

Introduction

The class **Frame** is a top level window with border and title. It uses BorderLayout as default layout manager.

Class declaration

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

```
public class Frame
    extends Window
        implements MenuContainer
```

Field

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

- **static float BOTTOM_ALIGNMENT** -- Ease-of-use constant for getAlignmentY.
- **static int CROSSHAIR_CURSOR** -- Deprecated. replaced by Cursor.CROSSHAIR_CURSOR.
- **static int DEFAULT_CURSOR** -- Deprecated. replaced by Cursor.DEFAULT_CURSOR.
- **static int E_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.E_RESIZE_CURSOR.
- **static int HAND_CURSOR** -- Deprecated. replaced by Cursor.HAND_CURSOR.
- **static int ICONIFIED** -- This state bit indicates that frame is iconified.
- **static int MAXIMIZED_BOTH** -- This state bit mask indicates that frame is fully maximized *that is both horizontally and vertically.*
- **static int MAXIMIZED_HORIZ** -- This state bit indicates that frame is maximized in the horizontal direction.
- **static int MAXIMIZED_VERT** -- This state bit indicates that frame is maximized in the vertical direction.
- **static int MOVE_CURSOR** -- Deprecated. replaced by Cursor.MOVE_CURSOR.
- **static int N_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.N_RESIZE_CURSOR.
- **static int NE_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.NE_RESIZE_CURSOR.
- **static int NORMAL** -- Frame is in the "normal" state.
- **static int NW_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.NW_RESIZE_CURSOR.
- **static int S_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.S_RESIZE_CURSOR.
- **static int SE_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.SE_RESIZE_CURSOR.
- **static int SW_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.SW_RESIZE_CURSOR.
- **static int TEXT_CURSOR** -- Deprecated. replaced by Cursor.TEXT_CURSOR.
- **static int W_RESIZE_CURSOR** -- Deprecated. replaced by Cursor.W_RESIZE_CURSOR.
- **static int WAIT_CURSOR** -- Deprecated. replaced by Cursor.WAIT_CURSOR.

Class constructors

S.N.	Constructor & Description
------	---------------------------

- | | |
|---|--|
| 1 | Frame
Constructs a new instance of Frame that is initially invisible. |
| 2 | FrameGraphicsConfigurationgc
Constructs a new, initially invisible Frame with the specified GraphicsConfiguration. |
| 3 | FrameStringtitle
Constructs a new, initially invisible Frame object with the specified title. |
| 4 | FrameStringtitle, GraphicsConfigurationgc
Constructs a new, initially invisible Frame object with the specified title and a GraphicsConfiguration. |

Class methods

S.N.	Method & Description
------	----------------------

- | | |
|---|---|
| 1 | void addNotify
Makes this Frame displayable by connecting it to a native screen resource. |
| 2 | AccessibleContext getAccessibleContext
Gets the AccessibleContext associated with this Frame. |
| 3 | int getCursorType
Deprecated. As of JDK version 1.1, replaced by Component.setCursor. |
| 4 | int getExtendedState
Gets the state of this frame. |
| 5 | static Frame[] getFrames
Returns an array of all Frames created by this application. |
| 6 | Image getIconImage
Returns the image to be displayed as the icon for this frame. |

7	Rectangle getMaximizedBounds	Gets maximized bounds for this frame.
8	MenuBar getMenuBar	Gets the menu bar for this frame.
9	int getState	Gets the state of this frame <i>obsolete</i> .
10	String getTitle	Gets the title of the frame.
11	boolean isResizable	Indicates whether this frame is resizable by the user.
12	boolean isUndecorated	Indicates whether this frame is undecorated.
13	protected String paramString	Returns a string representing the state of this Frame.
14	void removeMenuComponent <i>m</i>	Removes the specified menu bar from this frame.
15	void removeNotify	Makes this Frame undisplayable by removing its connection to its native screen resource.
16	void setCursor <i>int cursorType</i>	Deprecated. As of JDK version 1.1, replaced by <code>Component.setCursorCursor</code> .
17	void setExtendedState <i>int state</i>	Sets the state of this frame.
18	void setIconImage <i>Image image</i>	Sets the image to be displayed as the icon for this window.

