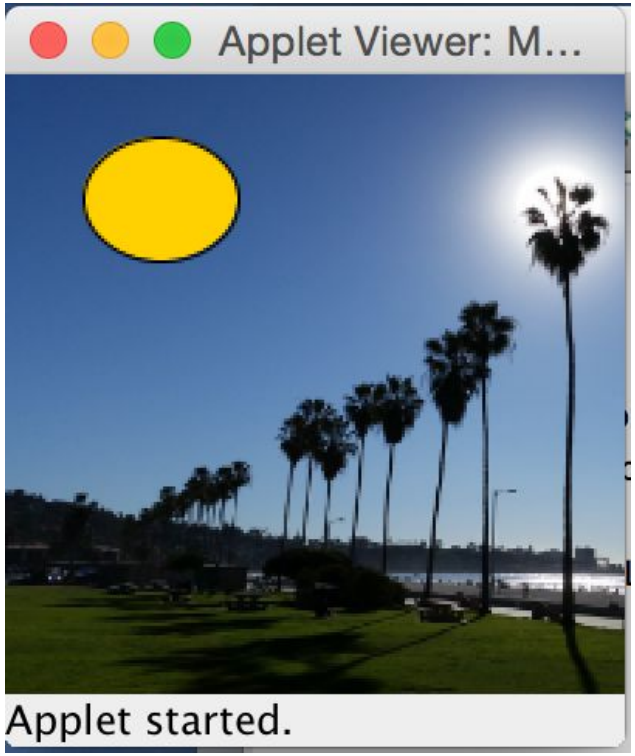
