[All Packages](http://www.geom.uiuc.edu/~daeron/docs/apidocs/packages.html) [This Package](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.html) [Previous](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Frame.html#_top_) [Next](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.GridBagConstraints.html#_top_)

**Class java.awt.Graphics**

[java.lang.Object](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.Object.html#_top_)

|

+----java.awt.Graphics

public class **Graphics**

extends [Object](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.Object.html#_top_)

Graphics is the abstract base class for all graphic contexts for various devices.

**Version:**

1.23, 09/18/95

**Author:**

Sami Shaio, Arthur van Hoff

**Constructor Index**

 o  [**Graphics**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#Graphics())()

Constructs a new Graphics Object.

**Method Index**

 o  [**clearRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#clearRect(int, int, int, int))(int, int, int, int)

Clears the specified rectangle by filling it with the current background color of the current drawing surface.

 o  [**clipRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#clipRect(int, int, int, int))(int, int, int, int)

Clips to a rectangle.

 o  [**copyArea**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#copyArea(int, int, int, int, int, int))(int, int, int, int, int, int)

Copies an area of the screen.

 o  [**create**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#create())()

Creates a new Graphics Object that is a copy of the original Graphics Object.

 o  [**create**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#create(int, int, int, int))(int, int, int, int)

Creates a new Graphics Object with the specified parameters, based on the original Graphics Object.

 o  [**dispose**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#dispose())()

Disposes of this graphics context.

 o  [**draw3DRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#draw3DRect(int, int, int, int, boolean))(int, int, int, int, boolean)

Draws a highlighted 3-D rectangle.

 o  [**drawArc**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawArc(int, int, int, int, int, int))(int, int, int, int, int, int)

Draws an arc bounded by the specified rectangle from startAngle to endAngle.

 o  [**drawBytes**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawBytes(byte[], int, int, int, int))(byte[], int, int, int, int)

Draws the specified bytes using the current font and color.

 o  [**drawChars**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawChars(char[], int, int, int, int))(char[], int, int, int, int)

Draws the specified characters using the current font and color.

 o  [**drawImage**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawImage(java.awt.Image, int, int, java.awt.image.ImageObserver))(Image, int, int, ImageObserver)

Draws the specified image at the specified coordinate (x, y).

 o  [**drawImage**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawImage(java.awt.Image, int, int, int, int, java.awt.image.ImageObserver))(Image, int, int, int, int, ImageObserver)

Draws the specified image inside the specified rectangle.

 o  [**drawLine**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawLine(int, int, int, int))(int, int, int, int)

Draws a line between the coordinates (x1,y1) and (x2,y2).

 o  [**drawOval**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawOval(int, int, int, int))(int, int, int, int)

Draws an oval inside the specified rectangle using the current color.

 o  [**drawPolygon**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawPolygon(int[], int[], int))(int[], int[], int)

Draws a polygon defined by an array of x points and y points.

 o  [**drawPolygon**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawPolygon(java.awt.Polygon))(Polygon)

Draws a polygon defined by the specified point.

 o  [**drawRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawRect(int, int, int, int))(int, int, int, int)

Draws the outline of the specified rectangle using the current color.

 o  [**drawRoundRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawRoundRect(int, int, int, int, int, int))(int, int, int, int, int, int)

Draws an outlined rounded corner rectangle using the current color.

 o  [**drawString**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawString(java.lang.String, int, int))(String, int, int)

Draws the specified String using the current font and color.

 o  [**fill3DRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fill3DRect(int, int, int, int, boolean))(int, int, int, int, boolean)

Paints a highlighted 3-D rectangle using the current color.

 o  [**fillArc**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillArc(int, int, int, int, int, int))(int, int, int, int, int, int)

Fills an arc using the current color.

 o  [**fillOval**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillOval(int, int, int, int))(int, int, int, int)

Fills an oval inside the specified rectangle using the current color.

 o  [**fillPolygon**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillPolygon(int[], int[], int))(int[], int[], int)

Fills a polygon with the current color.

 o  [**fillPolygon**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillPolygon(java.awt.Polygon))(Polygon)

Fills the specified polygon with the current color.

 o  [**fillRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillRect(int, int, int, int))(int, int, int, int)

Fills the specified rectangle with the current color.

 o  [**fillRoundRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillRoundRect(int, int, int, int, int, int))(int, int, int, int, int, int)

