AWT FLOWLAYOUT CLASS

Introduction

The class **FlowLayout** components in a left-to-right flow.

Class declaration

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

```
public class FlowLayout
   extends Object
      implements LayoutManager, Serializable
```

Field

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

- **static int CENTER** -- This value indicates that each row of components should be centered.
- **static int LEADING** -- This value indicates that each row of components should be justified to the leading edge of the container's orientation, for example, to the left in left-to-right orientations.
- **static int LEFT** -- This value indicates that each row of components should be left-justified.
- **static int RIGHT** -- This value indicates that each row of components should be right-justified.
- **static int TRAILING** -- This value indicates that each row of components should be justified to the trailing edge of the container's orientation, for example, to the right in left-to-right orientations.

Class constructors

S.N.	Constructor & Description
1	**FlowLayout** Constructs a new FlowLayout with a centered alignment and a default 5-unit horizontal and vertical gap.
2	**FlowLayout*intalign*** Constructs a new FlowLayout with the specified alignment and a default 5-unit horizontal and vertical gap.
3	**FlowLayout*intalign, inthgap, intvgap*** Creates a new flow layout manager with the indicated alignment and the indicated horizontal and vertical gaps.

Class methods

S.N.	Method & Description
1	**void addLayoutComponent***Stringname, Componentcomp* Adds the specified component to the layout.
2	**int getAlignment** Gets the alignment for this layout.
3	**int getHgap** Gets the horizontal gap between components.
4	**int getVgap** Gets the vertical gap between components.
5	**void layoutContainer***Containertarget* Lays out the container.
6	**Dimension minimumLayoutSize***Containertarget* Returns the minimum dimensions needed to layout the visible components contained in the specified target container.
7	**Dimension preferredLayoutSize***Containertarget* Returns the preferred dimensions for this layout given the visible components in the specified target container.
8	**void removeLayoutComponent***Componentcomp* Removes the specified component from the layout.
9	**void setAlignment***intalign* Sets the alignment for this layout.
10	**void setHgap***inthgap* Sets the horizontal gap between components.
11	**void setVgap***intvgap* Sets the vertical gap between components.

String toString

Returns a string representation of this FlowLayout object and its values.

Methods inherited

This class inherits methods from the following classes:

- java.lang.Object

FlowLayout Example

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

AwtLayoutDemo.java

```
package com.tutorialspoint.gui;

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

public class AwtLayoutDemo {
   private Frame mainFrame;
   private Label headerLabel;
   private Label statusLabel;
   private Panel controlPanel;
   private Label msglabel;

   public AwtLayoutDemo(){
      prepareGUI();
   }

   public static void main(String[] args){
      AwtLayoutDemo  awtLayoutDemo = new AwtLayoutDemo();
      awtLayoutDemo.showFlowLayoutDemo();
   }

   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 showFlowLayoutDemo(){
      headerLabel.setText("Layout in action: FlowLayout");

      Panel panel = new Panel();
      panel.setBackground(Color.darkGray);
      panel.setSize(200,200);
      FlowLayout layout = new FlowLayout();
      layout.setHgap(10);
      layout.setVgap(10);
      panel.setLayout(layout);
      panel.add(new Button("OK"));
      panel.add(new Button("Cancel"));

      controlPanel.add(panel);

      mainFrame.setVisible(true);
   }
}
```

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

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

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

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

Verify the following output