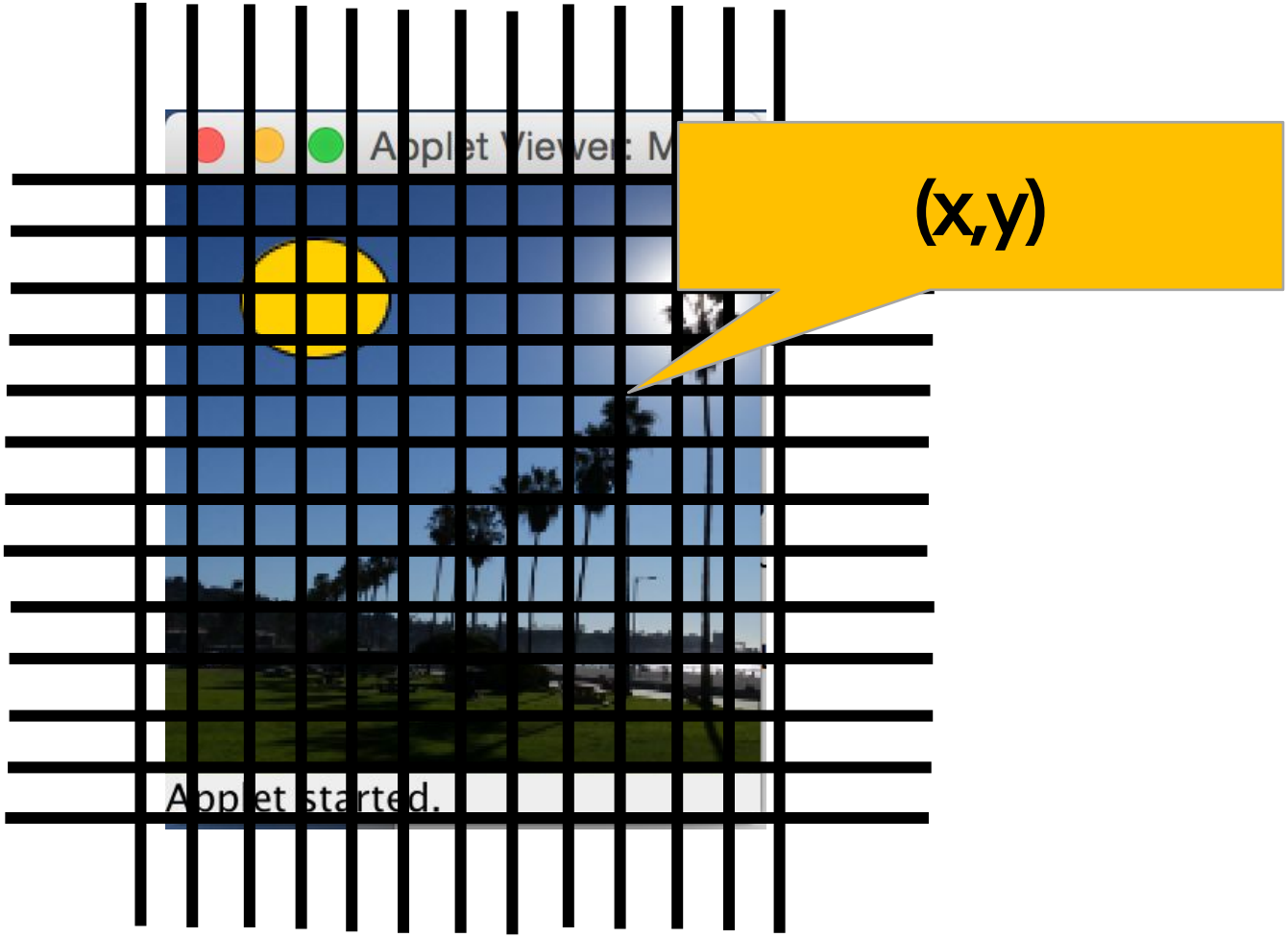
