

Processing Tutorials

Tutorial 1

Introduction to Processing

- ❖ Processing is a programming language, development environment, and online community.
- ❖ It is initially created to serve as a software sketchbook and to teach computer programming fundamentals within a visual context.
- ❖ Website: www.processing.org or processing.org

Download Processing

- ❖ V 2.2.1 (19 May 2014)

- ❖ Windows 64/32-bit
- ❖ Linux 64/32-bit
- ❖ Mac OS X

[Download Processing](#). Please consider making a donation to the Processing Foundation before downloading the software.

Processing is open source, free software. All donations fund the [Processing Foundation](#), a nonprofit organization devoted to advancing the role of programming within the visual arts through developing Processing.

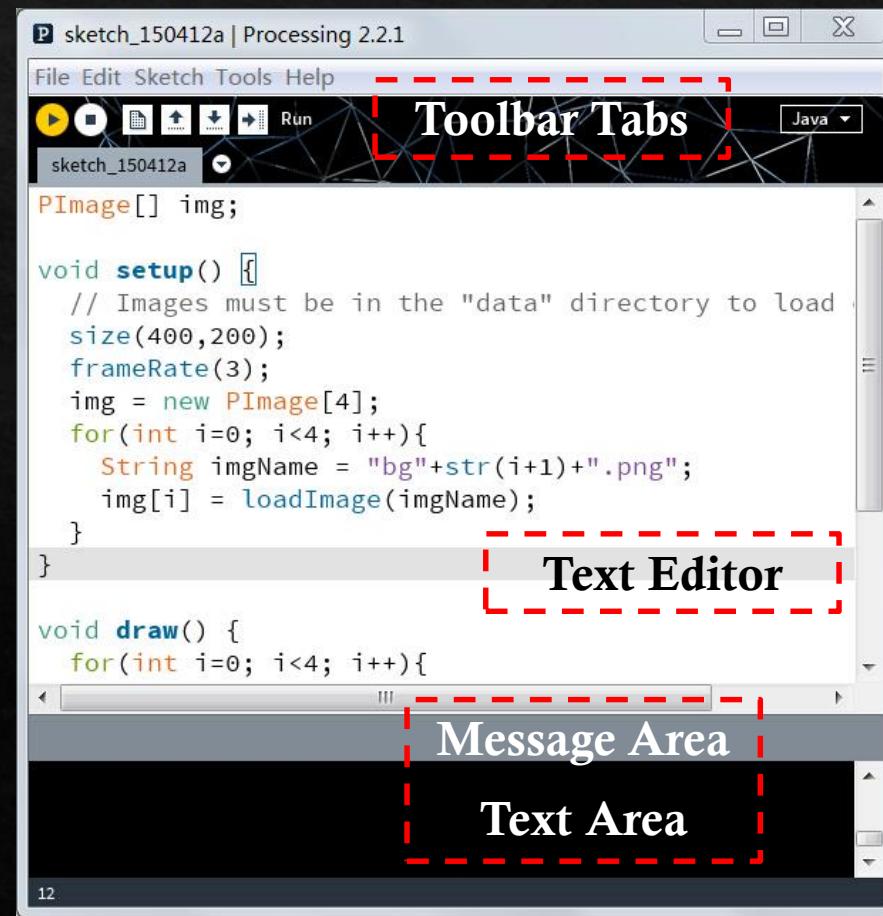
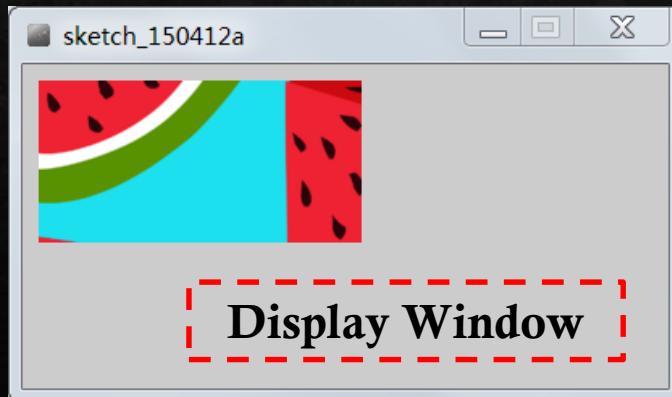
No Donation \$10 \$25 \$50 \$100 \$

