



# Graphics Programming

---

## Swing

Written by Paul Pu All Rights  
Reserved  
[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Play with Demo

---

- `D:\jdk1.3\demo\jfc\SwingSet2>appletviewer  
Swingset2.html`



# Graphics Programming

---

- AWT:

  - Abstract Window Toolkit

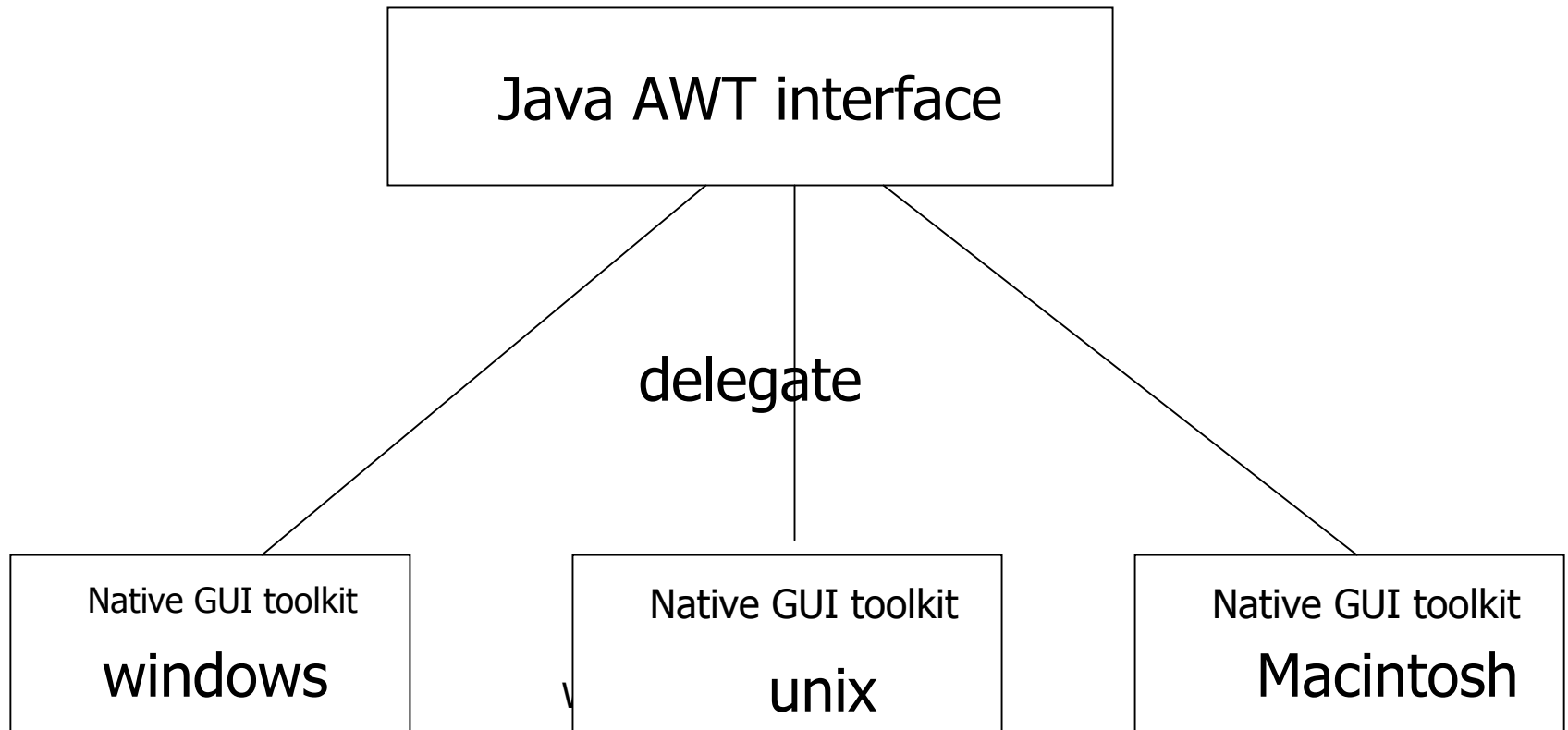
- How does AWT work?

The way the basic AWT library deals with user interface elements is to delegate their creation and behavior to the native GUI toolkit on each target platform.



