

1. 概述

官方JavaDocsApi: [javax.swing.JToggleButton](https://docs.oracle.com/javase/8/docs/api/javax.swing/JToggleButton.html)

JToggleButton, 开关按钮。JToggleButton 是 JRadioButton, JCheckBox 的父类, 主要实现一个按钮的两种状态 (选中 和 未选中) 来实现开关切换的效果。

JToggleButton 常用构造方法:

```
1 // 无文本, 默认未选中
2 JToggleButton()
3 // 有文本, 默认未选中
4 JToggleButton(String text)
5 // 有文本, 并指定是否选中
6 JToggleButton(String text, boolean selected)
```

JToggleButton 常用方法:

```
1 // 设置开关按钮的 文本、字体 和 字体颜色
2 void setText(String text)
3 void setFont(Font font)
4 void setForeground(Color fg)
5 /* 以下方法定义在 javax.swing.AbstractButton 基类中 */
6 // 设置开关按钮是否选中状态
7 void setSelected(boolean b)
8 // 判断开关按钮是否选中
9 boolean isSelected()
10 // 设置开关按钮是否可用
11 void setEnabled(boolean enable)
12 // 设置开关按钮在 默认(关)、被选中(开)、不可用 时显示的图片
13 void setIcon(Icon defaultIcon)
14 void setPressedIcon(Icon pressedIcon)
15 void setDisabledIcon(Icon disabledIcon)
16 // 设置图片和文本之间的间距
17 void setIconTextGap(int iconTextGap)
```

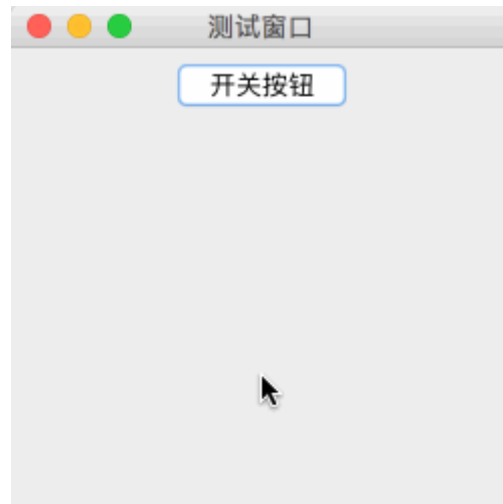
JToggleButton 常用监听器:

```
1 // 添加状态改变监听器
2 void addChangeListener(ChangeListener l)
```

2. 代码实例: 默认的开关按钮

```
1 package com.xiets.swing;
2 import javax.swing.*;
3 import javax.swing.event.ChangeEvent;
4 import javax.swing.event.ChangeListener;
5 public class Main{
6     public static void main(String[] args){
7         JFrame jf =newJFrame("测试窗口");
8         jf.setSize(250,250);
9         jf.setLocationRelativeTo(null);
10        jf.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
11        JPanel panel =new JPanel();
12        // 创建开关按钮
13        JToggleButton toggleBtn =new JToggleButton("开关按钮");
14        // 添加 toggleBtn 的状态被改变的监听
15        toggleBtn.addChangeListener(new ChangeListener(){
16            @Override
17            public void stateChanged(ChangeEvent e){
18                // 获取事件源（即开关按钮本身）
19                JToggleButton toggleBtn =(JToggleButton) e.getSource();
20                System.out.println(toggleBtn.getText()+" 是否选中: "+ toggleBtn.isSelected());
21            }
22        });
23        panel.add(toggleBtn);
24        jf.setContentPane(panel);
25        jf.setVisible(true);
26    }
27 }
```

结果展示:



result_01.gif

运行代码，点击开关按钮，查看控制台输出。

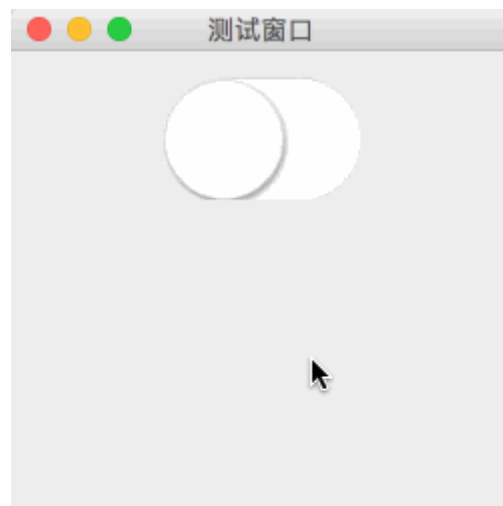
3. 代码实例: 自定义图片开关

使用下面两个图片分别代表开关按钮的 选中(开) 和 未选中(关) 状态，分别命名为: **toggle_on.png**, **toggle_off.png**:

```
1 package com.xiets.swing;
2 import javax.swing.*;
3 import javax.swing.event.ChangeEvent;
4 import javax.swing.event.ChangeListener;
5 public class Main{
6     public static void main(String[] args){
7         JFrame jf =newJFrame("测试窗口");
8         jf.setSize(250,250);
9         jf.setLocationRelativeTo(null);
10        jf.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
11        JPanel panel =new JPanel();
12        // 创建开关按钮
13        JToggleButton toggleBtn =new JToggleButton();
14        // 首先设置不绘制按钮边框
15        toggleBtn.setBorderPainted(false);
16        // 设置 选中(开) 和 未选中(关) 时显示的图片
17        toggleBtn.setSelectedIcon(new ImageIcon("toggle_on.png"));
18        toggleBtn.setIcon(new ImageIcon("toggle_off.png"));
19        // 添加 toggleBtn 的状态被改变的监听
20        toggleBtn.addChangeListener(
21            new ChangeListener(){
22                @Override
23                public void stateChanged(ChangeEvent e){
```

```
24 // 获取事件源（即开关按钮本身）
25 JToggleButton toggleBtn =(JToggleButton) e.getSource();
26 System.out.println(toggleBtn.getText()+" 是否选中: "+ toggleBtn.isSelected());
27 }
28 });
29 panel.add(toggleBtn);
30 jf.setContentPane(panel);
31 jf.setVisible(true);
32 }
33 }
```

结果展示:



result_02.gif

运行代码，点击开关按钮，查看控制台输出。