Class java.awt.Font

```
java.lang.Object
   |
   +----java.awt.Font
```

public class **Font**
extends Object

A class that produces font objects.

Version:
    1.12, 10/03/95
Author:
    Sami Shaio, Arthur van Hoff, Jim Graham

Variable Index

o **BOLD**
        The bold style constant.
o **ITALIC**
        The italicized style constant.
o **PLAIN**
        The plain style constant.
o **name**
        The logical name of this font.
o **size**
        The point size of this font.
o **style**
        The style of the font.

Constructor Index

o **Font**(String, int, int)
        Creates a new font with the specified name, style and point size.

Method Index

o **equals**(Object)

    Compares this object to the specifed object.

o **getFamily**()

    Gets the platform specific family name of the font.

o **getFont**(String)

    Gets a font from the system properties list.

o **getFont**(String, Font)

    Gets the specified font from the system properties list.

o **getName**()

    Gets the logical name of the font.

o **getSize**()

    Gets the point size of the font.

o **getStyle**()

    Gets the style of the font.

o **hashCode**()

    Returns a hashcode for this font.

o **isBold**()

    Returns true if the font is bold.

o **isItalic**()

    Returns true if the font is italic.

o **isPlain**()

    Returns true if the font is plain.

o **toString**()

    Converts this object to a String representation.

Variables

o **PLAIN**

```
public final static int PLAIN
```

The plain style constant. This can be added with the other style constants for mixed styles.

o **BOLD**

```
public final static int BOLD
```

The bold style constant. This can be added with the other style constants for mixed styles.

o **ITALIC**

```
public final static int ITALIC
```

The italicized style constant. This can be added with the other style constants for

mixed styles.

o **name**

```
protected String name
```

The logical name of this font.

o **style**

```
protected int style
```

The style of the font. This is the sum of the constants PLAIN, BOLD, or ITALIC.

o **size**

```
protected int size
```

The point size of this font.

Constructors

o **Font**

```
public Font(String name,
            int style,
            int size)
```

Creates a new font with the specified name, style and point size.
Parameters:
    name – the font name
    style – the constant style used
    size – the point size of the font
See Also:
    getFontList

Methods

o **getFamily**

```
public String getFamily()
```

Gets the platform specific family name of the font. Use getName to get the logical name of the font.
See Also:
    getName

o **getName**

```
public String getName()
```

> Gets the logical name of the font.
> **See Also:**
> > getFamily

o **getStyle**

```
public int getStyle()
```

> Gets the style of the font.
> **See Also:**
> > isPlain, isBold, isItalic

o **getSize**

```
public int getSize()
```

> Gets the point size of the font.

o **isPlain**

```
public boolean isPlain()
```

> Returns true if the font is plain.
> **See Also:**
> > getStyle

o **isBold**

```
public boolean isBold()
```

> Returns true if the font is bold.
> **See Also:**
> > getStyle

o **isItalic**

```
public boolean isItalic()
```

> Returns true if the font is italic.
> **See Also:**
> > getStyle

o **getFont**

```
public static Font getFont(String nm)
```

Gets a font from the system properties list.
Parameters:
    nm – the property name

o **getFont**

```
public static Font getFont(String nm,
                           Font font)
```

Gets the specified font from the system properties list.
Parameters:
    nm – the property name
    font – a default font to return if property 'nm' is not defined

o **hashCode**

```
public int hashCode()
```

Returns a hashcode for this font.
Overrides:
    hashCode in class Object

o **equals**

```
public boolean equals(Object obj)
```

Compares this object to the specifed object.
Parameters:
    obj – the object to compare with
Returns:
    true if the objects are the same; false otherwise.
Overrides:
    equals in class Object

o **toString**

```
public String toString()
```

Converts this object to a String representation.
Overrides:
    toString in class Object