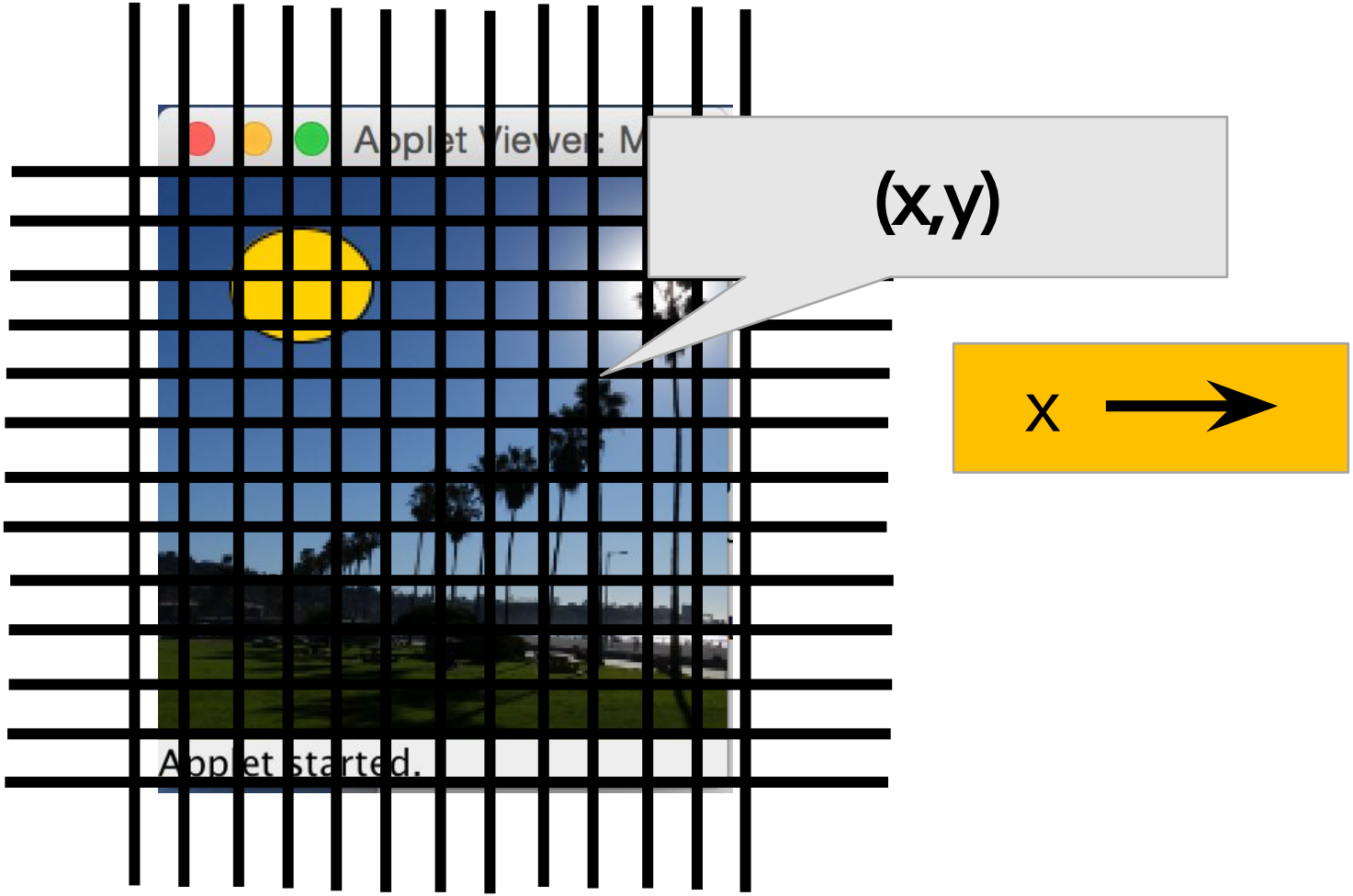
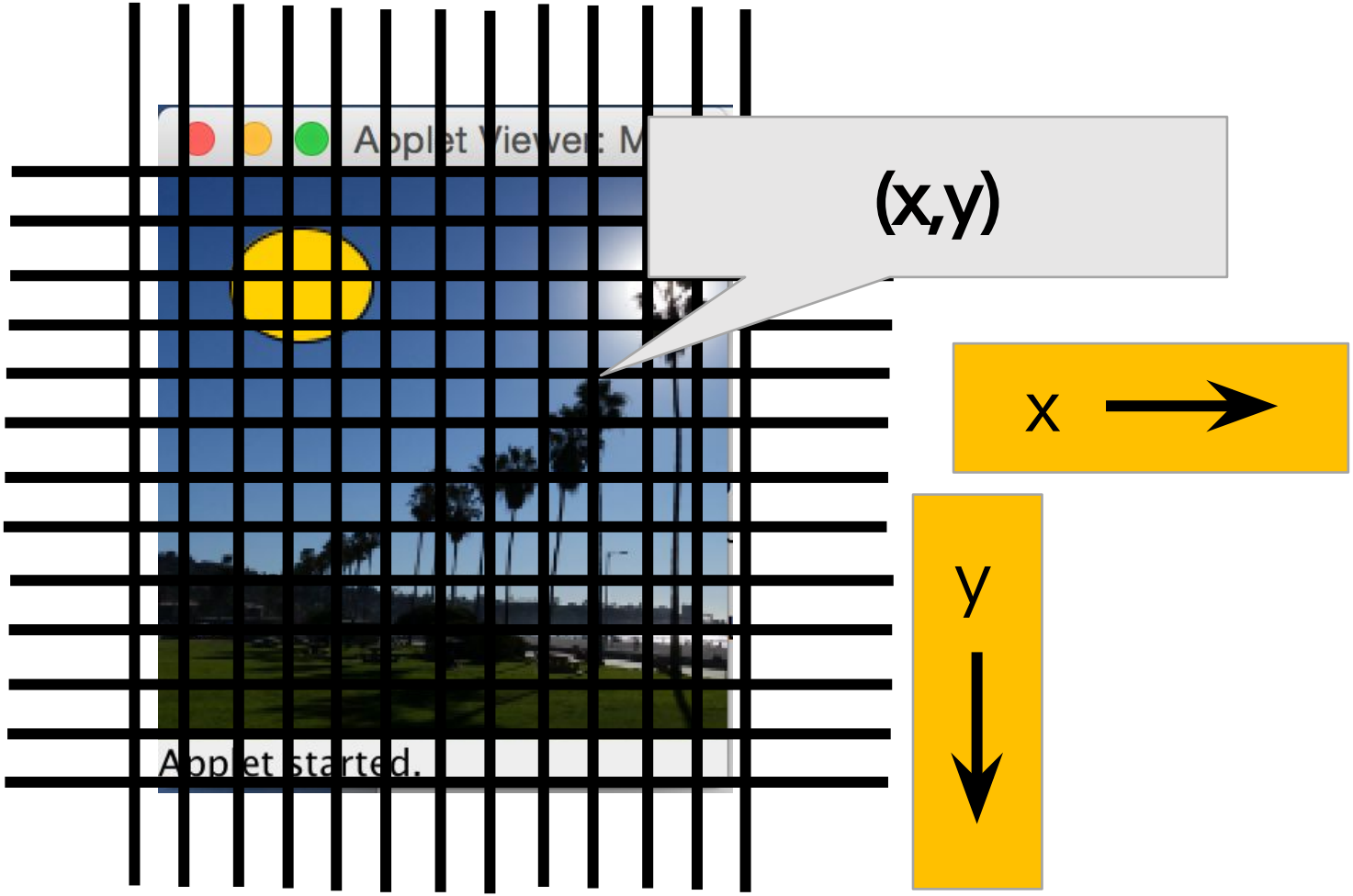
