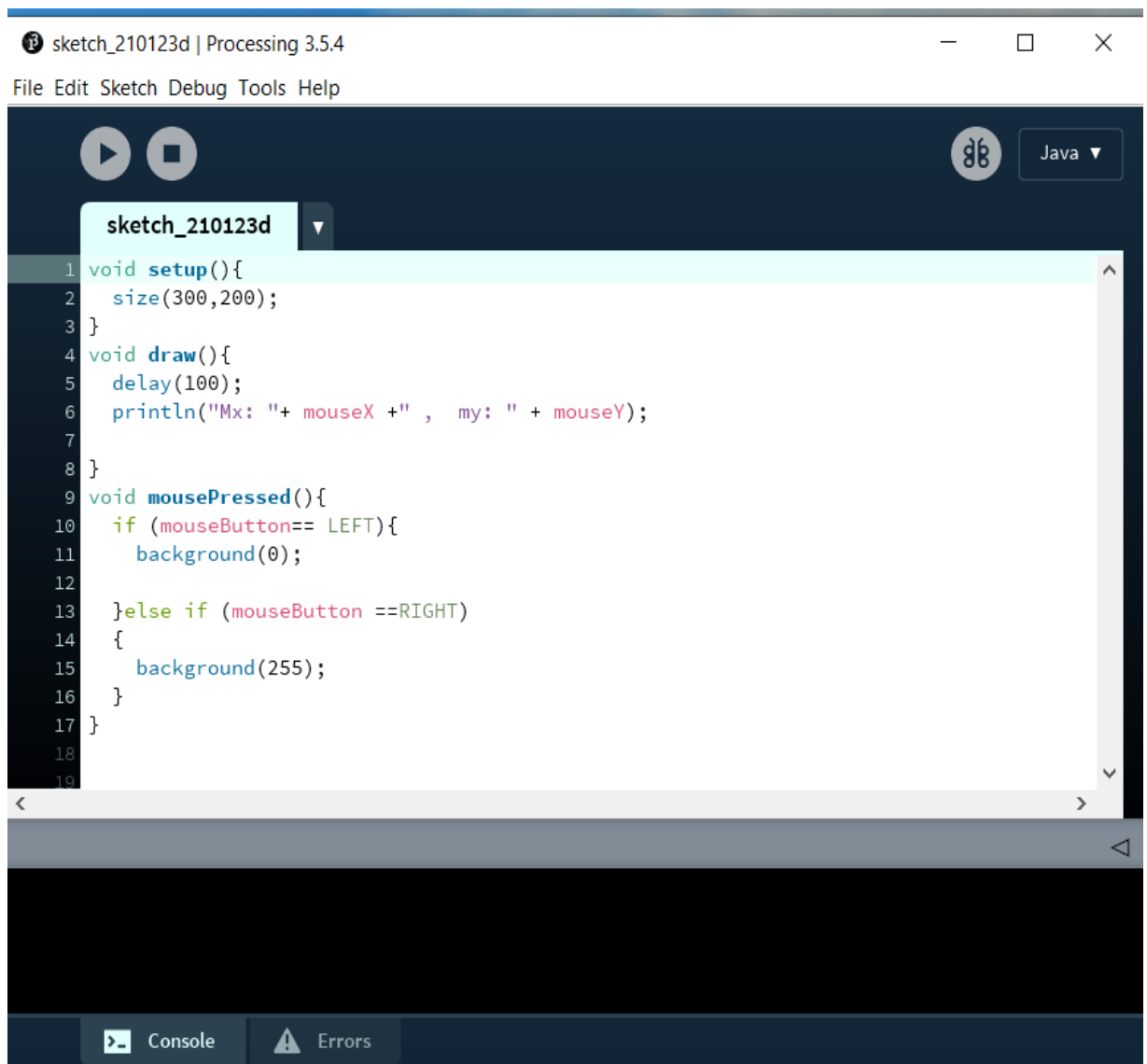


មេរៀនទី៥

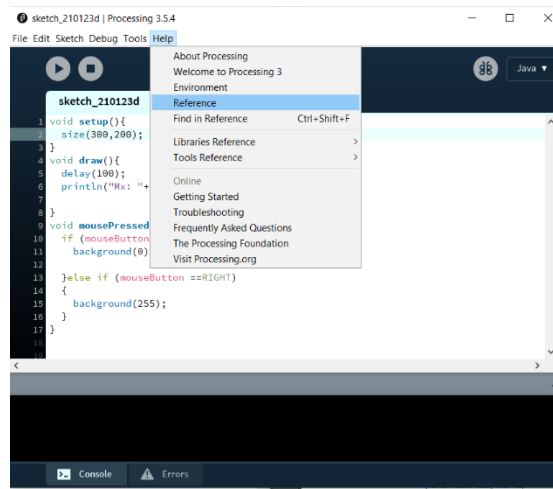
Software Processing

I. សេចក្តីផ្តើម

processing ជា Open software មួយរបស់ Processing.org ដែលអ្នកទាញយកមកប្រើប្រាស់ ដោយមិនចាំបាច់បង់ប្រាក់។ processing ប្រើសម្រាប់បង្កើត GUI ។ Processing ជា Software តូចមួយ ដែលគេនិយមយកមកប្រើជាមួយ Arduino ព្រោះទំរង់ភាសារបស់វាស្រដៀងគ្នា នឹង