

ID2116_Week3: introduction to p5.js

<https://p5js.org/>

What is p5.js ?

- A JavaScript library for “**Creative Coding**”.
 - <https://p5js.org/>
- Created by [Lauren McCarthy](#) & Developed by Open-source community.
- You can create web applications that run on a web browser.
 - graphic / sound / video / 3D / network

hello.p5.js

<http://hello.p5js.org/>

What can we create with p5.js?

- Interactive Installation: <https://visionsofthefuture.sg/exhibition/rewind>
- Interactive Web App: (e.g. **PATATAP** <https://patatap.com/>)
- NFT Art: (Generative Mask <https://generativemasks.on.fleek.co/#/>)
- Machine Learning: <https://learn.ml5js.org/#/>
- More examples can be found on
 - Open Processing <https://www.openprocessing.org/>
 - p5.js examples <https://p5js.org/examples/>

What you will be able to create after this course

F



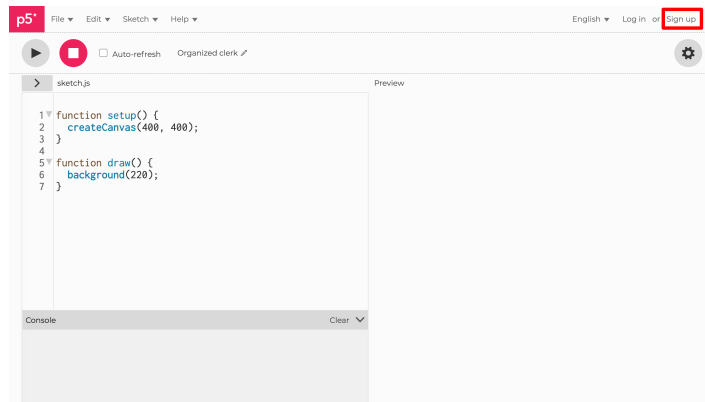
Getting Started with p5.js

Create a p5.js account

1. Open p5.js web editor

<https://editor.p5js.org/>

2. Click “Sign up” → Select Login with Google or Sign Up with your email address.



3. Now you can save and share your sketches!!!

p5.js Web Editor

The screenshot shows the p5.js Web Editor interface. At the top is a navigation bar with the p5* logo, menu items (File, Edit, Sketch, Help), language selection (English), and login/signup options. Below this is a toolbar containing a Run/Stop button (a play and stop icon), an Auto-refresh checkbox, an Organized clerk link, a Project Name input field, and a settings gear icon. The main workspace is divided into three sections: a left sidebar for Project Files (listing index.html, sketch.js, and style.css), a central Editor showing the sketch.js code with function definitions for setup() and draw(), and a right Preview window displaying a gray rectangle labeled 'Sketch'. At the bottom is a Console Terminal area with a 'Clear' button.

p5* File Edit Sketch Help English Log in or Sign up

Run / Stop Auto-refresh Organized clerk Project Name

Sketch Files

- index.html
- sketch.js
- style.css

Project Files

```
1 function setup() {  
2   createCanvas(200, 200);  
3 }  
4  
5 function draw() {  
6   background(220);  
7 }
```

EDITOR

Preview

Sketch

Console Clear

Console Terminal

p5.js Fundamentals

p5.js Program Loop

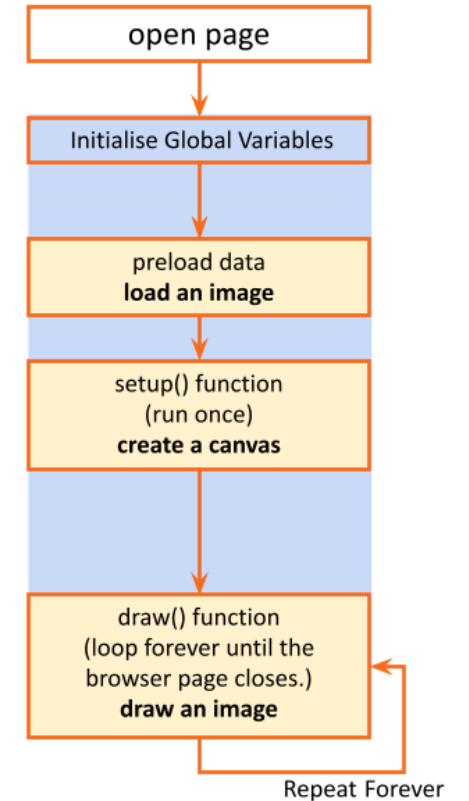
- **preload():**
load large
files(image,video,sound etc)
before the page is loaded.
- **setup():**
run once when the script is
loaded.
- **draw():**
repeatedly execute code at a
refresh-rate specified with
`frameRate()`
(e.g. `frameRate(30)`: 30 fps)

```
// Declare Global Variables
let img ;

// The preload() event runs once, when the sketch starts
function preload() {
  // load image
  img = loadImage("assets/image.jpg");
}

// The setup() event runs once, when the sketch starts
function setup() {
  // create a canvas of 400x400px.
  createCanvas(400,400);
}


// the draw() event runs over and over again,
// as long as the sketch is loaded on a browser.
function draw()
{
  //draw something here
  image(img,100,100,20,20);
}
```