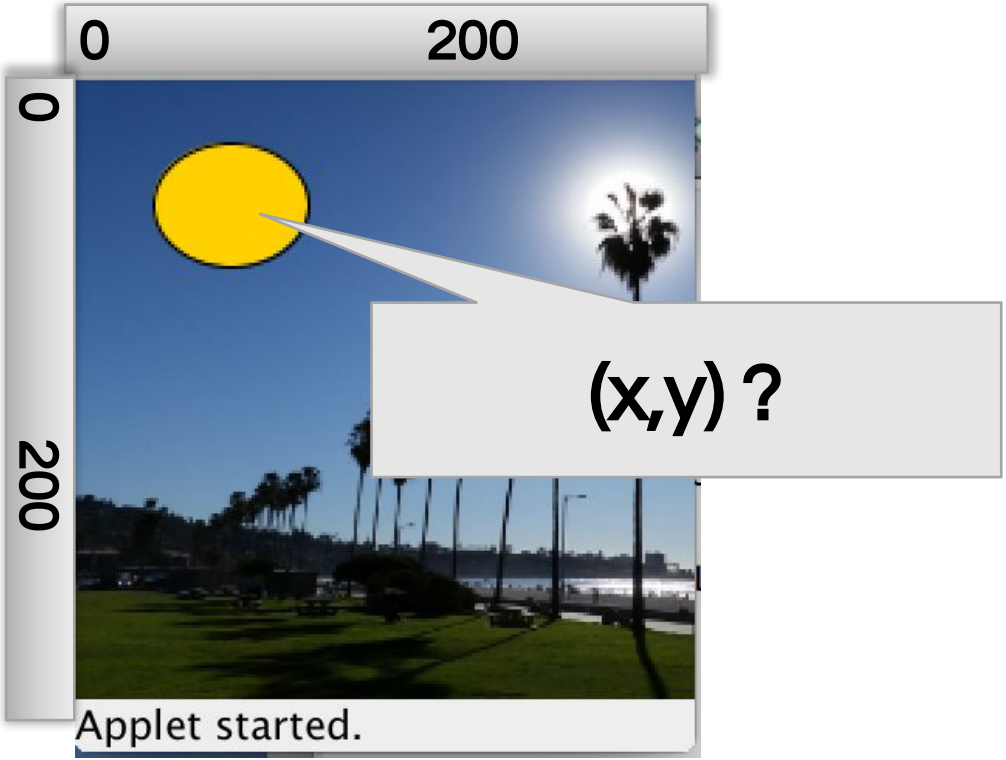