Arduino ដែលជាហេតុងាយស្រួលសម្រាប់អ្នកប្រើប្រាស់។ សូមបញ្ជាក់ថាយើងលើកយកតែផ្នែកខ្លះមកសិក្សាតែប៉ុណ្ណោះ។ ទំរង់ IDE processing 3 ដូចខាងក្រោម

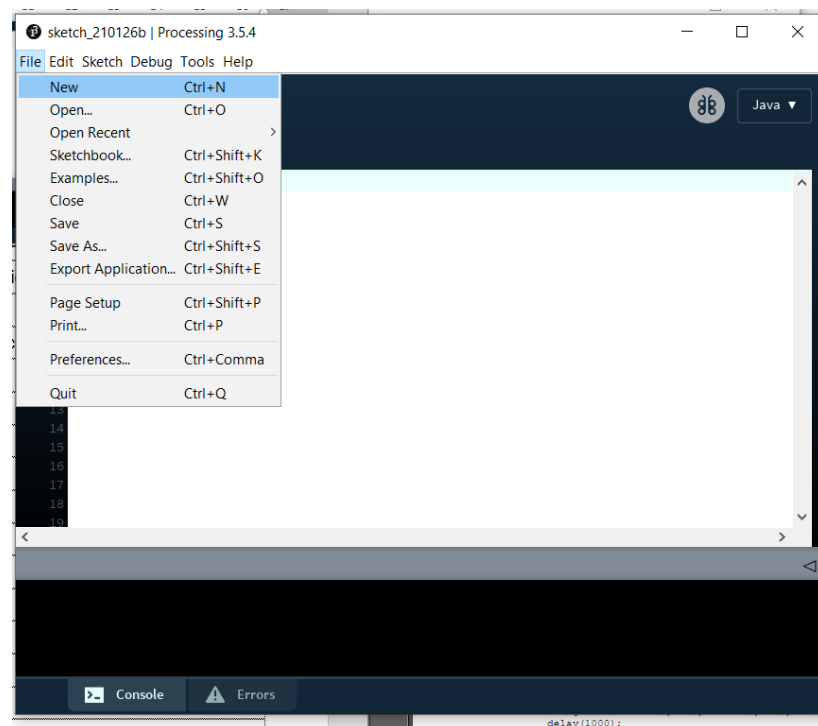


សម្រាប់ព័ត៌មានលំអិតមាននៅក្នុង Reference របស់វាដោយយើងចូលទៅក្នុង
Help→Reference



នៅក្នុងការសរសេរកូដវិញ្ញាសាស្រដៀង នឹង Arduino ដែរគឺ void setup() និង void draw() ។
II. របៀបបង្កើត Project

