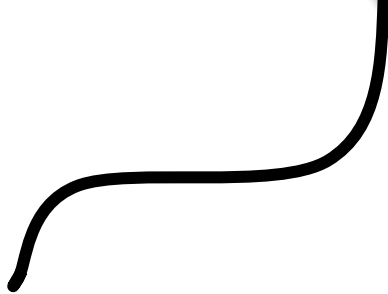
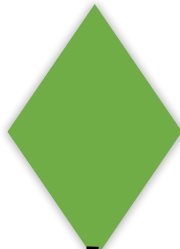
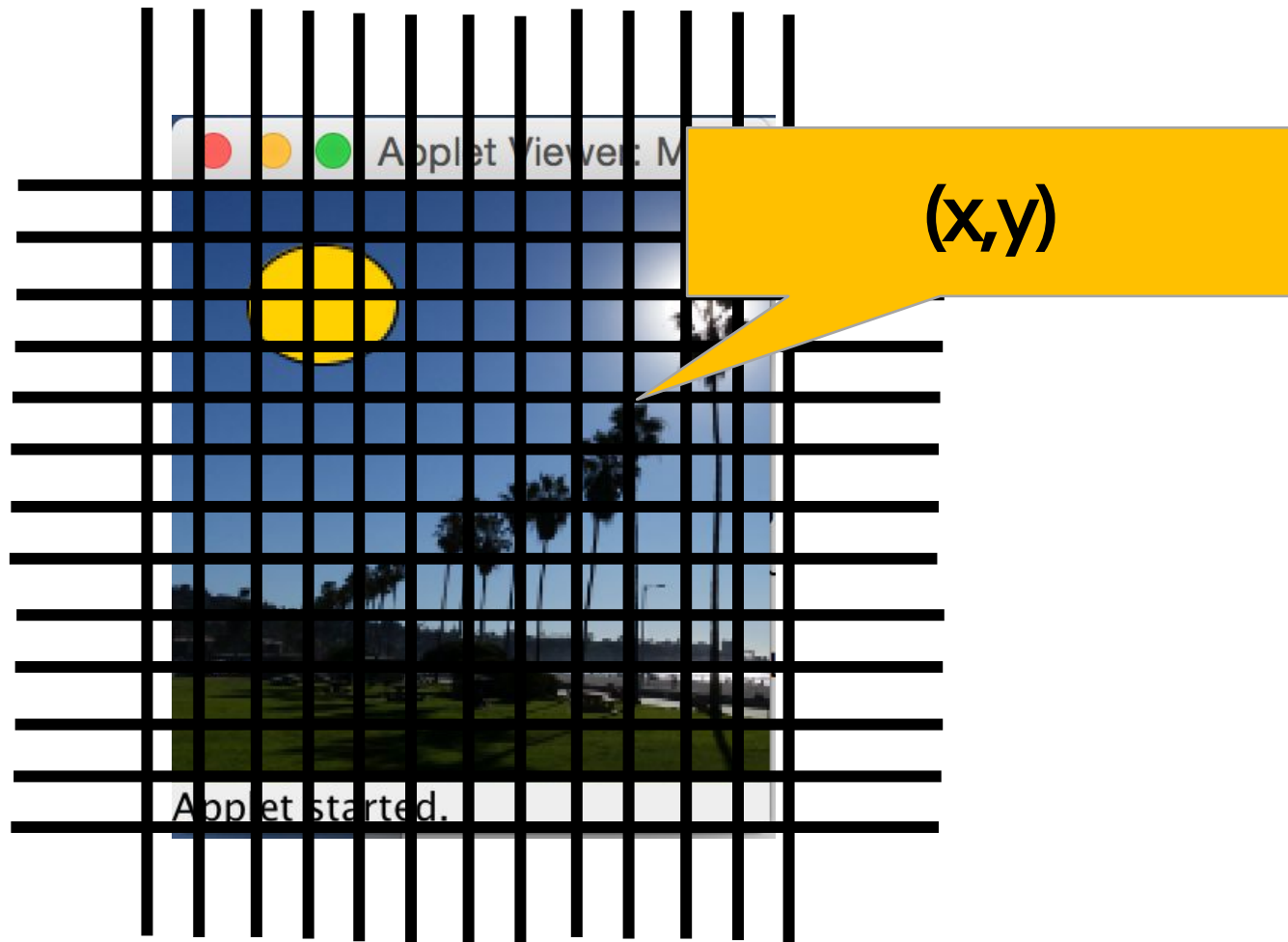
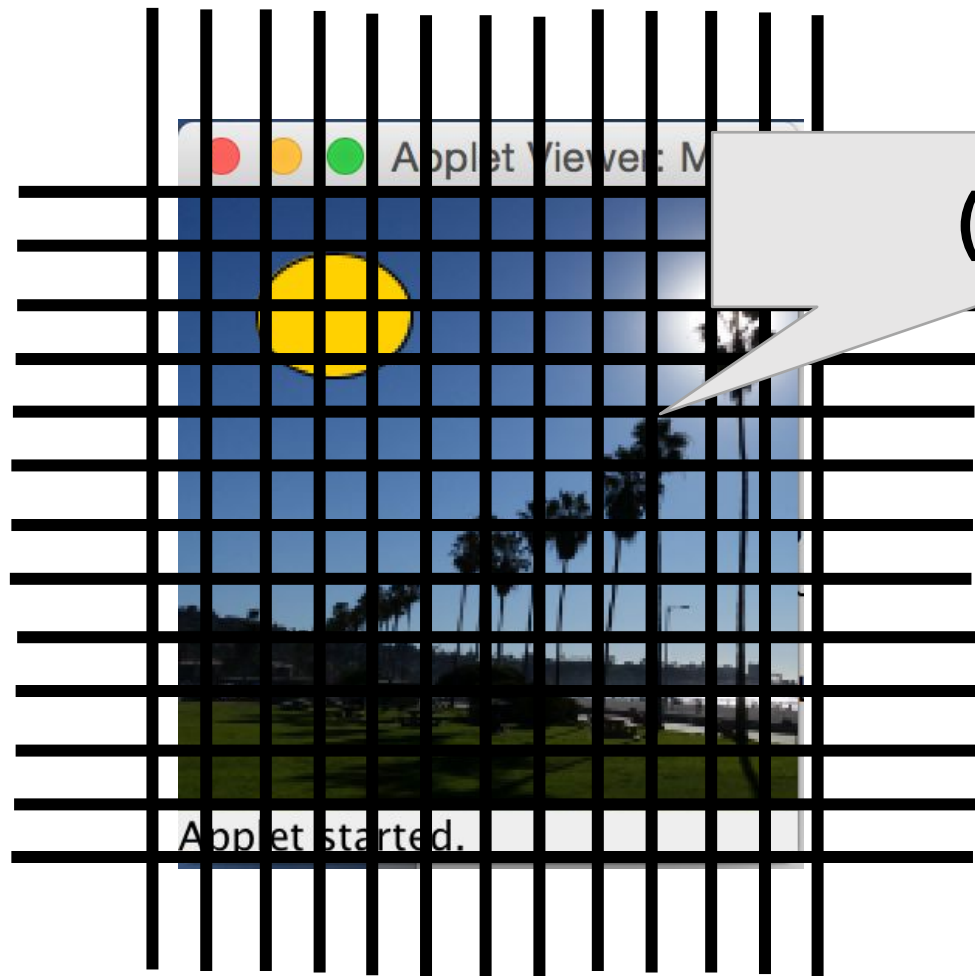