Draws a rounded rectangle filled in with the current color.

 o  [**finalize**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#finalize())()

Disposes of this graphics context once it is no longer referenced.

 o  [**getClipRect**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getClipRect())()

Returns the bounding rectangle of the current clipping area.

 o  [**getColor**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getColor())()

Gets the current color.

 o  [**getFont**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFont())()

Gets the current font.

 o  [**getFontMetrics**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFontMetrics())()

Gets the current font metrics.

 o  [**getFontMetrics**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFontMetrics(java.awt.Font))(Font)

Gets the current font metrics for the specified font.

 o  [**scale**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#scale(float, float))(float, float)

Scales the graphics context.

 o  [**setColor**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setColor(java.awt.Color))(Color)

Sets the current color to the specified color.

 o  [**setFont**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setFont(java.awt.Font))(Font)

Sets the font for all subsequent text-drawing operations.

 o  [**setPaintMode**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setPaintMode())()

Sets the default paint mode to overwrite the destination with the current color.

 o  [**setXORMode**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setXORMode(java.awt.Color))(Color)

Sets the paint mode to alternate between the current color and the new specified color.

 o  [**toString**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#toString())()

Returns a String object representing this Graphic's value.

 o  [**translate**](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#translate(int, int))(int, int)

Translates the specified parameters into the origin of the graphics context.

**Constructors**

 o  **Graphics**

protected Graphics()

Constructs a new Graphics Object. Graphic contexts cannot be created directly. They must be obtained from another graphics context or either a component can create them.

**See Also:**

[getGraphics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Component.html#getGraphics), [create](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#create)

**Methods**

 o  **create**

public abstract [Graphics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#_top_) create()

Creates a new Graphics Object that is a copy of the original Graphics Object.

 o  **create**

public [Graphics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#_top_) create(int x,

int y,

int width,

int height)

Creates a new Graphics Object with the specified parameters, based on the original Graphics Object. This method translates the specified parameters, x and y, to the proper origin coordinates and then clips the Graphics Object to the area.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the area

height - the height of the area

**See Also:**

[translate](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#translate)

 o  **translate**

public abstract void translate(int x,

int y)

Translates the specified parameters into the origin of the graphics context. All subsequent operations on this graphics context will be relative to this origin.

**Parameters:**

x - the x coordinate

y - the y coordinate

**See Also:**

[scale](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#scale)

 o  **scale**

public abstract void scale(float sx,

float sy)

Scales the graphics context. All subsequent operations on this graphics context will be affected.

**Parameters:**

sx - the scaled x coordinate

sy - the scaled y coordinate

**See Also:**

[translate](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#translate)

 o  **getColor**

public abstract [Color](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Color.html#_top_) getColor()

Gets the current color.

**See Also:**

[setColor](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setColor)

 o  **setColor**

public abstract void setColor([Color](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Color.html#_top_) c)

Sets the current color to the specified color. All subsequent graphics operations will use this specified color.

**Parameters:**

c - the color to be set

**See Also:**

[Color](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Color.html#_top_), [getColor](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getColor)

 o  **setPaintMode**

public abstract void setPaintMode()

Sets the default paint mode to overwrite the destination with the current color.

 o  **setXORMode**

public abstract void setXORMode([Color](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Color.html#_top_) c1)

Sets the paint mode to alternate between the current color and the new specified color.

**Parameters:**

c1 - the second color

 o  **getFont**

public abstract [Font](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Font.html#_top_) getFont()

Gets the current font.

**See Also:**

[setFont](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#setFont)

 o  **setFont**

public abstract void setFont([Font](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Font.html#_top_) font)

Sets the font for all subsequent text-drawing operations.

**Parameters:**

font - the specified font

**See Also:**

[Font](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Font.html#_top_), [getFont](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFont), [drawString](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawString), [drawBytes](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawBytes), [drawChars](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawChars)

 o  **getFontMetrics**

public [FontMetrics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.FontMetrics.html#_top_) getFontMetrics()

Gets the current font metrics.

**See Also:**

[getFont](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFont)

 o  **getFontMetrics**

public abstract [FontMetrics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.FontMetrics.html#_top_) getFontMetrics([Font](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Font.html#_top_) f)

Gets the current font metrics for the specified font.

