How to Create Menu in Swing

◆ **zentut.com**/java-swing/how-to-create-menu-in-swing

Menu components in Swing are subclasses of JComponent so they have all functionality of a normal Swing component.

Here are some remarkable characteristics of the Java Swing menu:

- Menu items can have both icons and text.
- Menu items can be checkboxes or radio buttons.
- Each menu item can have one keyboard accelerator that display after the menu item text.
- Almost standard Swing components can be used as menu items.

Here is the class hierarchy of Swing menu system.
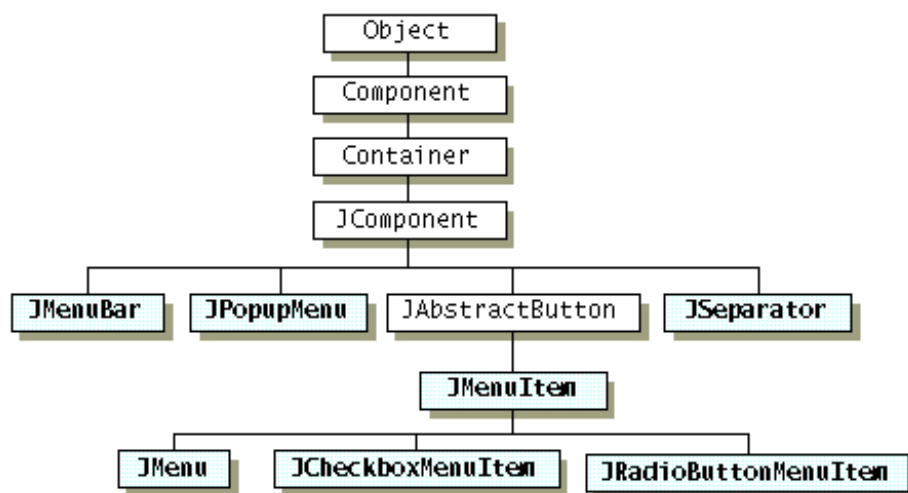
Create Menu Items with JMenuItem

In order to create menu items in Swing, you need to create new instances of *JMenuItem* and set different properties for them. You can create menu item with both text and icon.

Here are the constructors of *JMenuItem:*

Constructors	Description
JMenuItem()	Creates a JMenuItem instance without icon or text.
JMenuItem(Icon icon)	Creates a JMenuItem instance with a given icon.
JMenuItem(String text)	Creates a JMenuItem instance with a given text.
JMenuItem(String text, Icon icon)	Creates a JMenuItem instance with a given text and icon
JMenuItem(String text, int mnemonic)	Creates a JMenuItem instance with the given text and keyboard mnemonic.

Constructors	Description
JMenuItem(Action a)	Creates a JMenuItem instance whose properties are taken from the a given Action.

Here is the code snippet to create a menu item:

```
1  menuItem=newJMenuItem("New Project...",
2  newImageIcon("images/newproject.png"));
3  menuItem.setMnemonic(KeyEvent.VK_P);
4  menuItem.setAccelerator(KeyStroke.getKeyStroke(
5  KeyEvent.VK_1,ActionEvent.ALT_MASK));
6  menuItem.getAccessibleContext().setAccessibleDescription(
7  "New Project");
```

Create Menu with JMenu

After having menu items, you need to add them to a menu. In order to create a menu you need to use JMenu class. JMenu class represents the menu which can attach to a menu bar or another menu. Menu directly attached to a menu bar is known as top-level menu. If the menu is attached to a menu, it is called sub-menu.

Here are the constructors of JMenu class:

Constructors	Description
JMenu()	Creates an instance of JMenu without text.
JMenu(String s)	Creates an instance of JMenu a given text.
JMenu(String s, boolean tearOffMenu)	Creates an instance of JMenu a given text. and specify the menu as a tear-off menu or not.
JMenu(Action a)	Creates an instance of JMenu whose properties are taken from the specified Action.

