

Java Swing – JOptionPane showMessageDialog example

 mkyong.com/swing/java-swing-how-to-make-a-simple-dialog

This is a review of the `showMessageDialog()` method of `JOptionPane` Class. This method is a quick and easy way to tell the user about something that has happened . The `showMessageDialog()` can be called using the following combinations of parameters:

Component, Object
Component, Object, String, int
Component, Object, String, int, Icon

Copy

1. Component – The first parameter is a component which determines the Frame in which the dialog is displayed; if null, or if the `parentComponent` has no Frame, a default Frame is used.
2. Object – The second parameter can be any objects. *(In some older versions of Java you might get a compiler error when using primitive types directly).*
3. String – The third parameter is a String placed as the title of the message dialog window.
4. int – The int that follows the String is the `MessageType` . The different `MessageTypes` for `JOptionPane` , are:
 - `ERROR_MESSAGE`
 - `INFORMATION_MESSAGE`
 - `WARNING_MESSAGE`
 - `QUESTION_MESSAGE`
 - `PLAIN_MESSAGE`
5. Icon – The last parameter is an `Icon` that is displayed inside the dialog and overrides the default `MessageType` icon.

1. Component & Object

The simplest way to use the message dialog. Example with `Component` set to null and a `String` as second argument:

SimpleDialog1.java

```
package com.mkyong.messageDialog;

import javax.swing.JOptionPane;

public class SimpleDialog1 {

    public static void main(String[] args){
        JOptionPane.showMessageDialog(null, "Simple Information Message");
    }

}
```

[Copy](#)

Output:

swing-dialog-1a

2. Component, Object, String & int

Adding some more information to the message dialog. Example with **Component** set to null and a **double** as second parameter:

SimpleDialog2a.java

```
package com.mkyong.messageDialog;

import javax.swing.JOptionPane;

public class SimpleDialog2a {

    public static void main(String[] args){
        JOptionPane.showMessageDialog(null, 8.9, "This is not an integer.",
        JOptionPane.PLAIN_MESSAGE);
    }

}
```

[Copy](#)

Output:

Example of an Error Message (**Component** set to null, **String Object**):

swing-dialog-1b

SimpleDialog2b.java

```
package com.mkyong.messageDialog;

import javax.swing.JOptionPane;

public class SimpleDialog2b {

    public static void main(String[] args){
        JOptionPane.showMessageDialog(null, "Uh-oh!", "Error",
JOptionPane.ERROR_MESSAGE);
    }

}
Copy
```

Output:

swing-dialog-1c

3. Component, Object, String, int & Icon

Make your message dialog “prettier”. Example with **Icon** retrieved from a directory:

SimpleDialog3a.java

```
package messageDialog;

import javax.swing.ImageIcon;
import javax.swing.JOptionPane;

public class SimpleDialog3a {

    public static void main(String[] args){
        ImageIcon icon = new ImageIcon("src/images/turtle64.png");
        JOptionPane.showMessageDialog(null, "I like turtles.",
            "Customized Dialog", JOptionPane.INFORMATION_MESSAGE, icon);
    }

}
Copy
```

Output:

Example with **Component** set to a **frame** :

swing-dialog-1d

MessageDialogInFrame.java

```

package com.mkyong.messageDialog;

import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import java.awt.Color;

public class MessageDialogInFrame extends JFrame{


    public MessageDialogInFrame() {
        getContentPane().setBackground(Color.DARK_GRAY);
        setTitle("Message Dialog in Frame");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
        setResizable(false);
        setSize(400, 300);
        getContentPane().setLayout(null);
    }

    public static void main(String[] args){
        ImageIcon icon = new ImageIcon("src/images/turtle64.png");
        JOptionPane.showMessageDialog(new MessageDialogInFrame(),
            "I appear as part of the frame!!", "Customized Dialog",
            JOptionPane.INFORMATION_MESSAGE, icon);
    }
}

```

[Copy](#)

Output:



4. A more advanced example

For this example we are passing a `JPanel` as the Object parameter. The `JPanel` is customized and has a `JLabel` added to it. We are also manipulating the size of the `OptionPane` using a call to `UIManager`.

MessageDialogPanel.java

```

package com.mkyong.messageDialog;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import javax.swing.ImageIcon;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;

public class MessageDialogPanel {

    public static void main(String[] args){
        ImageIcon icon = new ImageIcon("src/images/turtle64.png");

        JPanel panel = new JPanel();
        panel.setBackground(new Color(102, 205, 170));
        panel.setSize(new Dimension(200, 64));
        panel.setLayout(null);

        JLabel label = new JLabel("Turtles are awesome!!! :D");
        label.setBounds(0, 0, 200, 64);
        label.setFont(new Font("Arial", Font.BOLD, 11));
        label.setHorizontalAlignment(SwingConstants.CENTER);
        panel.add(label);

        UIManager.put("OptionPane.minimumSize",new Dimension(300, 120));
        JOptionPane.showMessageDialog(null, panel, "Customized Message Dialog",
JOptionPane.PLAIN_MESSAGE, icon);
    }
}

```

[Copy](#)

swing-dialog-1f