# Graphics Programming

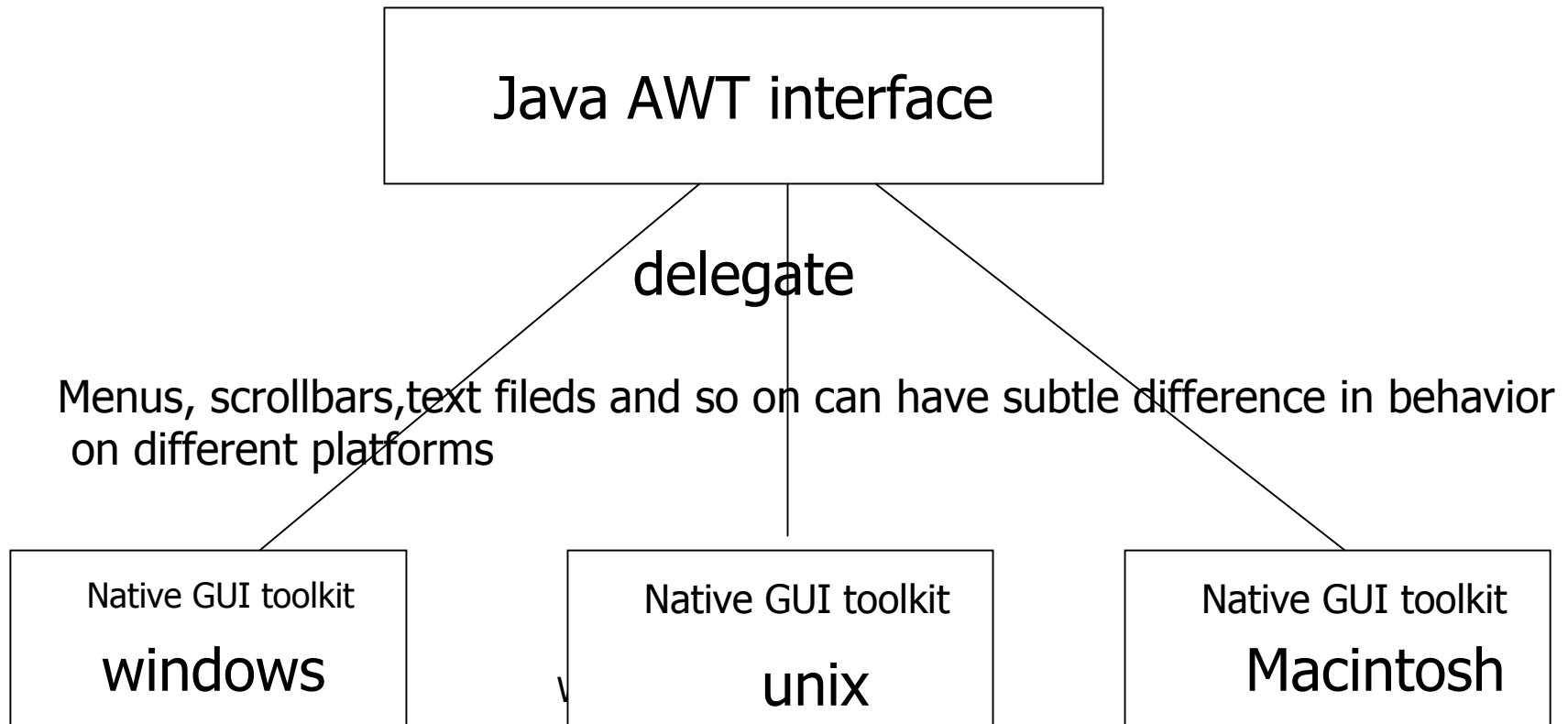
---





# Graphics Programming

---

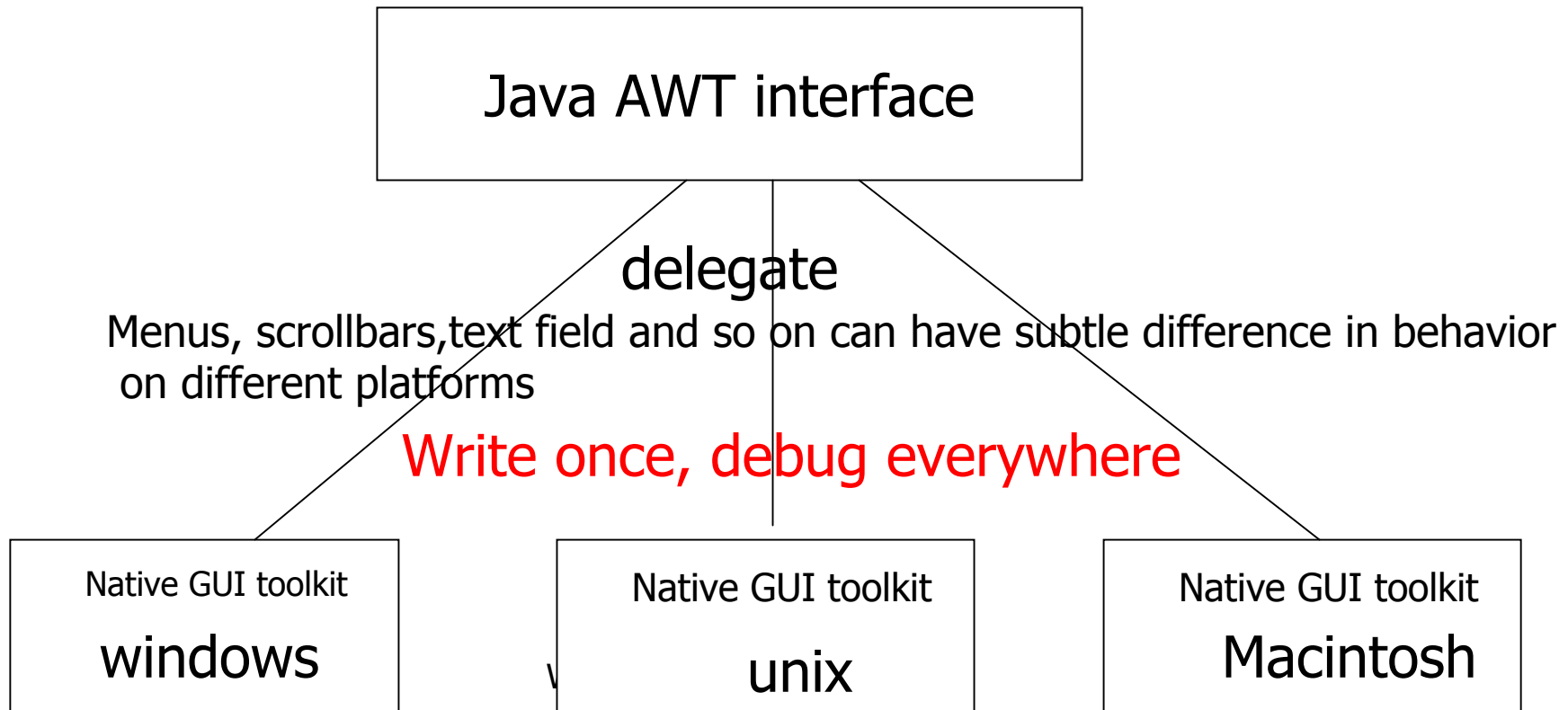


Reserved



# Graphics Programming

---



Reserved



# Graphics Programming

---

- **IFC(Internet Foundation Classes):** In 1996, Netscape created a GUI library IFC. With IFC, user interface elements were painted onto blank windows.
- **Netscape's IFC widgets** looked and behaved the same on matter which platform the program ran on.
- **Sun worked with Netscape** to perfect this approach, creating a user interface library with the code name "**Swing**"



# Graphics Programming

---

- Swing has a much richer and more convenient set of user interface elements
- Swing depends far less on the underlying platform
- Swing will give a consistent user experience across platforms
- Swing “Write Once, Run Anywhere”
- Swing is more robust, has more features, is more portable, and is easier to use than AWT
- Swing is slower than AWT
- By default, the browser does not support Swing.