- 19 **void setMaximizedBounds***Rectanglebounds*
Sets the maximized bounds for this frame.
- 20 **void setMenuBar***MenuBarmb*
Sets the menu bar for this frame to the specified menu bar.
- 21 **void setResizable***booleanresizable*
Sets whether this frame is resizable by the user.
- 22 **void setState***intstate*
Sets the state of this frame *obsolete*.
- 23 **void setTitle***Stringtitle*
Sets the title for this frame to the specified string.
- 24 **void setUndecorated***booleanundecorated*
Disables or enables decorations for this frame.

Methods inherited

This class inherits methods from the following classes:

- java.awt.Window
- java.awt.Container
- java.awt.Component
- java.lang.Object

Frame Example

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

AwtContainerDemo.java

```
package com.tutorialspoint.gui;

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

public class AwtContainerDemo {
    private Frame mainFrame;
    private Label headerLabel;
    private Label statusLabel;
    private Panel controlPanel;
    private Label msgLabel;

    public AwtContainerDemo(){
```

```

    prepareGUI();
}

public static void main(String[] args){
    AwtContainerDemo awtContainerDemo = new AwtContainerDemo();
    awtContainerDemo.showFrameDemo();
}

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);

    msglabel = new Label();
    msglabel.setAlignment(Label.CENTER);
    msglabel.setText("Welcome to Tutorialspoint AWT Tutorial.");

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

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

private void showFrameDemo(){
    headerLabel.setText("Container in action: Frame");

    final Frame frame = new Frame();
    frame.setSize(300, 300);
    frame.setLayout(new FlowLayout());
    frame.add(msglabel);
    frame.addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent windowEvent){
            frame.dispose();
        }
    });
    Button okButton = new Button("Open a Frame");

    okButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            statusLabel.setText("A Frame shown to the user.");
            frame.setVisible(true);
        }
    });
    controlPanel.add(okButton);

    mainFrame.setVisible(true);
}
}

```

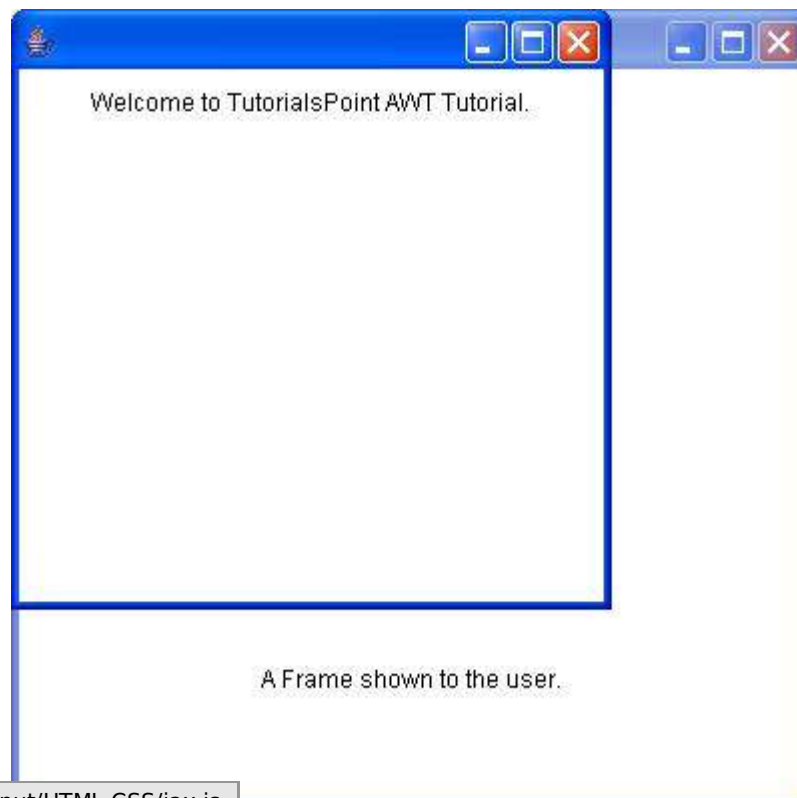
Compile the program using command prompt. Go to **D:/ > AWT** and type the following command.

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

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

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

Verify the following output



Loading [MathJax]/jax/output/HTML-CSS/jax.js