[Donate & Download](#)

- ❖ Download website: www.processing.org/download/
- ❖ <http://sse.tongji.edu.cn/yingshen/course/HCI2015Spring/software/processing-2.2.1.zip> (Windows 64-bit)

Processing Development Environment

- ❖ PDE consists of a text editor, a message area, a text console, tabs for managing files, a toolbar with buttons for common actions, and a series of menus.



Tool Bar Buttons

- ❖  Run
- ❖  Stop
- ❖  New
- ❖  Open
- ❖  Save
- ❖  Export

Sketches and Sketchbook

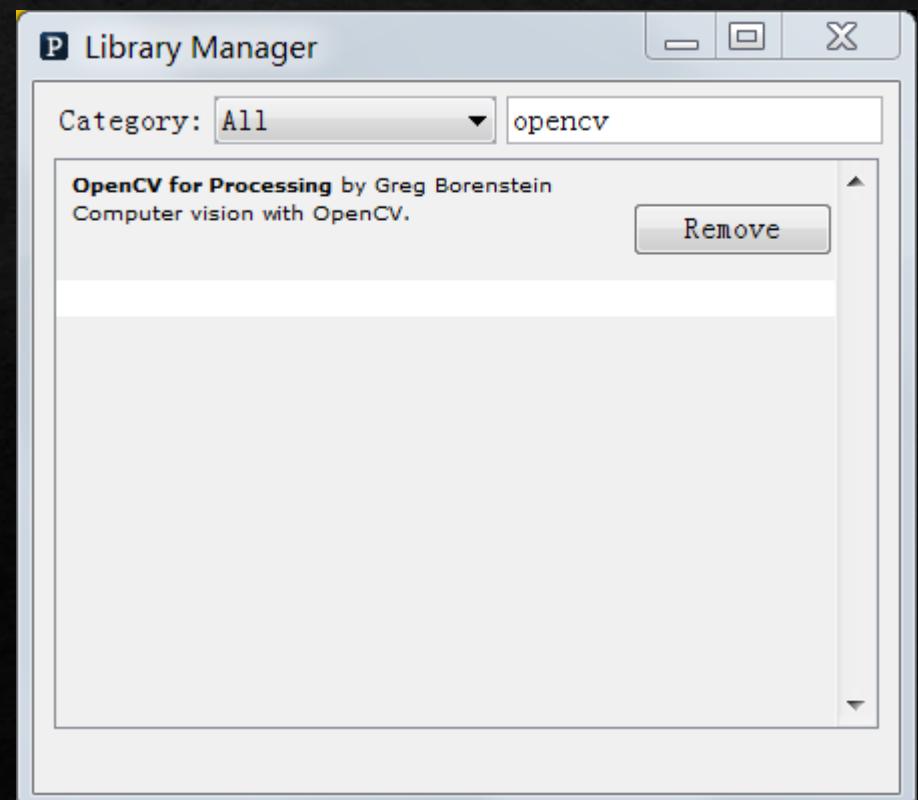
- ❖ All Processing projects are called sketches.
- ❖ Each sketch has its own folder.
- ❖ The main file for each sketch has the same name as the folder and is found inside.

Don't forget to save your sketches first!

- ❖ Add other resources (fonts or images) into current project
 - ❖ “Sketch” menu → "Add File..."
 - ❖ A “data” folder will be created.
 - ❖ All images, fonts, sounds, and other data files loaded in the sketch must be in “data” folder.