# Graphics Programming

---

- Notes:

When Swing came to the market, the Swing package started out as `com.sun.java.swing`



# Graphics Programming

## Creating a Frame

---

- Frame: A top-level window- that is, a window that is not contained inside another window.
- `import javax.swing.*;`
- `public class SimpleFrameTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `SimpleFrame frame = new SimpleFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`

Written by Paul Pu All Rights  
Reserved  
[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Creating a Frame

---

```
■ class SimpleFrame extends JFrame
■ {
■     public static final int WIDTH = 300;
■     public static final int HEIGHT = 200;
■
■     public SimpleFrame()
■     {
■         setSize(WIDTH, HEIGHT);
■     }
■ }
```



# Graphics Programming

## Creating a Frame

---

- We use the following statement to exit the program

```
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

- In other programs with multiple frames, you would not want the program to exit just because the user closes one of the frames. By default, a frame is hidden when the user closes it, but the program does not terminate

Written by Paul Pu All Rights  
Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Creating a Frame

---

- You can replace the call to the `setDefaultCloseOperation` method with the following code:

```
frame.addWindowListener( new WindowAdapter()  
    {  
        public void windowClosing(WindowEvent e) {  
            System.exit(0);  
        }  
    });
```



# Graphics Programming

## Frame Positioning

---

- `import java.awt.*;`
- `import java.awt.event.*;`
- `import javax.swing.*;`
  
- `public class CenteredFrameTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `CenteredFrame frame = new CenteredFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming

## Frame Positioning

---

- class CenteredFrame extends JFrame
- {
- public CenteredFrame()
- {
- // get screen dimensions
  
- Toolkit kit = Toolkit.getDefaultToolkit();
- Dimension screenSize = kit.getScreenSize();
- int screenHeight = screenSize.height;
- int screenWidth = screenSize.width;
  
- // center frame in screen

Written by Paul Pu All Rights  
Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Frame Positioning

---

- `setSize(screenWidth / 2, screenHeight / 2);`
- `setLocation(screenWidth / 4, screenHeight / 4);`
  
- `// set frame icon and title`
  
- `Image img = kit.getImage("icon.gif");`
- `setIconImage(img);`
- `setTitle("CenteredFrame");`
- `}`
- `}`



# Graphics Programming

## Displaying Information in a Panel

---

- `import javax.swing.*;`
- `import java.awt.*;`
  
- `public class NotHelloWorld`
- `{`
- `public static void main(String[] args)`
- `{`
- `NotHelloWorldFrame frame = new NotHelloWorldFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming

## Displaying Information in a Panel

---

- class NotHelloWorldFrame extends JFrame
- {
- public static final int WIDTH = 300;
- public static final int HEIGHT = 200;
- 
- public NotHelloWorldFrame()
- {
- setTitle("NotHelloWorld");
- setSize(WIDTH, HEIGHT);
- 
- // add panel to frame
- 
- NotHelloWorldPanel panel = new NotHelloWorldPanel();