ដើម្បីបង្កើត Project យើងត្រូវចូលទៅ program Processing 3 →File→New



```
void setup(){
    size(640,300);
```

```

    }
    void draw(){
        background(#FFFFFF);
        fill(255,0,10);
        textSize(50);
        text("hello Bunchun",20,50);
    }

```

size(640,300); សម្រាប់កំណត់ទំហំ screen

background(0,0,0); ដាក់ background អោយ screen

stroke(255); កំណត់ពណ៌

rect(10, 30, 45, 21); គូសនូវចតុកោណកែង

line(10,height/2,width-10, height/2); គូសនូវបន្ទាត់

ចូរអ្នកចូលទៅកាន់ Reference របស់វាដើម្បីអាចយល់បន្ថែមអំពីការប្រើប្រាស់នូវ 2D draw មួយចំនួនថែមទៀត។

Example

សរសេរកូដគូសវាសអោយចេញជាក្រាហ្វស៊ីនុសសូអ៊ីតស្រដៀងខាងក្រោម

```

float A=150;
float F=20;
float phi=0;
void setup(){
    size(640,380);
    background(#FFFFFF);
}
void draw(){

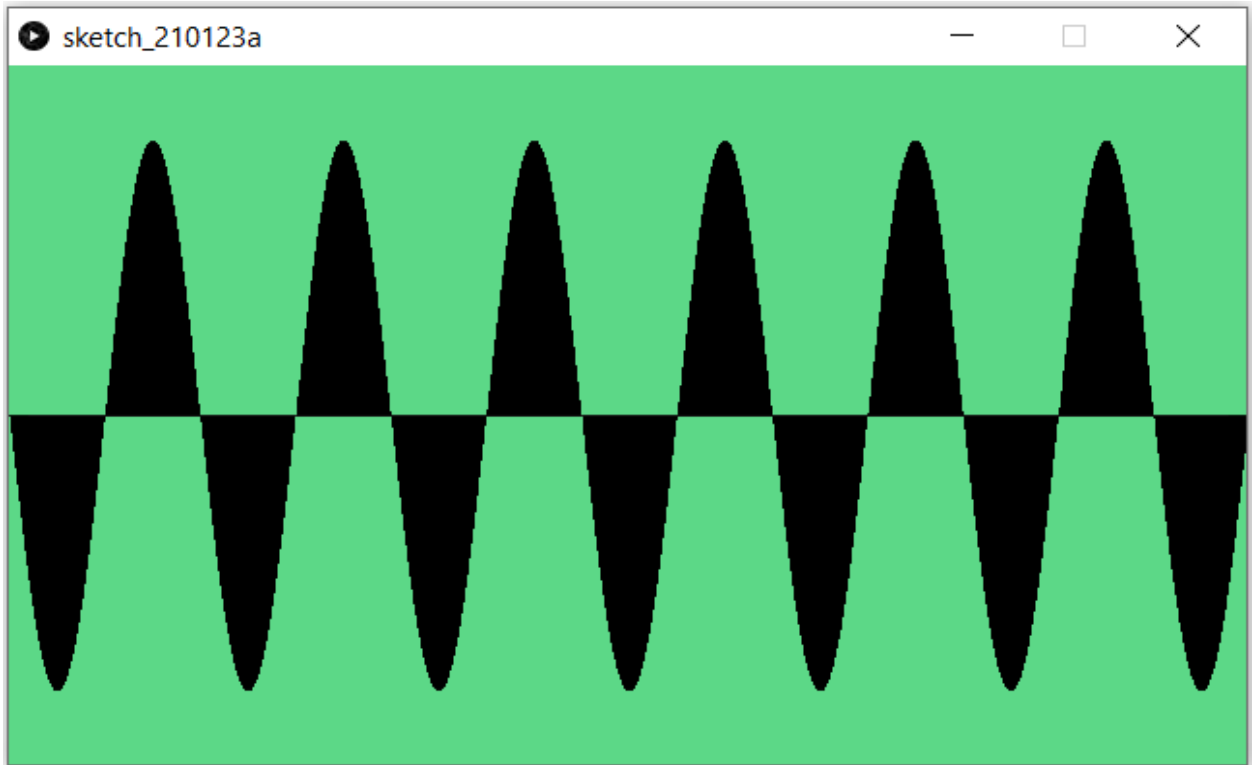
    background(random(0,255),random(0,256),random(0,256));
    stroke(255);
    line(10,height/2,width-10,height/2);
    for (int t=0;t<width;t++){
        int y=int(A*sin(2*3.14*F*t)+phi);
        stroke(0);
        line(t,height/2,t,height/2-y);
    }
}