This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/) by Christine Alvarado, Mia Minnes, and Leo Porter, 2015.



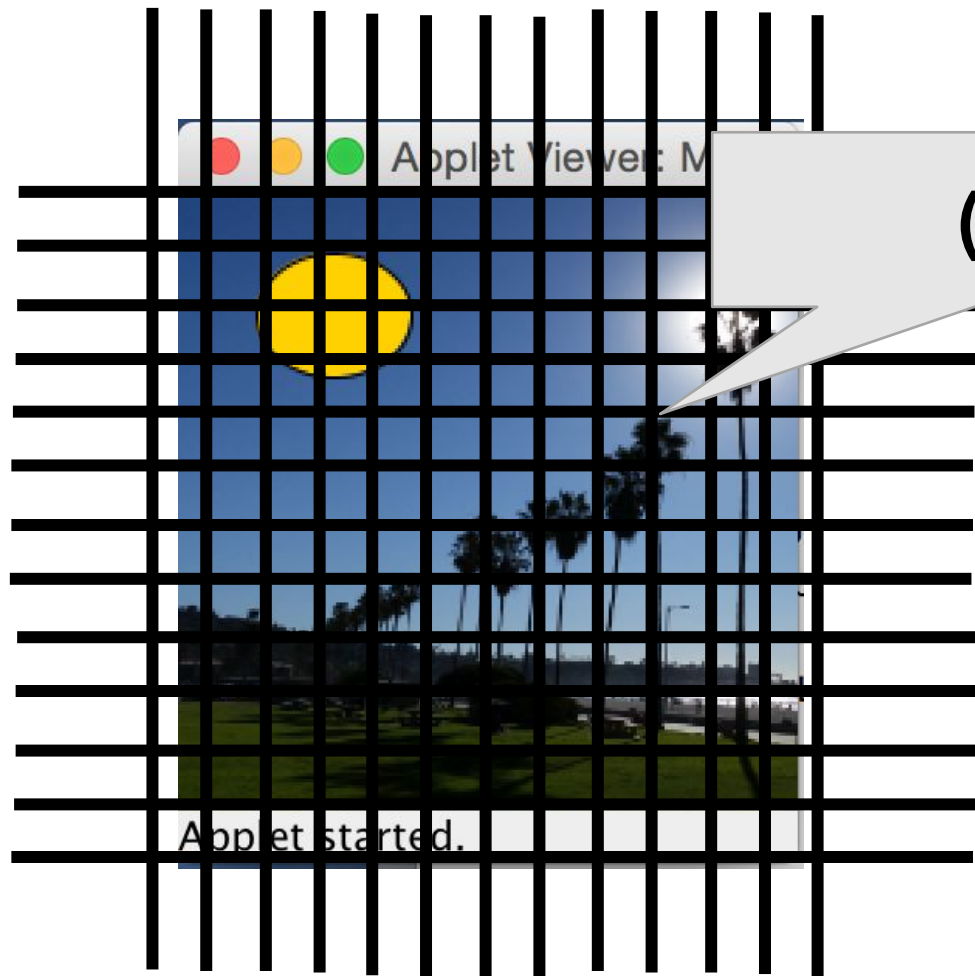




(x,y)

x





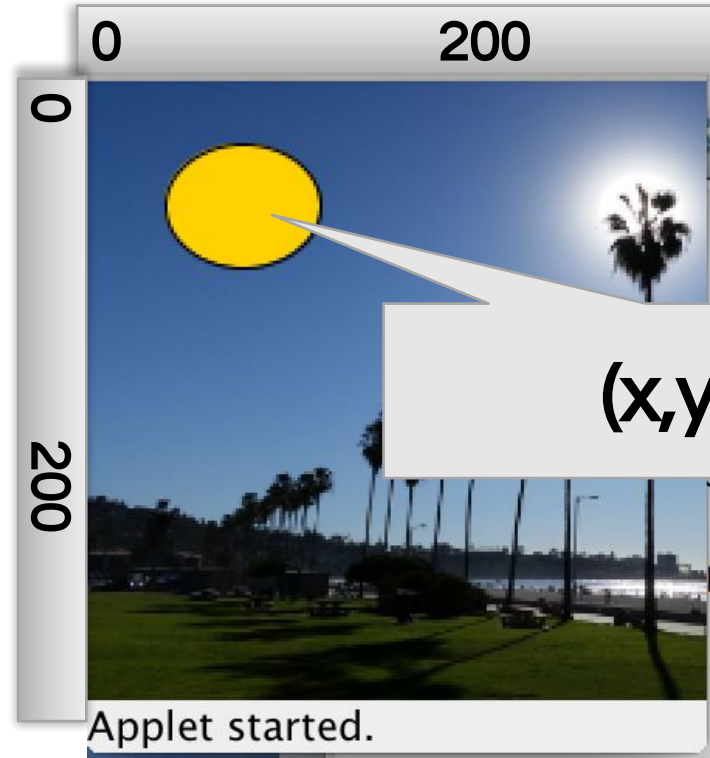
(x,y)

