AWT IMAGE CLASS

Introduction

Image control is superclass for all image classes representing graphical images.

Class declaration

Following is the declaration for **java.awt.Image** class:

```
public abstract class Image
   extends Object
```

Field

Following are the fields for **java.awt.Image** class:

- **protected float accelerationPriority** -- Priority for accelerating this image.
- **static int SCALE_AREA_AVERAGING** -- Use the Area Averaging image scaling algorithm.
- **static int SCALE_DEFAULT** -- Use the default image-scaling algorithm.
- **static int SCALE_FAST** -- Choose an image-scaling algorithm that gives higher priority to scaling speed than smoothness of the scaled image.
- **static int SCALE_REPLICATE** -- Use the image scaling algorithm embodied in the ReplicateScaleFilter class.
- **static int SCALE_SMOOTH** -- Choose an image-scaling algorithm that gives higher priority to image smoothness than scaling speed.
- **static Object UndefinedProperty** -- The UndefinedProperty object should be returned whenever a property which was not defined for a particular image is fetched.

Class constructors

S.N.	Constructor & Description
1	**Image**

Class methods

S.N.	Method & Description
1	**void flush** Flushes all reconstructable resources being used by this Image object.
2	**float getAccelerationPriority** Returns the current value of the acceleration priority hint.
3	

ImageCapabilities getCapabilities*GraphicsConfigurationgc*

Returns an ImageCapabilities object which can be inquired as to the capabilities of this Image on the specified GraphicsConfiguration.

4

abstract Graphics getGraphics

Creates a graphics context for drawing to an off-screen image.

5

abstract int getHeight*ImageObserverobserver*

Determines the height of the image.

6

abstract Object getProperty*Stringname, ImageObserverobserver*

Gets a property of this image by name.

7

Image getScaledInstance*intwidth, intheight, inthints*

Creates a scaled version of this image.

8

abstract ImageProducer getSource

Gets the object that produces the pixels for the image.

9

abstract int getWidth*ImageObserverobserver*

Determines the width of the image.

10

void setAccelerationPriority*floatpriority*

Sets a hint for this image about how important acceleration is.

Methods inherited

This class inherits methods from the following classes:

- java.lang.Object

Image Example

Create the following java program using any editor of your choice in say **D:/ > AWT > com > tutorialspoint > gui >**

AwtControlDemo.java

```
package com.tutorialspoint.gui;

import java.awt.*;
import java.awt.event.*;

public class AwtControlDemo {

   private Frame mainFrame;
```

```
   private Label headerLabel;
   private Label statusLabel;
   private Panel controlPanel;

   public AwtControlDemo(){
      prepareGUI();
   }

   public static void main(String[] args){
      AwtControlDemo  awtControlDemo = new AwtControlDemo();
      awtControlDemo.showImageDemo();
   }

   private void prepareGUI(){
      mainFrame = new Frame("Java AWT Examples");
      mainFrame.setSize(400,400);
      mainFrame.setLayout(new GridLayout(3, 1));
      mainFrame.addWindowListener(new WindowAdapter() {
         public void windowClosing(WindowEvent windowEvent){
            System.exit(0);
         }
      });
      headerLabel = new Label();
      headerLabel.setAlignment(Label.CENTER);
      statusLabel = new Label();
      statusLabel.setAlignment(Label.CENTER);
      statusLabel.setSize(350,100);

      controlPanel = new Panel();
      controlPanel.setLayout(new FlowLayout());

      mainFrame.add(headerLabel);
      mainFrame.add(controlPanel);
      mainFrame.add(statusLabel);
      mainFrame.setVisible(true);
   }

   private void showImageDemo(){
      headerLabel.setText("Control in action: Image");

      controlPanel.add(new ImageComponent("resources/java.jpg"));
      mainFrame.setVisible(true);
   }

   class ImageComponent extends Component {

      BufferedImage img;

      public void paint(Graphics g) {
         g.drawImage(img, 0, 0, null);
      }

      public ImageComponent(String path) {
         try {
            img = ImageIO.read(new File(path));
         } catch (IOException e) {
            e.printStackTrace();
         }
      }

      public Dimension getPreferredSize() {
         if (img == null) {
            return new Dimension(100,100);
         } else {
            return new Dimension(img.getWidth(), img.getHeight());
         }
      }
   }
}
```

Compile the program using command prompt. Go to **D:/ > AWT** and type the following command.

```
D:\AWT>javac com\tutorialspoint\gui\AwtControlDemo.java
```

If no error comes that means compilation is successful. Run the program using following command.

```
D:\AWT>java com.tutorialspoint.gui.AwtControlDemo
```

Verify the following output