```

```

        delay(1500);
    }

```



III.របៀបប្រើប្រាស់Tool (View Object ឬ Tool មួយចំនួន)

-Text ជាTool ប្រើសម្រាប់ដាក់អក្សរទៅកាន់ផ្ទាំង Background

```

size(640,300);
fill( 255,0,0);
textSize(50);
text("Hello arrowDot",30,50);

```

-កាបផ្ដើតប៊ុតុងតែ

III.ការបង្កើត View Tool

-ការបង្កើត Text

ខាងក្រោមនេះជាកូដសម្រាប់សរសេរអក្សរលើផ្ទាំងScreen

```

size(640,300);
fill(255,0,0);
textSize(50);

```

```

        text("Hello ",30,50);
-ការបង្កើតប៊ូតុង(Button)
import controlP5.*;
Serial ports;
ControlP5 cp5;
Button btn1;
        Button btn2;
void setup(){
    size(640,320);
    ports=new Serial(this,"COM9",9600);
    cp5=new ControlP5(this);
    btn1=cp5.addButton("bt_On")
        .setCaptionLabel("On")
        .setColorActive(color(34,23,23))
        .setColorCaptionLabel(color(23,44,23))
        .setColorForeground(color(12,253,23))
        .setColorBackground(color(0,255,0))
        .setPosition(50,50)
        .setSize(100,50)
        .setFont(createFont(" ",50));
    btn2=cp5.addButton("bt_off")
        .setCaptionLabel("off")
        .setPosition(50,120)
        .setSize(100,50)
        .setFont(createFont(" ",50))
        .setColorActive(#D60B26)
        .setColorCaptionLabel(#0B42D6)
        .setColorForeground(color(12,253,23))
        .setColorBackground(color(0,255,0))
        ;
}
void draw(){
}

```

```

void bt_On(){
    ports.write("on");
}

void bt_off(){
    ports.write("off");
}

```

VI ការប្រើប្រាស់ Keyboard and Mouse

keyboard and Mouse ត្រូវបានយើងប្រើជាញឹកញាប់ និងទូទៅ ជាមួយនឹង Processing យើងអាច control វាបានតាមចិត្តដែលយើងចង់បាន ជាជាងប្រើមុខងារដែលគេកំណត់សម្រាប់យើងស្រាប់។

-keyboard

មានFunction មួយចំនួនដូចខាងក្រោម

-keyPressed() វាដំណើរការនៅរៀងរាល់ពេលដែលយើងចុច Button

Keyboard

-keyTyped() វាដំណើរការនៅរៀងរាល់ពេលដែលអ្នកចុច Button Keyboard

ណាមួយលើកលែងតែ Key Shift, Alt, Ctrl, Left, Right, Up, down ។

-keyReleased() វាដំណើរការនៅរៀងរាល់ពេលដែលអ្នកលែងដៃពី Button

Keyboard ណាមួយ។

-key វាជាអញ្ញត្តិដែលប្រមូលផ្តុំដោយតំលៃ Key ទាំងអស់។

example

```

void setup()
{

}

void draw() {
}

void keyTyped()
{
    println(key);
}

void keyPressed()
{

