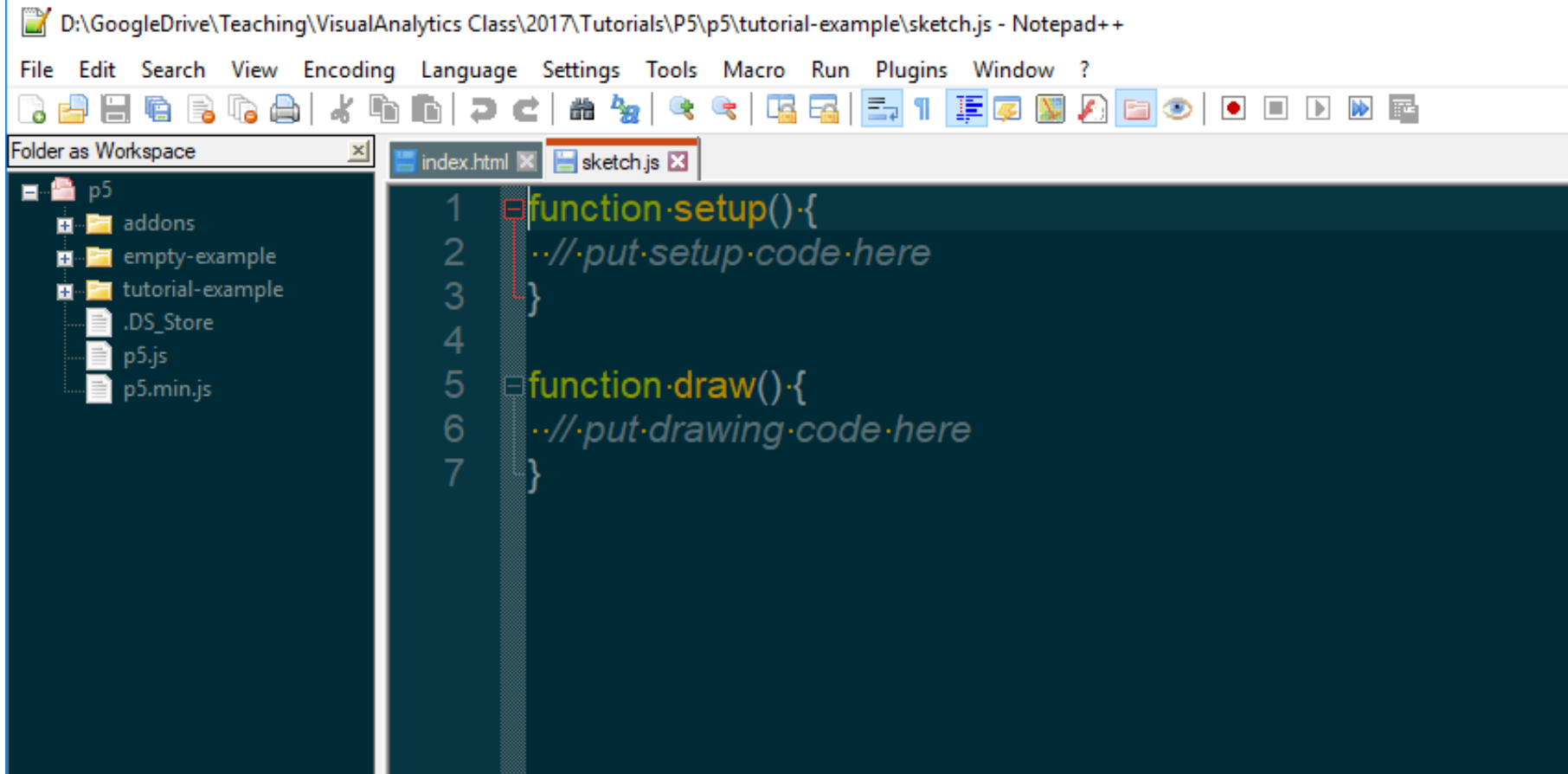


The screenshot shows the Notepad++ interface with a workspace. The left sidebar displays a folder tree for 'p5' containing subfolders 'addons' and 'empty-example', and files '.DS_Store', 'p5.js', and 'p5.min.js'. The main editor area shows the content of 'index.html' with the following code:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta name="viewport" width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0>
5 <style> body {padding: 0; margin: 0;} </style>
6 <script src="..p5.min.js"></script>
7 <script src="..addons/p5.dom.min.js"></script>
8 <script src="..addons/p5.sound.min.js"></script>
9 <script src="sketch.js"></script>
10 </head>
11 <body>
12 </body>
13 </html>
```

The status bar at the bottom indicates: Hyper Text Markup Language file, length : 405 lines : 13, Ln : 1 Col : 1 Sel : 0 | 0, Unix (LF), UTF-8, INS.