Program loop: micro:bit vs p5.js

on start → `setup()` : run once

forever → `draw()` : repeat forever

<u>micro:bit</u>	p5.js
 The image shows two blue blocks from the Micro:bit programming interface. The top block is labeled 'on start' and has a single input slot on its left side. The bottom block is labeled 'forever' and has a single input slot on its left side. Both blocks have a notch on their right side, indicating they are designed to be connected in a sequence.	<pre>function setup() { createCanvas(400, 400); } function draw() { background(220); }</pre>

p5.js Programming Fundamentals

Variable, Conditionals, Loop, Functions

ID2116

Programming Fundamentals 1

Variables

Operators

Conditionals

Functions

ID2116

Programming Fundamentals 2

Functions

Arrays

Loops

Week3 P5.JS Hands-On

H01: Drawing a Smiley

let's draw a smiley face with p5.js!

- What steps do we need to go through to draw this smile?
- What are the properties of the face?
- Use the problem decomposition technique we learned last week and try to figure it out in a minute.



steps for drawing a face

- Draw the outline.
 - Paint the inside with color.
- Draw the eyes (right eye and left eye)
- Draw the mouth
- Draw the text

p5*js steps for drawing a face

1. create a drawing area: `createCanvas(width,height);`
2. Set the background color of the canvas: `background(color);`
3. Set the color and thickness of the lines `stroke(color); strokeWeight(lineWidth);`
4. Set the Fill Color `fill(color);`
5. Draw the outline: `circle(x,y,w,h);`
 - i. Draw the eyes (right eye and left eye)
`line(x1,y1,x2,y2);`
 - ii. Draw the mouth: `arc(x,y,w,h,start,stop);`
 - iii. Draw the text `textSize(size); text(text, x, y);`

p5*js smiley

https://editor.p5js.org/didny/sketches/d_QxCFCRy

```
function setup() {  
  //create a drawing area  
  createCanvas(400, 400);  
}  
  
function draw() {  
  //set background color  
  background(220);  
  
  //set color and weight of lines  
  stroke(0);  
  strokeWeight(10);  
  //set fill color  
  fill(255,255,0);  
  //draw the outline  
  circle(200,200,200);  
  
  //draw the eyes  
  line(170,150,170,220);  
  line(230,150,230,220);  
  
  //draw the mouth  
  arc(200,230,100,100,0,PI);  
  
  //draw a text  
  fill(0);  
  textSize(50);  
  text("YAY!!!", 130, 350);  
}
```

E01: Challenges

- the position and size of each part of the face is specified directly.
Define variables for these so that they can be moved freely.
- Once the variables are defined, let's try to create faces that represent other emotions by adjusting the values of the variables.
- Create functions that draw faces representing different emotions, such as `sadFace()`.
- Referring back to the MicroBits exercise, define a new variable, the hunger value, and create a conditional that changes the emotion depending on its value.

E02: Smiley Image Version

Next, let's try loading images or videos instead of the drawing with graphic functions.

To do so, we need to follow the steps below.

1. Upload image(s) to p5.editor.
2. Specify the image path and load the image into the variable.
3. Read the image drawing function and display the image.

E02: Upload Image to p5.editor

- create folder "img"
- upload images

https://www.youtube.com/watch?v=1Rs_MT82hI4

E02: Create variables to load images

- use `loadImage("path/filename")`

```
// define global variables for image
let sadFaceImage;
let neutralFaceImage;
let happyFaceImage;

function preload() {
  // load images
  sadFaceImage = loadImage("img/sad.png");
  neutralFaceImage = loadImage("img/neutral.png");
  happyFaceImage = loadImage("img/happy.png");
}
```


References 1/2:

p5*js cheat sheet

https://docs.google.com/document/d/12d8zkS8sLMzR_0oCEbfXAOM8GmB3X5LDrMvxU9ggp34/edit?usp=drivesdk

p5.js Refereces:

<https://p5js.org/reference/>

Coding Train by Daniel Shiffman

<https://thecodingtrain.com/beginners/p5js/>

References 2/2:

w3school: JavaScript Tutorial

<https://www.w3schools.com/js/>

CodeAcademy JavaScript tutorial

<https://www.codecademy.com/learn/introduction-to-javascript>

p5.play

<https://molleindustria.github.io/p5.play/>