Here is the code snippet to create a menu and add a menu item to that menu.
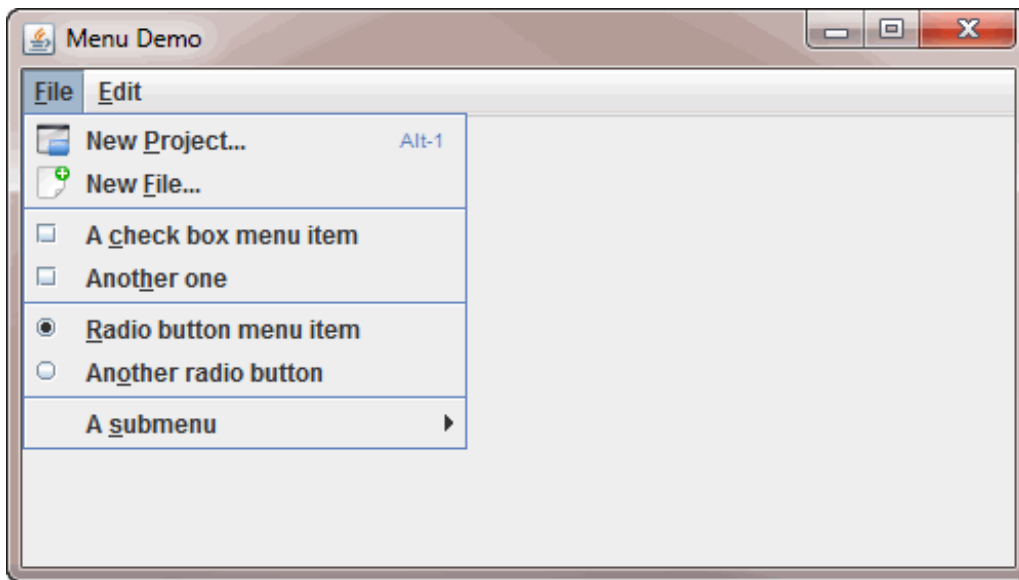
```
1  //Build the File Menu.
2  menu=newJMenu("File");
3  menu.setMnemonic(KeyEvent.VK_F);
4  menu.getAccessibleContext().setAccessibleDescription("Dealing with Files");
5  // create menu item and add it to the menu
6  menuItem=newJMenuItem("New File...",
7  newImageIcon("images/newfile.png"));
8  menuItem.setMnemonic(KeyEvent.VK_F);
9  menu.add(menuItem);
```

Create menu bar and attach it to a frame

JMenuBar class represents menu bar and can attach it to a frame. One or more menu can be added to a menu bar by using *add()* method. To attach menu bar to a frame you use method *setJMenuBar()* of frame.

Here is the menu demo application:

```
1   packagejmenudemo;
2   importjavax.swing.*;
3   importjava.awt.event.*;
4   publicclassMain{
5   publicstaticJMenuBar createMenuBar(){
6   JMenuBar menuBar;
7   JMenu menu,submenu;
8   JMenuItem menuItem;
9   JRadioButtonMenuItem rdmi;
10  JCheckBoxMenuItem cbmi;
11  //Create the menu bar.
12  menuBar=newJMenuBar();
13  //Build the File Menu.
14  menu=newJMenu("File");
15  menu.setMnemonic(KeyEvent.VK_F);
16  menu.getAccessibleContext().setAccessibleDescription("Dealing with Files");
17  menuBar.add(menu);
18  //a group of JMenuItems
19  menuItem=newJMenuItem("New Project...",
20  newImageIcon("images/newproject.png"));
21  menuItem.setMnemonic(KeyEvent.VK_P);
22  menuItem.setAccelerator(KeyStroke.getKeyStroke(
23  KeyEvent.VK_1,ActionEvent.ALT_MASK));
24  menuItem.getAccessibleContext().setAccessibleDescription(
25  "New Project");
26  menu.add(menuItem);
27  menuItem=newJMenuItem("New File...",
28  newImageIcon("images/newfile.png"));
29  menuItem.setMnemonic(KeyEvent.VK_F);
30  menu.add(menuItem);
31  //a group of check box menu items
32  menu.addSeparator();
33  cbmi=newJCheckBoxMenuItem("A check box menu item");
34  cbmi.setMnemonic(KeyEvent.VK_C);
35  menu.add(cbmi);
36  cbmi=newJCheckBoxMenuItem("Another one");
37  cbmi.setMnemonic(KeyEvent.VK_H);
38  menu.add(cbmi);
39  //a group of radio button menu items
40  menu.addSeparator();
41  ButtonGroup group=newButtonGroup();
42  rdmi=newJRadioButtonMenuItem("Radio button menu item");
43  rdmi.setSelected(true);
44  rdmi.setMnemonic(KeyEvent.VK_R);
45  group.add(rdmi);
46  menu.add(rdmi);
47  rdmi=newJRadioButtonMenuItem("Another radio button");
48  rdmi.setMnemonic(KeyEvent.VK_O);
```

```java
49    group.add(rdmi);
50    menu.add(rdmi);
51    //a submenu
52    menu.addSeparator();
53    submenu=newJMenu("A submenu");
54    submenu.setMnemonic(KeyEvent.VK_S);
55    menuItem=newJMenuItem("Menu item in the submenu" );
56    menuItem.setAccelerator(KeyStroke.getKeyStroke(
57    KeyEvent.VK_2,ActionEvent.ALT_MASK));
58    submenu.add(menuItem);
59    menuItem=newJMenuItem("Another menu item");
60    submenu.add(menuItem);
61    menu.add(submenu);
62    //Build Edit menu in the menu bar.
63    menu=newJMenu("Edit");
64    menu.setMnemonic(KeyEvent.VK_E);
65    menu.getAccessibleContext().setAccessibleDescription(
66    "Edit Menu");
67    menuBar.add(menu);
68    returnmenuBar;
69    }
70    publicstaticvoidmain(String[]args){
71    finalJFrame frame=newJFrame("Menu Demo");
72    frame.setJMenuBar(createMenuBar());
73    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
74    frame.setSize(500,300);
75    frame.setVisible(true);
76    }
77    }
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
```