- Container contentPane = getContentPane();
- contentPane.add(panel);
- }
- }

Written by Paul Pu All Rights Reserved  
www.torontocollege.com



# Graphics Programming

## Displaying Information in a Panel

---

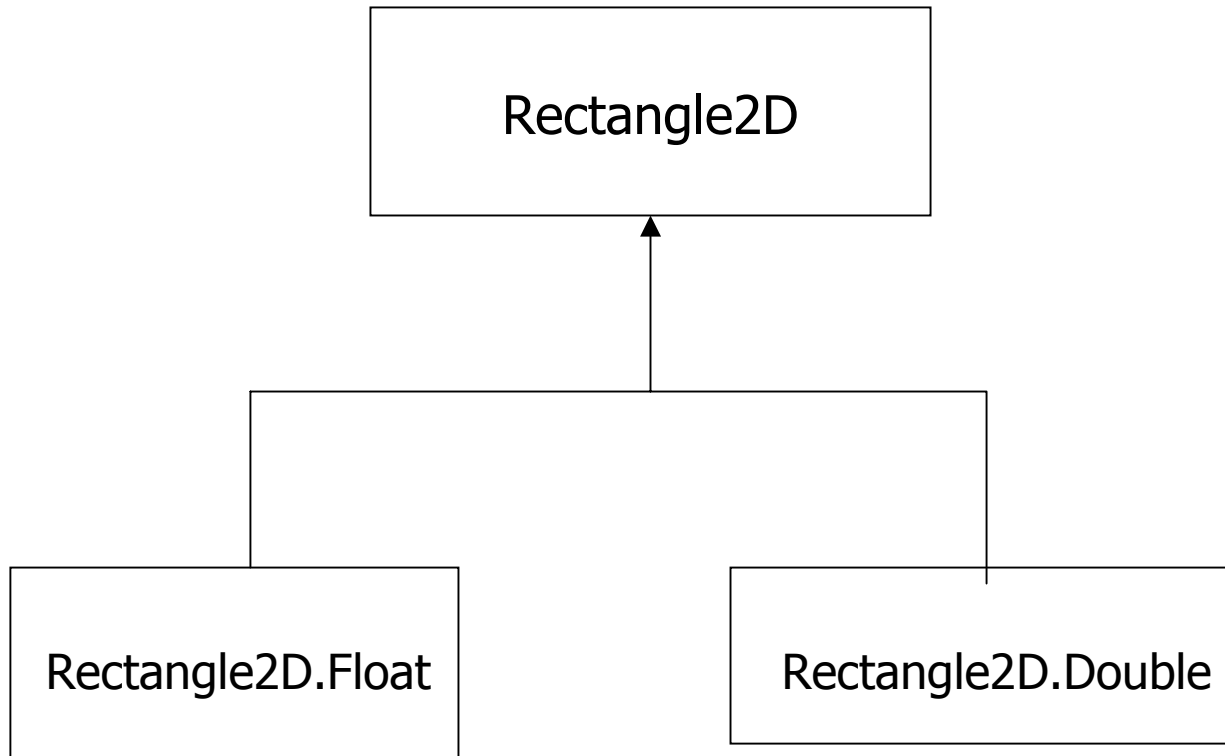
```
■ /**  
■   A panel that displays a message.  
■ */  
■ class NotHelloWorldPanel extends JPanel  
■ {  
■   public static final int MESSAGE_X = 75;  
■   public static final int MESSAGE_Y = 100;  
■  
■   public void paintComponent(Graphics g)  
■   {  
■     super.paintComponent(g);  
■  
■     g.drawString("Not a Hello, World program",  
■       MESSAGE_X, MESSAGE_Y);  
■   }  
■ }
```

Written by Paul Pu All Rights  
Reserved  
[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming 2D Shapes

---





# Graphics Programming

## 2D Shapes

---

- `import java.awt.*;`
- `import java.awt.geom.*;`
- `import javax.swing.*;`
  
- `public class DrawTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `DrawFrame frame = new DrawFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming

## 2D Shapes

---

- `/**`
- `A frame that contains a panel with drawings`
- `*/`
- `class DrawFrame extends JFrame`
- `{`
- `public DrawFrame()`
- `{`
- `setTitle("DrawTest");`
- `setSize(WIDTH, HEIGHT);`
  
- `// add panel to frame`
  
- `DrawPanel panel = new DrawPanel();`
- `Container contentPane = getContentPane();`
- `contentPane.add(panel);`
- `}`

Written by Paul Pu All Rights Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming 2D Shapes

---

- `public static final int WIDTH = 400;`
- `public static final int HEIGHT = 400;`
- `}`



# Graphics Programming

## 2D Shapes

---

- `/**`
- `A panel that displays rectangles and ellipses.`
- `*/`
- `class DrawPanel extends JPanel`
- `{`
- `public void paintComponent(Graphics g)`
- `{`
- `super.paintComponent(g);`
- `Graphics2D g2 = (Graphics2D)g;`
  
- `// draw a rectangle`
  
- `double leftX = 100;`
- `double topY = 100;`
- `double width = 200;`
- `double height = 150;`

Written by Paul Pu All Rights Reserved



# Graphics Programming

## 2D Shapes

---

- `Rectangle2D rect = new Rectangle2D.Double(leftX, topY,`
- `width, height);`
- `g2.draw(rect);`
  
- `// draw the enclosed ellipse`
  
- `Ellipse2D ellipse = new Ellipse2D.Double();`
- `ellipse setFrame(rect);`
- `g2.draw(ellipse);`
  
- `// draw a diagonal line`
  
- `g2.draw(new Line2D.Double(leftX, topY,`
- `leftX + width, topY + height));`



# Graphics Programming

## 2D Shapes

---

- `// draw a circle with the same center`
- `double centerX = rect.getCenterX();`
- `double centerY = rect.getCenterY();`
- `double radius = 150;`
- `Ellipse2D circle = new Ellipse2D.Double();`
- `circle.setFrameFromCenter(centerX, centerY,`
- `centerX + radius, centerY + radius);`
- `g2.draw(circle);`
- `}`
- `}`



# Graphics Programming Colors

---

- The `java.awt.Color` class offers predefines constants for the 13 standard colors

black	green	red
blue	lightGray	white
cyan	magenta	yellow
darkGray	orange	
gray	pink	

- ✍ You can specify a custom color by creating a `Color` object by its red, green, and blue components. Using a scale of 0-255 for the redness, blueness, and greenness, call the `Color` like this:

```
Color(int redness, int greenness, int blueness)
```



# Graphics Programming Colors

---

- `import java.awt.*;`
- `import java.awt.geom.*;`
- `import javax.swing.*;`
  
- `public class FillTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `FillFrame frame = new FillFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming Colors

---

- `/**`
- `A frame that contains a panel with drawings`
- `*/`
- `class FillFrame extends JFrame`
- `{`
- `public FillFrame()`
- `{`
- `setTitle("FillTest");`
- `setSize(WIDTH, HEIGHT);`
  
- `// add panel to frame`
  
- `FillPanel panel = new FillPanel();`
- `Container contentPane = getContentPane();`
- `contentPane.add(panel);`
- `}`



# Graphics Programming Colors

---

- `public static final int WIDTH = 400;`
- `public static final int HEIGHT = 400;`
- `}`
  
- `/**`
- `A panel that displays filled rectangles and ellipses`
- `*/`
- `class FillPanel extends JPanel`
- `{`
- `public void paintComponent(Graphics g)`
- `{`
- `super.paintComponent(g);`
- `Graphics2D g2 = (Graphics2D)g;`



# Graphics Programming Colors

---

- `// draw a rectangle`
- `double leftX = 100;`
- `double topY = 100;`
- `double width = 200;`
- `double height = 150;`
- `Rectangle2D rect = new Rectangle2D.Double(leftX, topY,`
- `width, height);`
- `g2.setPaint(Color.red);`
- `g2.fill(rect);`



# Graphics Programming Colors

---

- `// draw the enclosed ellipse`
- `Ellipse2D ellipse = new Ellipse2D.Double();`
- `ellipse setFrame(rect);`
- `g2.setPaint(new Color(0, 128, 128)); // a dull blue-green`
- `g2.fill(ellipse);`
- `}`
- `}`



# Graphics Programming

## Text and Fonts

---

- To find out which fonts are available on a particular computer, call the `getAvailableFontFamilyNames` method of the `GraphicsEnvironment` class.
- ```
import java.awt.*;  
public class ListFonts  
{  
    public static void main(String[] args)  
    {  
        String[] fontNames = GraphicsEnvironment  
            .getLocalGraphicsEnvironment()  
            .getAvailableFontFamilyNames();  
  