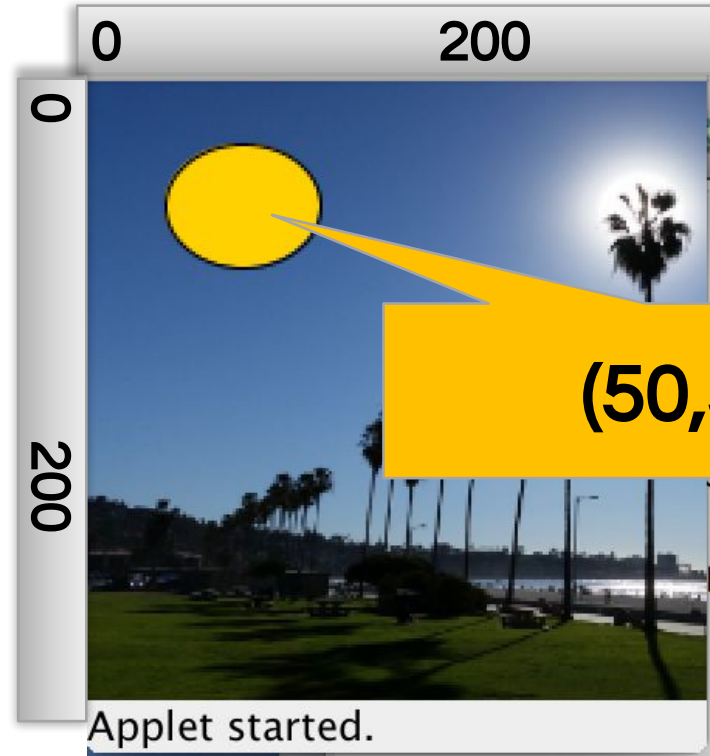
x



y







(50,50)

Now in Processing ...

All Classes

Packages

processing.core
processing.data
processing.event
processing.opengl

All Classes

Event
FloatDict
FloatList
FrameBuffer
IntDict
IntList
JSONArray
JSONObject
KeyEvent
LinePath
LinePath.PathIterator
LineStroker
MouseEvent
PApplet

Class PApplet

```
java.lang.Object
  java.awt.Component
    java.awt.Container
      java.awt.Panel
        java.applet.Applet
          processing.core.PApplet
```

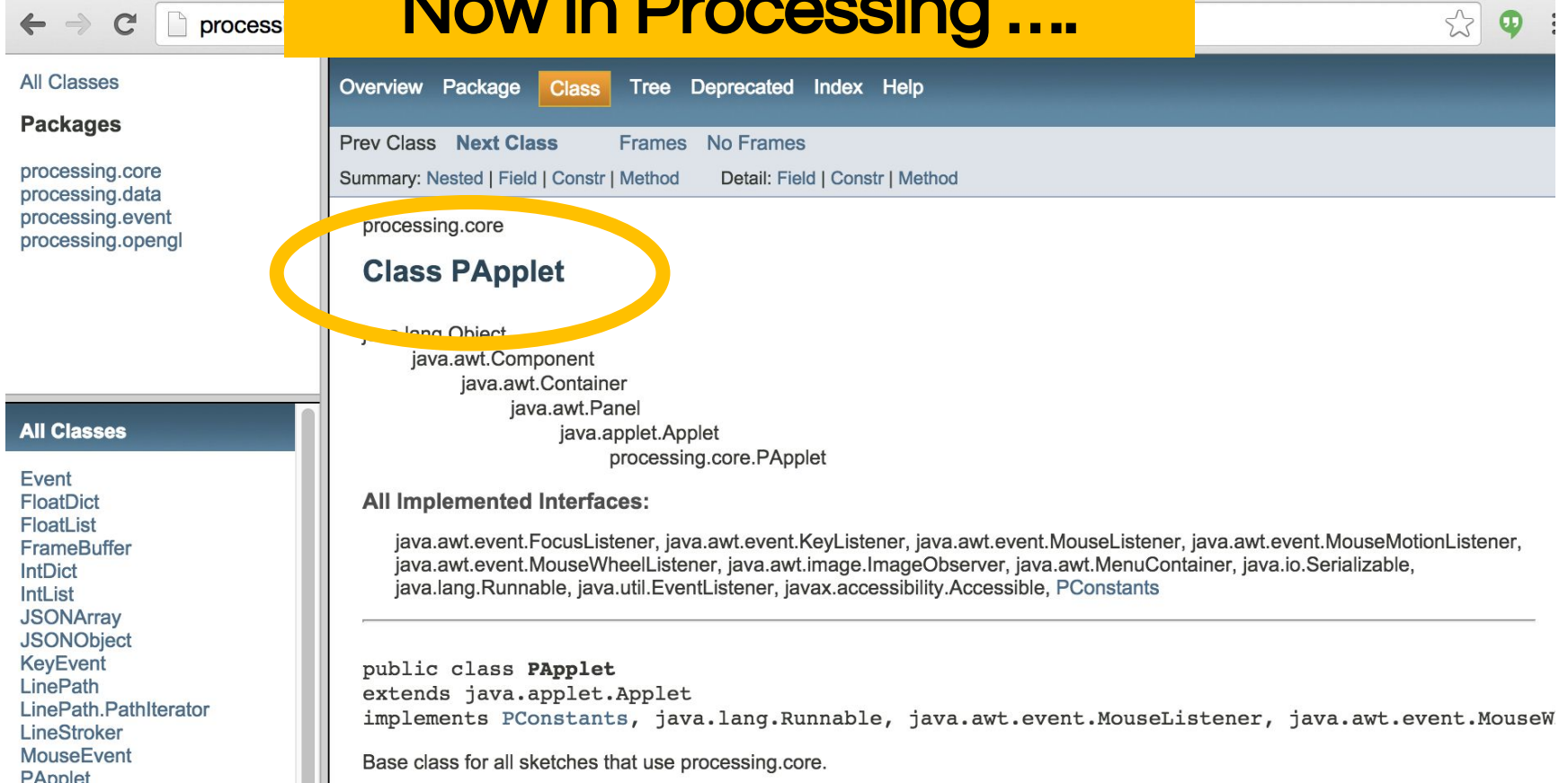
All Implemented Interfaces:

```
java.awt.event.FocusListener, java.awt.event.KeyListener, java.awt.event.MouseListener, java.awt.event.MouseMotionListener,
java.awt.event.MouseWheelListener, java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable,
java.lang.Runnable, java.util.EventListener, javax.accessibility.Accessible, PConstants
```

```
public class PApplet
  extends java.applet.Applet
  implements PConstants, java.lang.Runnable, java.awt.event.MouseListener, java.awt.event.MouseW
```

