**JMenuItem**

**Constructors:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.util.Random;

public class menuitem3 {

    public static void main(String[] args) {

        // Create a JFrame

        JFrame frame = new JFrame("JMenuItem Constructors Example");

        frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        frame.setSize(400, 200);

        frame.setLayout(new BorderLayout());

        // Create a JPanel to change its background color

        JPanel panel = new JPanel();

        frame.add(panel, BorderLayout.CENTER);

        // Create a JMenuBar

        JMenuBar menuBar = new JMenuBar();

        // Create a JMenu

        JMenu menu = new JMenu("Menu");

        // JMenuItem with no set text or icon

        JMenuItem menuItem1 = new JMenuItem();

        //menuItem1.setText("Empty Item");

        // JMenuItem with specified text

        JMenuItem menuItem2 = new JMenuItem("Item with Text");

        // JMenuItem with specified text and keyboard mnemonic

        JMenuItem menuItem3 = new JMenuItem("Item with Mnemonic", 'M');

        menuItem3.addActionListener(e -> {

            // Show a dialog box when the menu item is clicked

            JOptionPane.showMessageDialog(frame, "Mnemonic Item Clicked!", "Action Triggered", JOptionPane.INFORMATION\_MESSAGE);

        });

        // JMenuItem with specified text and icon

        Icon icon = new ImageIcon("icon.png"); // Placeholder for icon

        JMenuItem menuItem4 = new JMenuItem("Item with Icon", icon);

        // JMenuItem with an Action that changes the background color of the panel

        Action changeColorAction = new AbstractAction("Change Background Color") {

            @Override

            public void actionPerformed(ActionEvent e) {

                // Generate a random color

                Random rand = new Random();

                float r = rand.nextFloat();

                float g = rand.nextFloat();

                float b = rand.nextFloat();

                Color randomColor = new Color(r, g, b);

                // Set the panel background to the new random color

                panel.setBackground(randomColor);

            }

        };

        JMenuItem menuItem5 = new JMenuItem(changeColorAction);

        // JMenuItem with specified icon

        JMenuItem menuItem6 = new JMenuItem(icon);

        // Add all menu items to the menu

        menu.add(menuItem1);

        menu.add(menuItem2);

        menu.add(menuItem3); // This one will show a message dialog when clicked

        menu.add(menuItem4);

        menu.add(menuItem5); // This one changes background color

        menu.add(menuItem6);

        // Add menu to the menu bar

        menuBar.add(menu);

        // Set the menu bar to the frame

        frame.setJMenuBar(menuBar);

        // Set the frame visibility

        frame.setVisible(true);

    }

}

**Methods:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.util.Random;

public class menufunc{

public static void main(String[] args) {

// Create a JFrame

JFrame frame = new JFrame("JMenuItem Methods Example");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(500, 400);

frame.setLayout(new BorderLayout());

// Create a JPanel to change its background color

JPanel panel = new JPanel();

frame.add(panel, BorderLayout.CENTER);

// Create a JMenuBar

JMenuBar menuBar = new JMenuBar();

// Create a JMenu

JMenu menu = new JMenu("Menu");

// JMenuItem with setText() and getText()

JMenuItem menuItem1 = new JMenuItem();

menuItem1.setText("Item with Text and Icon");

Icon icon = new ImageIcon("icon.png"); // Placeholder for an icon

menuItem1.setIcon(icon); // Set an icon

menuItem1.setForeground(Color.RED); // Text color set to red

menuItem1.setBackground(Color.YELLOW); // Background color set to yellow

//menuItem1.setOpaque(false); // Make sure the background color shows

menuItem1.setFont(new Font("Serif", Font.BOLD, 16)); // Set a custom font

// Triggered when the menu item is clicked

menuItem1.addActionListener(e -> {

JOptionPane.showMessageDialog(frame, "You clicked: " + menuItem1.getText());

});

// JMenuItem with setEnabled()

JMenuItem menuItem2 = new JMenuItem("Disabled Item");

menuItem2.setEnabled(false); // Disable the item

// JMenuItem with setAccelerator() and setMnemonic()

JMenuItem menuItem3 = new JMenuItem("Item with Shortcut and Mnemonic");

menuItem3.setAccelerator(KeyStroke.getKeyStroke('S', Toolkit.getDefaultToolkit().getMenuShortcutKeyMaskEx())); // Ctrl + S or Cmd + S

menuItem3.setMnemonic('M'); // Pressing Alt+M will trigger this item

menuItem3.addActionListener(e -> {

JOptionPane.showMessageDialog(frame, "Shortcut or Mnemonic triggered!");

});

// JMenuItem with setHorizontalTextPosition()

JMenuItem menuItem4 = new JMenuItem("Text and Icon Position", icon);

menuItem4.setHorizontalTextPosition(SwingConstants.LEFT); // Text on the left of icon

// JMenuItem with setToolTipText()

JMenuItem menuItem5 = new JMenuItem("Item with Tooltip");

menuItem5.setToolTipText("This is a helpful tooltip! Hover over me.");

// JMenuItem with doClick() to simulate a click

JMenuItem menuItem6 = new JMenuItem("Simulate Click");

menuItem6.addActionListener(e -> JOptionPane.showMessageDialog(frame, "Click simulated programmatically"));

// Simulate a click on menuItem6 after 20 seconds

new Timer(20000, new AbstractAction() {

@Override

public void actionPerformed(ActionEvent e) {

menuItem6.doClick();

}

}).start();

// Add all menu items to the menu

menu.add(menuItem1); // Text, Icon, Foreground, Background, Font

menu.add(menuItem2); // Disabled Item

menu.add(menuItem3); // Shortcut (Accelerator) and Mnemonic

menu.add(menuItem4); // Text and Icon Positioning

menu.add(menuItem5); // Tooltip

menu.add(menuItem6); // Simulate Click

// Add menu to the menu bar

menuBar.add(menu);

// Set the menu bar to the frame

frame.setJMenuBar(menuBar);

// Set the frame visibility

frame.setVisible(true);

}}

**Sub Menu Item**

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class submenuitem {

public static void main(String[] args) {

// Create a JFrame

JFrame frame = new JFrame("SubMenu Example");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(400, 300);

// Create a JMenuBar

JMenuBar menuBar = new JMenuBar();

// Create a main JMenu

JMenu mainMenu = new JMenu("Main Menu");

// Create a submenu as a JMenu

JMenu subMenu = new JMenu("Sub Menu");

// Create JMenuItems for the submenu

JMenuItem subMenuItem1 = new JMenuItem("Sub Item 1");

JMenuItem subMenuItem2 = new JMenuItem("Sub Item 2");

JMenuItem subMenuItem3 = new JMenuItem("Sub Item 3");

// Add action listeners to the submenu items

subMenuItem1.addActionListener(e -> JOptionPane.showMessageDialog(frame, "Sub Item 1 clicked"));

subMenuItem2.addActionListener(e -> JOptionPane.showMessageDialog(frame, "Sub Item 2 clicked"));

subMenuItem3.addActionListener(e -> JOptionPane.showMessageDialog(frame, "Sub Item 3 clicked"));

// Add submenu items to the submenu

subMenu.add(subMenuItem1);

subMenu.add(subMenuItem2);

subMenu.add(subMenuItem3);

// Add the submenu to the main menu

mainMenu.add(subMenu);

// Add the main menu to the menu bar

menuBar.add(mainMenu);

// Set the menu bar for the frame

frame.setJMenuBar(menuBar);

// Set frame visibility

frame.setVisible(true);

}}