**Parameters:**

f - the specified font

**See Also:**

[getFont](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFont), [getFontMetrics](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getFontMetrics)

 o  **getClipRect**

public abstract [Rectangle](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Rectangle.html#_top_) getClipRect()

Returns the bounding rectangle of the current clipping area.

**See Also:**

[clipRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#clipRect)

 o  **clipRect**

public abstract void clipRect(int x,

int y,

int width,

int height)

Clips to a rectangle. The resulting clipping area is the intersection of the current clipping area and the specified rectangle. Graphic operations have no effect outside of the clipping area.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[getClipRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#getClipRect)

 o  **copyArea**

public abstract void copyArea(int x,

int y,

int width,

int height,

int dx,

int dy)

Copies an area of the screen.

**Parameters:**

x - the x-coordinate of the source

y - the y-coordinate of the source

width - the width

height - the height

dx - the horizontal distance

dy - the vertical distance

 o  **drawLine**

public abstract void drawLine(int x1,

int y1,

int x2,

int y2)

Draws a line between the coordinates (x1,y1) and (x2,y2). The line is drawn below and to the left of the logical coordinates.

**Parameters:**

x1 - the first point's x coordinate

y1 - the first point's y coordinate

x2 - the second point's x coordinate

y2 - the second point's y coordinate

 o  **fillRect**

public abstract void fillRect(int x,

int y,

int width,

int height)

Fills the specified rectangle with the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[drawRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawRect), [clearRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#clearRect)

 o  **drawRect**

public void drawRect(int x,

int y,

int width,

int height)

Draws the outline of the specified rectangle using the current color. Use drawRect(x, y, width-1, height-1) to draw the outline inside the specified rectangle.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[fillRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillRect), [clearRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#clearRect)

 o  **clearRect**

public abstract void clearRect(int x,

int y,

int width,

int height)

Clears the specified rectangle by filling it with the current background color of the current drawing surface. Which drawing surface it selects depends on how the graphics context was created.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[fillRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillRect), [drawRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawRect)

 o  **drawRoundRect**

public abstract void drawRoundRect(int x,

int y,

int width,

int height,

int arcWidth,

int arcHeight)

Draws an outlined rounded corner rectangle using the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

arcWidth - the diameter of the arc

arcHeight - the radius of the arc

**See Also:**

[fillRoundRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillRoundRect)

 o  **fillRoundRect**

public abstract void fillRoundRect(int x,

int y,

int width,

int height,

int arcWidth,

int arcHeight)

Draws a rounded rectangle filled in with the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

arcWidth - the diameter of the arc

arcHeight - the radius of the arc

**See Also:**

[drawRoundRect](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawRoundRect)

 o  **draw3DRect**

public void draw3DRect(int x,

int y,

int width,

int height,

boolean raised)

Draws a highlighted 3-D rectangle.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

raised - a boolean that states whether the rectangle is raised or not

 o  **fill3DRect**

public void fill3DRect(int x,

int y,

int width,

int height,

boolean raised)

Paints a highlighted 3-D rectangle using the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

raised - a boolean that states whether the rectangle is raised or not

 o  **drawOval**

public abstract void drawOval(int x,

int y,

int width,

int height)

Draws an oval inside the specified rectangle using the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[fillOval](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillOval)

 o  **fillOval**

public abstract void fillOval(int x,

int y,

int width,

int height)

Fills an oval inside the specified rectangle using the current color.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

**See Also:**

[drawOval](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawOval)

 o  **drawArc**

public abstract void drawArc(int x,

int y,

int width,

int height,

int startAngle,

int arcAngle)

Draws an arc bounded by the specified rectangle from startAngle to endAngle. 0 degrees is at the 3-o'clock position.Positive arc angles indicate counter-clockwise rotations, negative arc angles are drawn clockwise.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

startAngle - the beginning angle

arcAngle - the angle of the arc (relative to startAngle).

**See Also:**

[fillArc](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillArc)

 o  **fillArc**

public abstract void fillArc(int x,

int y,

int width,

int height,

int startAngle,

int arcAngle)

Fills an arc using the current color. This generates a pie shape.

**Parameters:**

x - the x coordinate

y - the y coordinate

width - the width of the arc

height - the height of the arc

startAngle - the beginning angle

arcAngle - the angle of the arc (relative to startAngle).