Base class for all sketches that use processing.core.

Now in Processing ...



The screenshot shows the Processing IDE's class browser. The top navigation bar includes 'Overview', 'Package', 'Class' (selected), 'Tree', 'Deprecated', 'Index', and 'Help'. Below this, there are tabs for 'Prev Class', 'Next Class', 'Frames', and 'No Frames'. The main content area displays the class hierarchy for 'PApplet', which includes 'processing.core', 'java.lang.Object', 'java.awt.Component', 'java.awt.Container', 'java.awt.Panel', 'java.applet.Applet', and 'processing.core.PApplet'. A yellow oval highlights the 'Class PApplet' text. Below the hierarchy, the 'All Implemented Interfaces:' section lists 'java.awt.event.FocusListener', 'java.awt.event.KeyListener', 'java.awt.event.MouseListener', 'java.awt.event.MouseMotionListener', 'java.awt.event.MouseWheelListener', 'java.awt.image.ImageObserver', 'java.awt.MenuContainer', 'java.io.Serializable', 'java.lang.Runnable', 'java.util.EventListener', 'javax.accessibility.Accessible', and 'PConstants'. The bottom section shows the class declaration: 'public class PApplet extends java.applet.Applet implements PConstants, java.lang.Runnable, java.awt.event.MouseListener, java.awt.event.MouseW'. The base class description is 'Base class for all sketches that use processing.core.'

← → ↺ process

All Classes

Packages

- processing.core
- processing.data
- processing.event
- processing.opengl

All Classes

- Event
- FloatDict
- FloatList
- FrameBuffer
- IntDict
- IntList
- JSONArray
- JSONObject
- KeyEvent
- LinePath
- LinePath.PathIterator
- LineStroker
- MouseEvent
- PApplet

Overview Package **Class** Tree Deprecated Index Help

Prev Class **Next Class** Frames No Frames

Summary: Nested | Field | Constr | Method Detail: Field | Constr | Method

processing.core

Class PApplet

- java.lang.Object
- java.awt.Component
- java.awt.Container
- java.awt.Panel
- java.applet.Applet
- processing.core.PApplet

All Implemented Interfaces:

java.awt.event.FocusListener, java.awt.event.KeyListener, java.awt.event.MouseListener, java.awt.event.MouseMotionListener, java.awt.event.MouseWheelListener, java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.lang.Runnable, java.util.EventListener, javax.accessibility.Accessible, [PConstants](#)

```
public class PApplet
extends java.applet.Applet
implements PConstants, java.lang.Runnable, java.awt.event.MouseListener, java.awt.event.MouseW
```

Base class for all sketches that use processing.core.

```
import processing.core.*;
```

```
public class MyPApplet extends PApplet {
```

```
    public void setup() {
```

```
        ...
```

```
    }
```

```
    public void draw() {
```

```
        ...
```

```
    }
```

```
}
```