        for (int i = 0; i < fontNames.length; i++)  
            System.out.println(fontNames[i]);  
    }  
}
```

Written by Paul Pu All Rights  
Reserved  
[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Text and Fonts

---

- `import java.awt.*;`
- `import java.awt.font.*;`
- `import java.awt.geom.*;`
- `import javax.swing.*;`
  
- `public class FontTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `FontFrame frame = new FontFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming

## Text and Fonts

---

- `/**`
- `A frame with a text message panel`
- `*/`
- `class FontFrame extends JFrame`
- `{`
- `public FontFrame()`
- `{`
- `setTitle("FontTest");`
- `setSize(WIDTH, HEIGHT);`
  
- `// add panel to frame`
  
- `FontPanel panel = new FontPanel();`
- `Container contentPane = getContentPane();`
- `contentPane.add(panel);`
- `}`



# Graphics Programming

## Text and Fonts

---

- `public static final int WIDTH = 300;`
- `public static final int HEIGHT = 200;`
- `}`
- `/**`
- A panel that shows a centered message in a box.
- `*/`
- `class FontPanel extends JPanel`
- `{`
- `public void paintComponent(Graphics g)`
- `{`
- `super.paintComponent(g);`
- `Graphics2D g2 = (Graphics2D)g;`
  
- `String message = "Hello, World!";`
  
- `Font f = new Font("Serif", Font.BOLD, 36);`
- `g2.setFont(f);`

Written by Paul Pu All Rights Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming

## Text and Fonts

---

- `// measure the size of the message`
- `FontRenderContext context = g2.getFontRenderContext();`
- `Rectangle2D bounds = f.getStringBounds(message, context);`
- `// set (x,y) = top left corner of text`
- `double x = (getWidth() - bounds.getWidth()) / 2;`
- `double y = (getHeight() - bounds.getHeight()) / 2;`
- `// add ascent to y to reach the baseline`
- `double ascent = -bounds.getY();`
- `double baseY = y + ascent;`



# Graphics Programming

## Text and Fonts

---

- `// draw the message`
- `g2.drawString(message, (int)x, (int)(baseY));`
- `g2.setPaint(Color.gray);`
- `// draw the baseline`
- `g2.draw(new Line2D.Double(x, baseY,`
- `x + bounds.getWidth(), baseY));`
- `// draw the enclosing rectangle`
- `Rectangle2D rect = new Rectangle2D.Double(x, y,`
- `bounds.getWidth(),`
- `bounds.getHeight());`
- `g2.draw(rect);`
- `}`
- `}`

Written by Paul Pu All Rights  
Reserved  
[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming Images

---

- To get a Toolkit object, use the static getDefaultToolkit method of the Toolkit class.

```
String name="blue.gif";
```

```
Image image=Toolkit.getDefaultToolkit().getImage(name);
```

- ✍ To get an image file from the Net, you must supply the URL

```
URL u=new URL(http://www.someplace.com/amImage.gif);
```

```
Image image=Toolkit.getDefaultToolkit().getImage(u);
```

- ✍ You can display image with the drawImage method of the Graphics class

```
public void paintComponent(Graphics g) {
```

```
...
```

```
g.drawImage(image, x, y, null)
```

Written by Paul Pu All Rights  
Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming Images

---

- `import java.awt.*;`
- `import java.awt.event.*;`
- `import javax.swing.*;`
  
- `public class ImageTest`
- `{`
- `public static void main(String[] args)`
- `{`
- `ImageFrame frame = new ImageFrame();`
- `frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- `frame.show();`
- `}`
- `}`



# Graphics Programming Images

---

- `/**`
- `A frame with an image panel`
- `*/`
- `class ImageFrame extends JFrame`
- `{`
- `public ImageFrame()`
- `{`
- `setTitle("ImageTest");`
- `setSize(WIDTH, HEIGHT);`
  
- `// add panel to frame`
  
- `ImagePanel panel = new ImagePanel();`
- `Container contentPane = getContentPane();`
- `contentPane.add(panel);`
- `}`



# Graphics Programming Images

---

- `public static final int WIDTH = 300;`
- `public static final int HEIGHT = 200;`
- `}`
- `/**`
- `A panel that displays a tiled image`
- `*/`
- `class ImagePanel extends JPanel`
- `{`
- `public ImagePanel()`
- `{`
- `// acquire the image`
- `image = Toolkit.getDefaultToolkit().getImage`
- `("blue-ball.gif");`
- `MediaTracker tracker = new MediaTracker(this);`
- `tracker.addImage(image, 0);`
- `try { tracker.waitForID(0); }`
- `catch (InterruptedException e) {`
- `}`
- `}`

Written by Paul Py All Rights Reserved

[www.torontocollege.com](http://www.torontocollege.com)



# Graphics Programming Images

---

```
public void paintComponent(Graphics g)
{
    super.paintComponent(g);

    int imageWidth = image.getWidth(this);
    int imageHeight = image.getHeight(this);

    // draw the image in the upper-left corner

    g.drawImage(image, 0, 0, null);

    // tile the image across the panel

    for (int i = 0; i * imageWidth <= getWidth(); i++)
        for (int j = 0; j * imageHeight <= getHeight(); j++)
            if (i + j > 0)
                g.copyArea(0, 0, imageWidth, imageHeight,
                    i * imageWidth, j * imageHeight);
}
```

Written by Paul Pu. All Rights Reserved



# Graphics Programming Images

---

- 
- private Image image;
- }