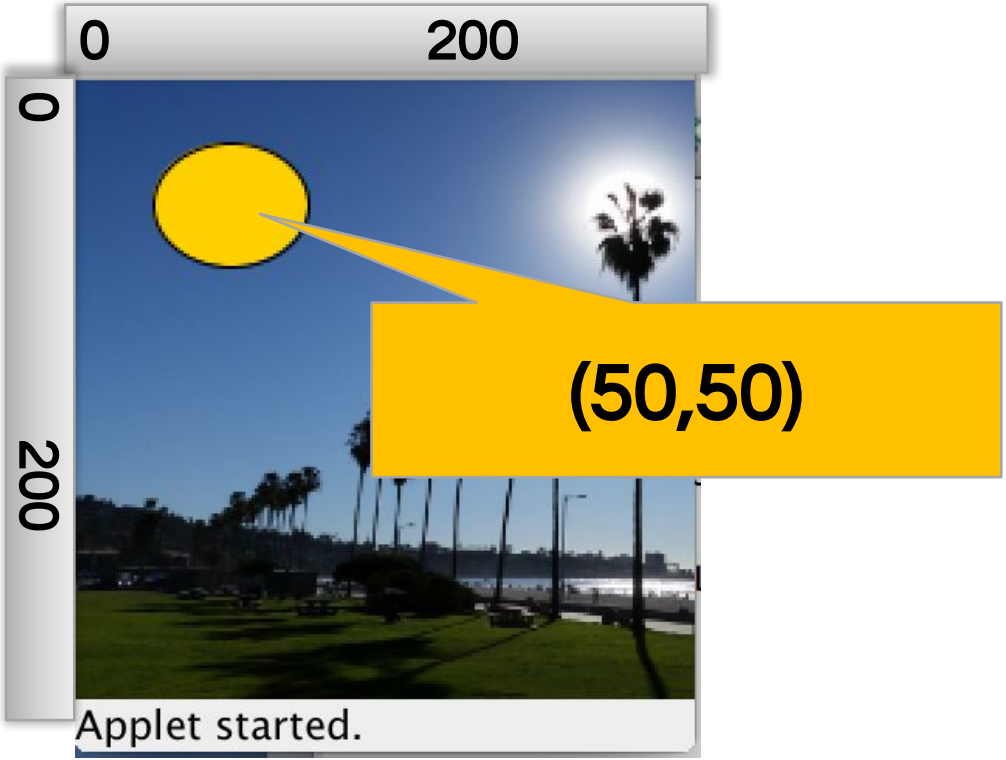



