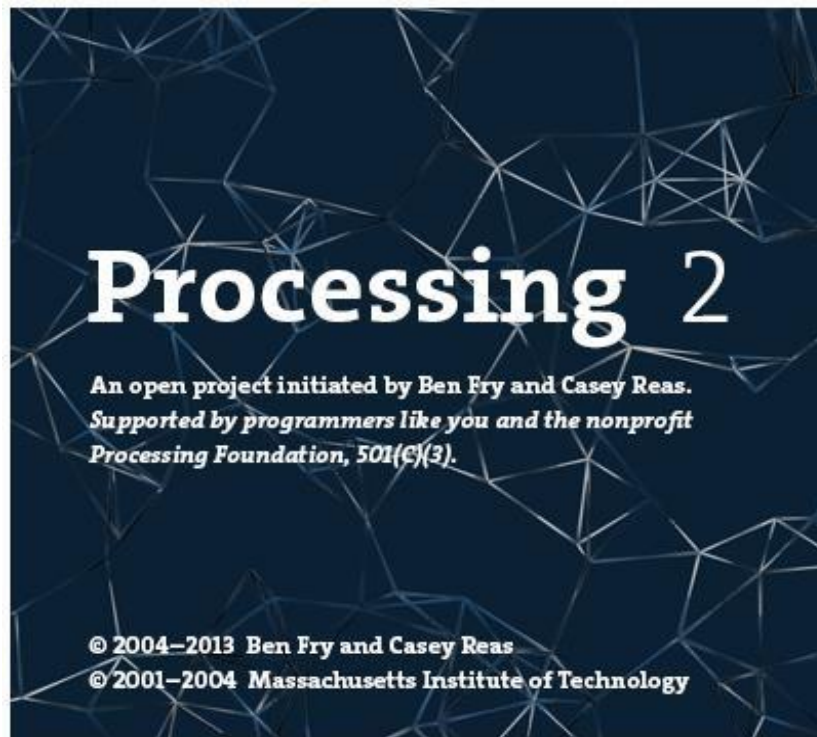




Processing: [Librerías y Matrices](#)

Processing: Librerías y Matrices



Insertar nuevas librerías a Processing



Manejar la librería ControlP5: Botones y Contenedores



Lectura de matrices



Almacenamiento de datos



Visualización de datos

Librerías Básicas

Nos ayudan a mejorar las capacidades de programación dentro de Processing con herramientas orientadas a diferentes temas::

Serial

Permite la comunicación entre un dispositivo electrónico conectado desde un Puerto COM hacia Processing para enviar o recibir datos

Sintaxis:

```
import processing.serial.*;
```

ControlP5

Permite insertar botones, sliders, graficas, entre otros. De esta manera se tiene mejor control de una interfaz y permite agregar mayores funciones a los eventos de teclado y mouse

Sintaxis:

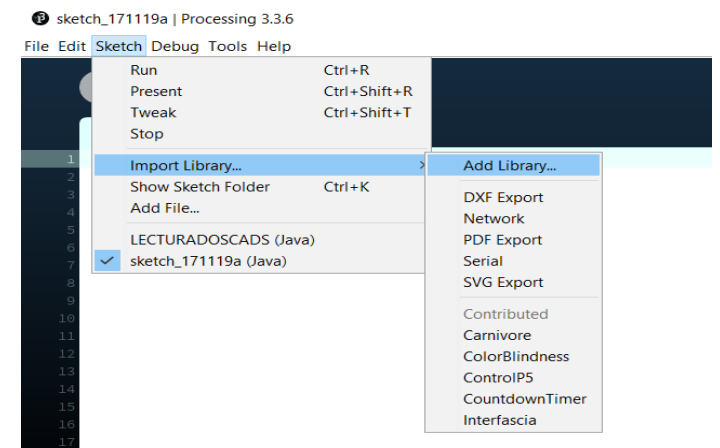
```
import controlP5.*;
```

```
ControlP5 cp5;
```

```
cp5= new ControlP5(this);
```

Importar nuevas librerías

Si es necesario agregar componentes extras y se desconoce de todas las herramientas disponibles, dentro del sketch se pueden buscar en línea.