**See Also:**

[drawArc](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawArc)

 o  **drawPolygon**

public abstract void drawPolygon(int xPoints[],

int yPoints[],

int nPoints)

Draws a polygon defined by an array of x points and y points.

**Parameters:**

xPoints - an array of x points

yPoints - an array of y points

nPoints - the total number of points

**See Also:**

[fillPolygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillPolygon)

 o  **drawPolygon**

public void drawPolygon([Polygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Polygon.html#_top_) p)

Draws a polygon defined by the specified point.

**Parameters:**

p - the specified polygon

**See Also:**

[fillPolygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#fillPolygon)

 o  **fillPolygon**

public abstract void fillPolygon(int xPoints[],

int yPoints[],

int nPoints)

Fills a polygon with the current color.

**Parameters:**

xPoints - an array of x points

yPoints - an array of y points

nPoints - the total number of points

**See Also:**

[drawPolygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawPolygon)

 o  **fillPolygon**

public void fillPolygon([Polygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Polygon.html#_top_) p)

Fills the specified polygon with the current color.

**Parameters:**

p - the polygon

**See Also:**

[drawPolygon](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawPolygon)

 o  **drawString**

public abstract void drawString([String](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.String.html#_top_) str,

int x,

int y)

Draws the specified String using the current font and color. The x,y position is the starting point of the baseline of the String.

**Parameters:**

str - the String to be drawn

x - the x coordinate

y - the y coordinate

**See Also:**

[drawChars](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawChars), [drawBytes](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawBytes)

 o  **drawChars**

public void drawChars(char data[],

int offset,

int length,

int x,

int y)

Draws the specified characters using the current font and color.

**Parameters:**

data - the array of characters to be drawn

offset - the start offset in the data

length - the number of characters to be drawn

x - the x coordinate

y - the y coordinate

**See Also:**

[drawString](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawString), [drawBytes](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawBytes)

 o  **drawBytes**

public void drawBytes(byte data[],

int offset,

int length,

int x,

int y)

Draws the specified bytes using the current font and color.

**Parameters:**

data - the data to be drawn

offset - the start offset in the data

length - the number of bytes that are drawn

x - the x coordinate

y - the y coordinate

**See Also:**

[drawString](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawString), [drawChars](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#drawChars)

 o  **drawImage**

public abstract boolean drawImage([Image](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Image.html#_top_) img,

int x,

int y,

[ImageObserver](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.image.ImageObserver.html#_top_) observer)

Draws the specified image at the specified coordinate (x, y). If the image is incomplete the image observer will be notified later.

**Parameters:**

img - the specified image to be drawn

x - the x coordinate

y - the y coordinate

observer - notifies if the image is complete or not

**See Also:**

[Image](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Image.html#_top_), [ImageObserver](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.image.ImageObserver.html#_top_)

 o  **drawImage**

public abstract boolean drawImage([Image](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Image.html#_top_) img,

int x,

int y,

int width,

int height,

[ImageObserver](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.image.ImageObserver.html#_top_) observer)

Draws the specified image inside the specified rectangle. The image is scaled if necessary. If the image is incomplete the image observer will be notified later.

**Parameters:**

img - the specified image to be drawn

x - the x coordinate

y - the y coordinate

width - the width of the rectangle

height - the height of the rectangle

observer - notifies if the image is complete or not

**See Also:**

[Image](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Image.html#_top_), [ImageObserver](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.image.ImageObserver.html#_top_)

 o  **dispose**

public abstract void dispose()

Disposes of this graphics context. The Graphics context cannot be used after being disposed of.

**See Also:**

[finalize](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#finalize)

 o  **finalize**

public void finalize()

Disposes of this graphics context once it is no longer referenced.

**See Also:**

[dispose](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Graphics.html#dispose)

 o  **toString**

public [String](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.String.html#_top_) toString()

Returns a String object representing this Graphic's value.

**Overrides:**

[toString](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.Object.html#toString()) in class [Object](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.lang.Object.html#_top_)

[All Packages](http://www.geom.uiuc.edu/~daeron/docs/apidocs/packages.html) [This Package](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.html) [Previous](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.Frame.html#_top_) [Next](http://www.geom.uiuc.edu/~daeron/docs/apidocs/java.awt.GridBagConstraints.html#_top_)