Applet Viewer: M
(x,y)
Applet started.

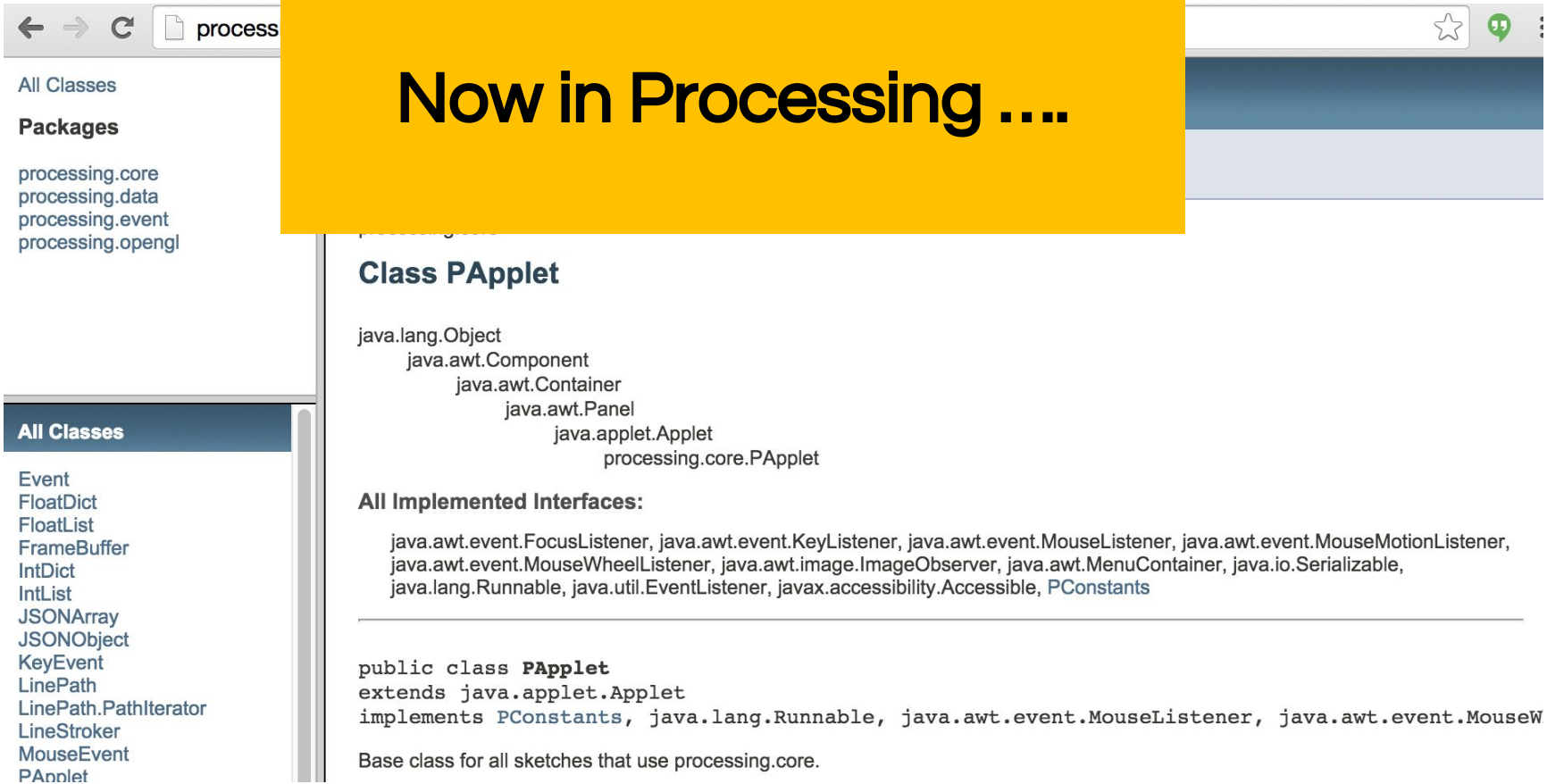

Applet Viewer: M
(x,y)
x
Applet started.