START







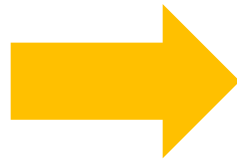
The image shows a Notepad++ window titled "D:\GoogleDrive\Teaching\VisualAnalytics Class\2017\Tutorials\P5\p5\tutorial-example\sketch.js - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and ?. The toolbar contains various icons for file operations and editing. The left sidebar shows a "Folder as Workspace" view of the "p5" directory, containing subfolders "addons", "empty-example", and "tutorial-example", and files ".DS_Store", "p5.js", and "p5.min.js". The main editor area shows the "sketch.js" file with the following code:

```
1 function setup(){
2   ..//.put.setup.code.here
3 }
4
5 function draw(){
6   ..//.put.drawing.code.here
7 }
```






DATA & LIBRARIES FOLDERS

Name ^

-  libraries
-  desktop.ini
-  index.html
-  sketch.js



Name ^

-  data
-  libraries
-  desktop.ini
-  index.html
-  sketch.js

COPY DATA FILE

- Into data folder
- Copy p5-min.js into libraries folder
- If you want to use chrome, start a webserver
- E.g. `python -m http.server` (python 3)

CHANGE HTML FILE

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
    <style> body {padding: 0; margin: 0;} </style>
    <script src="libraries/p5.min.js"></script>
    <script src="sketch.js"></script>
  </head>
  <body>
  </body>
</html>
```

```
var w = 1300;
var h = 900;

function setup() {

  createCanvas(w,h);
  noLoop();
  background(255,204,0);
}

function draw() {
}
```

```
function preload()
{
  table = loadTable("data/IEEE VIS papers 1990-2016 - Main dataset.csv");
}

function setup() {

  createCanvas(w,h);
  noLoop();
  background(255,204,0);

  console.log(table.getRowCount() + " total rows in table.");
  console.log(table.getColumnCount() + " total columns in table.");
}
```

```
function draw() {  
  
    var spacing = 10;  
    var x = 0;  
    var y = 5;  
    var length = 10;  
    var lineheight = 30;  
  
    for (var i = 0; i < table.getRowCount(); i++)  
    {  
        x = x + spacing;  
  
        if (x > w - spacing) {  
            x = x % w + spacing;  
            y = y + lineheight + 10;  
        }  
  
        line(x , y, x, y + lineheight)  
    }  
}
```

```
4 var table;
5
6 var yearCol;
7 var conferenceCol;
8 var minYear;
9 var maxYear;
10
11 var minWidth = 1;
12 var maxWidth = 5;
13
14 var fills = [0,50,100,150,200];
15 var conferences = ["VAST","InfoVis","SciVis","Vis"];
```



```
22 ▾ function setup() {  
23  
24     createCanvas(w, h);  
25     noLoop();  
26     background(255, 204, 0);  
27  
28     console.log(table.getRowCount() + " total rows in table");  
29     console.log(table.getColumnCount() + " total columns in table");  
30  
31     yearCol = table.getColumn("Year");  
32     minYear = min(yearCol);  
33     maxYear = max(yearCol);  
34  
35 }
```

```
45     for (var i = 0; i < table.getRowCount(); i++)  
46     {  
47  
48         x = x + spacing;  
49  
50     if (x > w - spacing) {  
51         x = x % w + spacing;  
52         y = y + lineHeight + 10;  
53     }  
54  
55     currentYear = yearCol[i]  
56     currentWidth = (currentYear - minYear) / (maxYear - minYear) * (maxWidth - minWidth) + minWidth;  
57  
58     strokeWeight(currentWidth);  
59  
60     line(x , y, x, y + lineHeight)  
61     }  
62
```

```
function setup() {  
  
  createCanvas(w, h);  
  noLoop();  
  background(255, 204, 0);  
  
  console.log(table.getRowCount() + " total rows in table");  
  console.log(table.getColumnCount() + " total columns in table");  
  
  yearCol = table.getColumn("Year");  
  minYear = min(yearCol);  
  maxYear = max(yearCol);  
  
  conferenceCol = table.getColumn("Conference");  
  
}
```

```
62     var conf = conferenceCol[i];
63     var index = conferences.indexOf(conf);
64     var fillColor = fills[index];
65
66     stroke(fillColor);
67
68     line(x , y, x, y + lineHeight)
69 }
70
```