Control P5

Botones

Permite agregar botones de forma simple y ordenada

Sintaxis:

```
cp5.addButton("Nombreboton")
    .setPosition(x,y)
    .setSize(w,h)
    .setCaptionLabel("Nombre en interfaz");
;
```

```
public void Nombreboton(){
    //instrucciones de boton
}
```

w -> ancho de la ventana (int)

h -> alto de la ventana (int)

Toogle

Botón de dos estados, encendido o apagado

Sintaxis:

```
cp5.addToggle("NombreToogle")
    .setValue(0) // valor de 1 o 0 al presionar
    .setPosition(x,y)
    .setSize(w,h)
    .setCaptionLabel("Nombre en interfaz");
;
void AUTOMATIC(boolean dato) {
    if(dato==true){
        //instrucciones
    }
}
```

Control P5

Slider

Es una herramienta que permite seleccionar dentro de una barra cierto valor en un rango seleccionado. Además, puede ser utilizado como un contenedor de datos

Sintaxis:

```
cp5.addSlider("slider")
    .setPosition(x,y)
    .setSize(w,h)
    .setRange(min,max)
    .setValue(0)
    .setNumberOfTickMarks(numero de marcas)
    .setCaptionLabel("Nombre en interfaz");
;
void slider (int dato){
    //instrucciones
}

w -> ancho de la ventana (int)
h -> alto de la ventana (int)
```

```
cp5.getController("slider").setValue(Valor);
```

Lectura Matrices

La mayoría de los datos se encuentran en relación de matrices (fila y columnas). Desde processing podemos leer como guardar en diferentes formatos

LECTURA

Sintaxis:

```
//cargar datos
Table Nombre_variable;
Nombre_variable=loadTable("Archivo.csv","header");
//leer =, forma 1
for (TableRow row:Nombre_variable.rows()){
    dato=row.getFloat("header"); // pueden ser otro tipo de datos
}
//forma 2
for(i=0;i<=tope;i++)
{
    TableRow row = Nombre_variable.getRow(i);
    X=((row.getFloat("header")
}
```

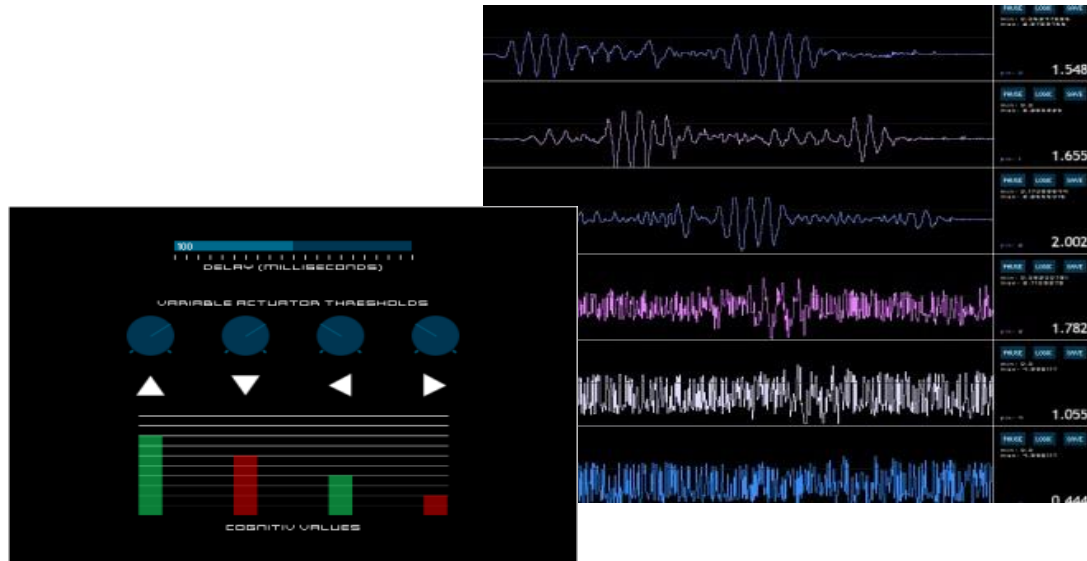
ESCRITURA

Sintaxis:

```
PrintWriter output;
output = createWriter("Archivo.csv");

output.flush(); //abrir archivo
output.println(Variable1+", "+variable2);
output.close(); // Finishes the file
```

Visualización Datos



Generalmente se tendrá dos formas de visualizar información. La primera es en tiempo real (datos adquiridos al mismo tiempo ser visualizados) y la segunda es con bases de datos guardadas, las mismas que se debe ser normalizadas y cambiar todo a positivos para poder visualizar en Processing. Posteriormente, podremos utilizar las herramientas vistas con anterioridad.