```

```

        println("keyPressed");
    }
    void keyReleased() {
        println("keyReleased");
    }

```

-Mouse

វាមាន៦ Function

- mouseClick() កើតឡើងពេលដែល Mouse ត្រូវបានចុចហើយប្រលែងវិញ
- mouseDragged() កើតឡើងដែល Mouse cursor ចុចហើយអ្នកអោយផ្លាស់

ទី

- mouseMoved() កើតឡើងពេលដែល Mouse cursor អ្នកអោយ

ផ្លាស់ទី

- mousePressed() កើតឡើងពេលដែល Mouse ត្រូវបានចុច
 - mouseReleased() កើតឡើងពេលដែល Mouse ត្រូវបានប្រលែងវិញ
 - mouseWheel() កើតពេលដែល Mouse Wheel ត្រូវបានបង្វិល
- រីឯអញ្ញាតរបស់វាវិញរួមមាន

- MouseButton
- MousePressed
- MouseX
- MouseY
- pmouseX
- pmouseY

example

```

void setup()
{
    size(640,320);
}
void draw() {
    rect(25,25,50,50);
}
void mousePressed()
{
    if (mouseButton==LEFT)

```

```

    {
        fill(0);
    }else if(mouseButton==RIGHT)
    {
        fill(200);
    }
    else{
        fill(100);
    }
}

```

V ការប្រើប្រាស់ Serial Communication

ក្នុងចំណុចនេះយើងនឹងសិក្សាអំពីការតភ្ជាប់ពី Processing ក្នុងកុំព្យូទ័រទៅកាន់ Arduino

ដែលនៅខាងក្រៅតាមរយៈ Serial communication រឺ COM port ។ដោយប្រើប្រាស់ Serial Library របស់ Processing យើងនឹងមានភាពងាយស្រួលក្នុងការតភ្ជាប់នេះ។រីឯមុខងារក៏ស្រដៀងគ្នា នឹង Serial របស់ Arduino ដែរ។

Methods

- Serial
- available()
- read()
- readChar()
- readBytes()
- readBytesUntil()
- readString()
- buffer()
- bufferUntil()
- last()
- lastChar()
- write()
- clear()
- stop()
- list()
- serialEvent()

Example1 serial Communication

នៅក្នុងឧទាហរណ៍នេះអ្នកនឹងបានឃើញពីរបៀបដែល Processing ធ្វើការ Control ទៅកាន់ LED ដើងទី១៣ របស់ Arduino តាមរយៈ Serial port របស់វា។ យើងនឹងត្រូវសរសេរ កូដ ចំនួនពីរ គឺមួយនៅក្នុង Arduino និងមួយទៀតនៅក្នុង Processing ។

Processing code

```
import processing.serial.*;
import controlP5.*;

Serial ports;
ControlP5 cp5;
Button btn1;
Button btn2;

void setup(){
  size(640,320);
  ports=new Serial(this,"COM9",9600);
  cp5=new ControlP5(this);
  btn1=cp5.addButton("bt_On")
    .setCaptionLabel("On")
    .setColorActive(color(34,23,23))
    .setColorCaptionLabel(color(23,44,23))
    .setColorForeground(color(12,253,23))
    .setColorBackground(color(0,255,0))
    .setPosition(50,50)
    .setSize(100,50)
    .setFont(createFont(" ",50));
  btn2=cp5.addButton("bt_off")
    .setCaptionLabel("off")
    .setPosition(50,120)
    .setSize(100,50)
    .setFont(createFont(" ",50))
    .setColorActive(#D60B26)
    .setColorCaptionLabel(#0B42D6)
    .setColorForeground(color(12,253,23))
    .setColorBackground(color(0,255,0)) ;
```

```
}  
void draw(){  
}  
void bt_On(){  
    ports.write("on");  
}  
void bt_off(){  
    ports.write("off");  
}
```



The End