Adding Libraries

- ❖ Add contributed libraries automatically
 - ❖ “Sketch” → “Import Library” → “Add Library”
 - ❖ Type in “opencv” in the search box and select “OpenCV for Processing”
 - ❖ Click “Install” button



Adding Libraries

- ❖ Add contributed libraries (e.g. OpenCV library) manually
 - ❖ “File” → “Preferences”: find your sketchbook location. E.g.: “C:\Users\Ying Shen\Documents\Processing”
 - ❖ Open “C:\Users\Ying Shen\Documents\Processing\libraries”
 - ❖ Copy “opencv_processing.zip” into the *libraries* folder
 - ❖ You will need to create the *libraries* folder if this is your first contributed library.
 - ❖ Unzip “opencv_processing.zip” in the *libraries* folder.
- ❖ Other contributed libraries can be found here:
processing.org/reference/libraries/

Processing Reference and Examples

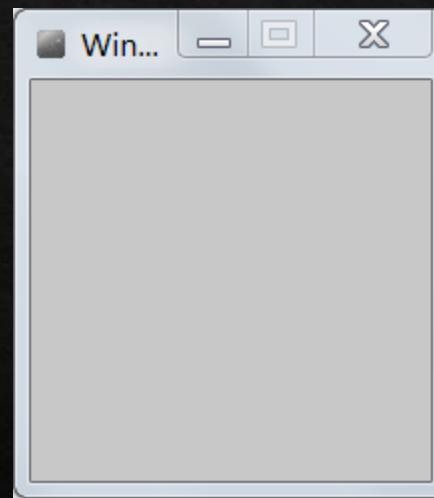
- ❖ <https://processing.org/reference/>
- ❖ <https://processing.org/examples/>

Example 1

Initialize a Window

Initialize a Window

```
void setup() {  
    size(200, 200);  
    background(200);  
}
```

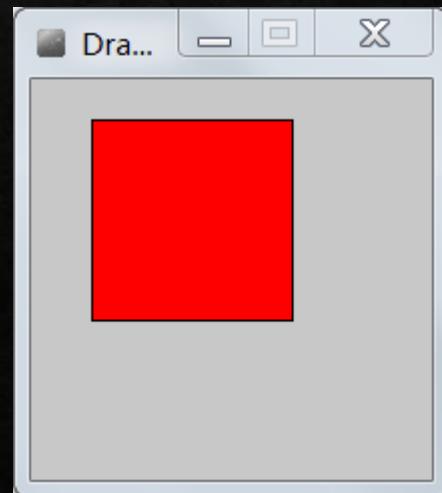
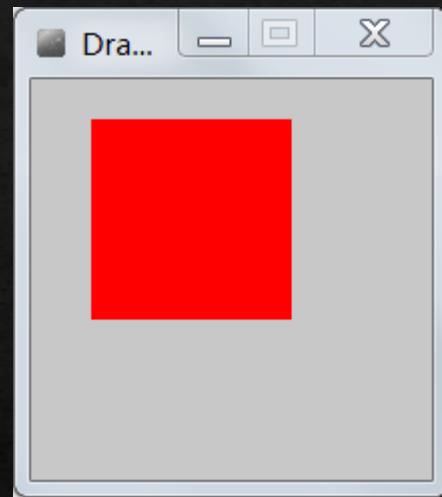


Example 2

Drawing on the Window

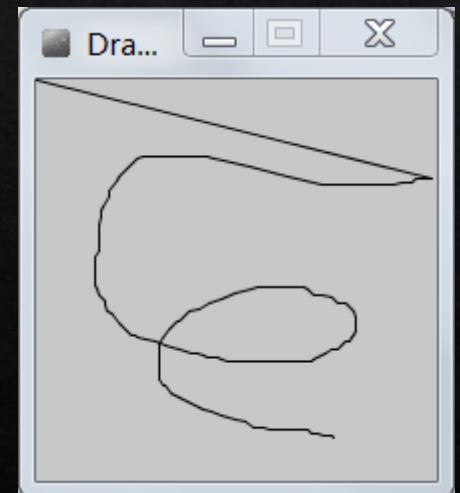
Draw a Rectangular

```
void setup() {  
    size(200, 200);  
    background(200);  
    frameRate(4);  
}  
  
void draw(){  
    noStroke(); //stroke(0);  
    fill(255,0,0);  
    rect(30, 20, 100, 100);  
}
```