Applet Viewer: M
Applet started.
(x,y)
x
y

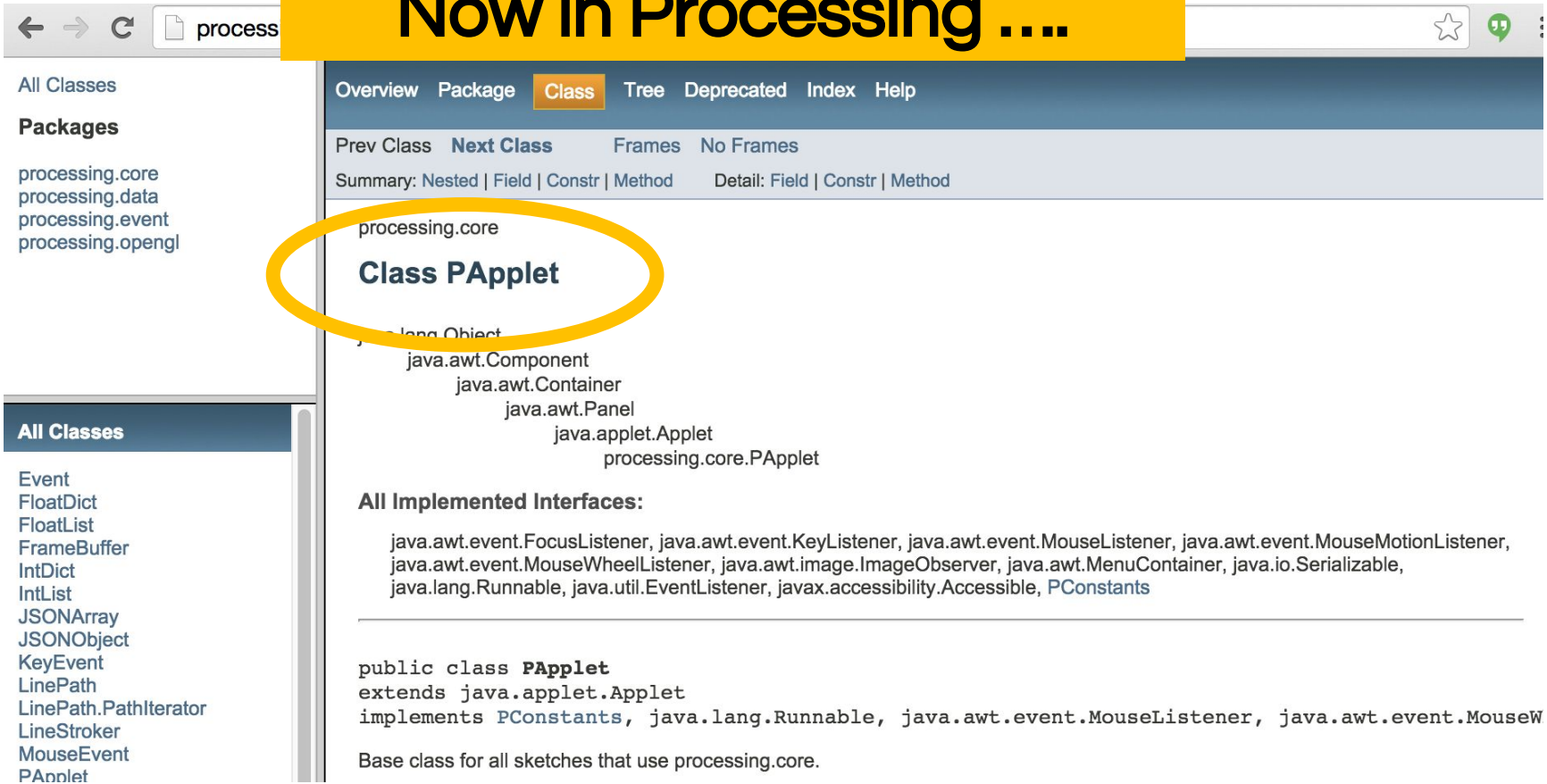

0
200
0
200
(x,y) ?
Applet started.


0
200
0
200
(50,50)
Applet started.

Now in Processing ….



Now in Processing ....



```
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() {
      ...
   }
   public void draw() {
      ...
   }
}
```

Configure canvas

Display content

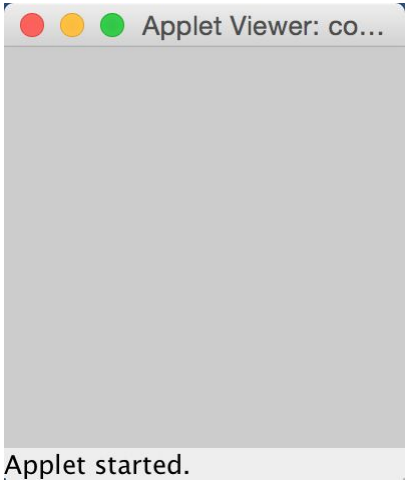
```
import processing.core.*;

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

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

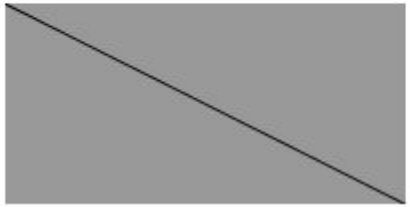
Setting
background()
clear()
colorMode()
fill()
noFill()
noStroke()
stroke()

```
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 backgroundImg;

    public void setup()
    {
        size(200,200);
     backgroundImg = loadImage(URL,"jpg");
     }

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