GUI data type



```
import processing.core.*;
```

Tell Java where to find PApplet

```
public class MyPApplet extends PApplet {
```

```
    public void setup() {
```

```
        ...
```

```
    }
```

```
    public void draw() {
```

```
        ...
```

```
    }
```

```
}
```



```
import processing.core.*;
```

```
public class MyPApplet extends PApplet {
```

```
    public void setup() {
```

```
        ...
```

Configure canvas

```
    }
```

```
    public void draw() {
```

```
        ...
```

Display content

```
    }
```

```
}
```



```
import processing.core.*;
```

```
public class MyPApplet extends PApplet {
```

```
    public void setup() {
```

```
        ...
```

```
    }
```

```
    public void draw() {
```

```
        ...
```

```
    }
```

```
}
```

Executed once

Loops often





```
import processing.core.*;
```

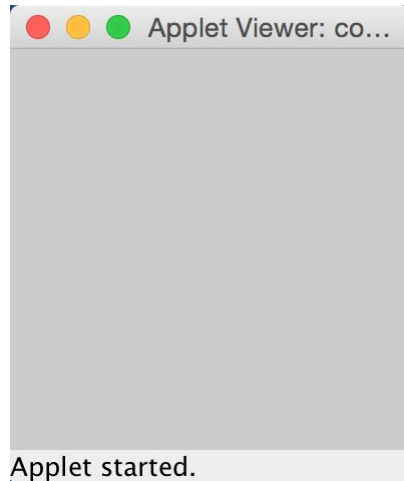
```
public class MyPApplet extends PApplet {
```

```
}
```

```
import processing.core.*;
```

```
public class MyPApplet extends PApplet {
```

```
}
```



```
import processing.core.*;
```

```
public class MyPApplet extends PApplet {
```

```
    public void setup() {
```

```
        ...
```

```
    }
```

```
    public void draw() {
```

```
        ...
```

```
    }
```

```
}
```



Processing

[Cover](#)[Download](#)[Exhibition](#)[Reference](#)[Libraries](#)[Tools](#)[Environment](#)[Tutorials](#)[Examples](#)[Books](#)[Handbook](#)

Reference. The Processing Language was designed to facilitate the creation of sophisticated visual structures.

Structure

`()` (parentheses)
`,` (comma)
`.` (dot)
`/* */` (multiline comment)
`/** */` (doc comment)
`//` (comment)
`;` (semicolon)
`=` (assign)
`[]` (array access)

Shape

`createShape()`
`loadShape()`
`PShape`

2D Primitives


`arc()`
`ellipse()`
`line()`
`point()`

Color

Setting
`background()`
`clear()`
`colorMode()`
`fill()`
`noFill()`
`noStroke()`
`stroke()`

```
import processing.core.*;

public class MyPApplet extends PApplet
{
    private String URL = "http://...jpg";
    private PImage backgroundImg;

    public void setup()
    {
        
    }

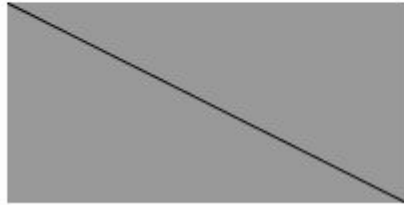
    public void draw()
    {
        // Body not shown
    }
}
```

```
import processing.core.*;
```

Name

size()

Examples



```
size(200, 100);  
background(153);  
line(0, 0, width, height);
```

```
size(200,200) ;
```

Description

Defines the dimension of the display window in units of pixels. The `size()` function must be the first line of code, or the first code inside `setup()`. Any code that appears before the `size()` command may run more than once, which can lead to confusing results.

```
{  
    // Body not shown  
}
```

```
}
```



```
PImage webImg;  
  
void setup() {  
  String url = "https://processing.org/img/processing-web.png";  
  // Load image from a web server  
  webImg = loadImage(url, "png");  
}  
  
void draw() {  
  background(0);  
  image(webImg, 0, 0);  
}
```

Description Loads an image into a variable of type PImage. Four types of images (.gif, .jpg, .tga, .png) images may be loaded. To load correctly, images must be located in the data directory of the current sketch.

```
backgroundImg = loadImage(URL, "jpg") ;  
}
```

```
public void draw()  
{  
  // Body not shown  
}  
}
```

```
import processing.core.*;
```

```
public class MyPApplet extends PApplet  
{
```

```
    private String URL = "http://...jpg";  
    private PImage backgroundImage;
```

```
    public void setup()  
    {
```

```
        size(200,200);  
        backgroundImage = loadImage(URL,"jpg");  
    }
```

```
    public void draw()  
    {
```

```
        // Body not shown
```

```
    }
```

```
}
```