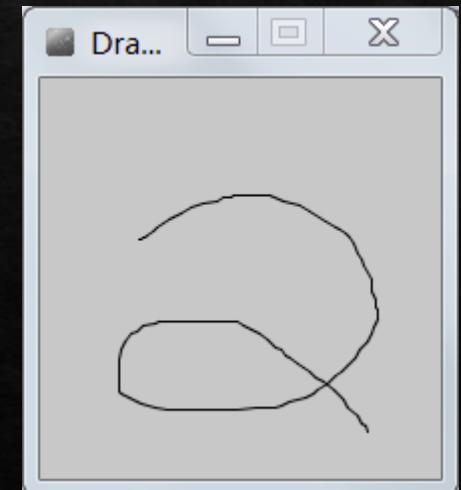
Draw Lines using Mouse

```
void setup() {  
    size(200, 200);  
    background(200);  
}  
  
void draw(){  
    line(mouseX,mouseY,pmouseX,pmouseY);  
    println(mouseX,mouseY);  
}
```



Draw Lines using Mouse (improved)

```
void setup() {  
    size(200, 200);  
    background(200);  
}  
  
void draw(){  
    if(mousePressed == true){  
        line(mouseX, mouseY, pmouseX, pmouseY);  
    }  
}
```



Example 3

Load Images

Load Images

```
PImage img;  
  
void setup() {  
    // Images must be in the "data" directory to load  
    correctly  
    size(640, 360);  
    img = loadImage("moonsurface.jpg");  
}  
  
void draw() {  
    image(img, 0, 0);  
}
```

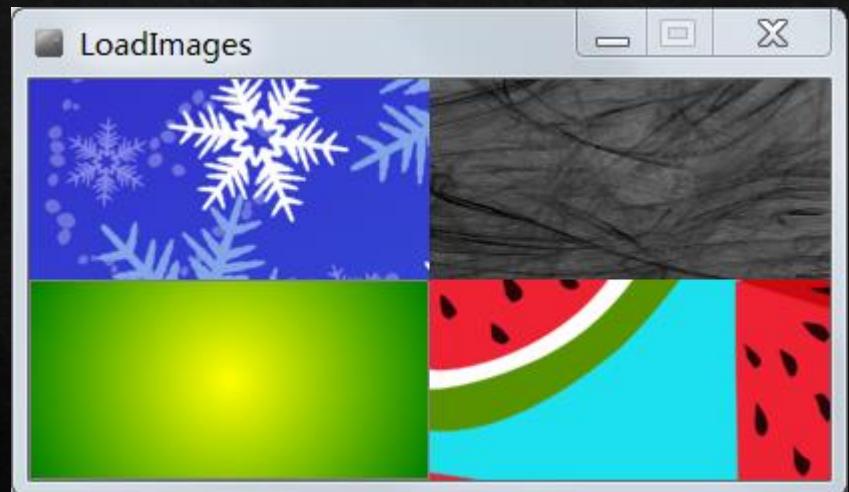


Load Images

```
PIImage[ ] img;  
void setup() {  
    size(400,200);  
    img = new PImage[4];  
    for(int i=0; i<4; i++)  
        img[i] = loadImage("bg"+str(i+1)+".png");  
}
```

Load Images

```
void draw() {  
    image(img[0],0,0);  
    image(img[1],200,0);  
    image(img[2],0,100);  
    image(img[3],200,100);  
}
```



Exercise

Exercise

1. Download Processing package
2. Run previous examples
3. Finish the following tasks:

Task 1:

- ◊ Draw a rectangular on the screen;
- ◊ The rectangular will move with the mouse.

Task 2:

- ◊ Load a series of images;
- ◊ Displayed one image on the window per second.