

[Prev \(http://ateraimemo.com/Swing/ButtonsInMenuItem.html\)](http://ateraimemo.com/Swing/ButtonsInMenuItem.html)

[Swing \(http://ateraimemo.com/Swing.html\)](http://ateraimemo.com/Swing.html) »

[ButtonWidth \(http://ateraimemo.com/Swing/ButtonWidth.html\)](http://ateraimemo.com/Swing/ButtonWidth.html)

[Next \(http://ateraimemo.com/Swing/CardLayoutTabbedPane.html\)](http://ateraimemo.com/Swing/CardLayoutTabbedPane.html)

JButtonなどの高さを変更せずに幅を指定

[JButton \(/tags.html#JButton-ref\)](#) [LayoutManager \(/tags.html#LayoutManager-ref\)](#)

[BoxLayout \(/tags.html#BoxLayout-ref\)](#) [SpringLayout \(/tags.html#SpringLayout-ref\)](#)

[GridLayout \(/tags.html#GridLayout-ref\)](#)

[en \(/java-swing-tips.blogspot.com/2011/10/fixed-width-jbuttons-in-bottom-right.html\)](#)

Total: 22429 , Today: 1 , Yesterday: 5

Posted by [aterai \(http://ateraimemo.com/Users/aterai.html\)](http://ateraimemo.com/Users/aterai.html) at [2004-11-29 \(http://ateraimemo.com/?cmd=backup&page=Swing/ButtonWidth\)](http://ateraimemo.com/?cmd=backup&page=Swing/ButtonWidth)

Last-modified: [2016-09-01 21:31 \(http://ateraimemo.com/?cmd=diff&page=Swing/ButtonWidth\)](http://ateraimemo.com/?cmd=diff&page=Swing/ButtonWidth)

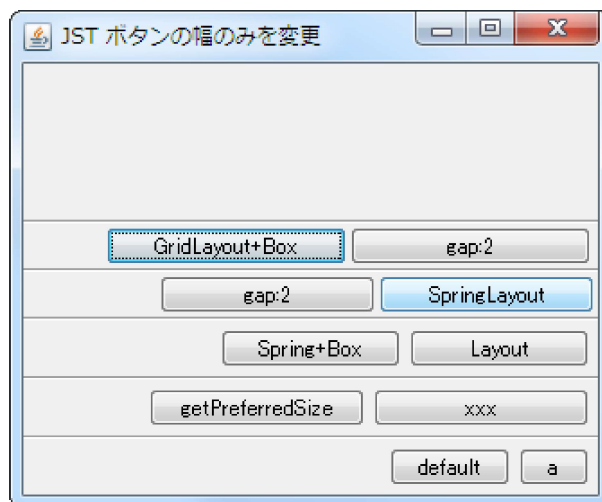
概要

高さはデフォルトのまま幅だけを指定して、JButton、JComboBox、JTextFieldなどのサイズを変更します。

[Runnable JARファイル example.jar \(http://ateraimemo.com/swing/buttonwidth/example.jar\)](http://ateraimemo.com/swing/buttonwidth/example.jar)

[ソースコード src.zip \(http://ateraimemo.com/swing/buttonwidth/src.zip\)](http://ateraimemo.com/swing/buttonwidth/src.zip)

[リポジトリ repository \(/github.com/aterai/java-swing-tips/tree/master/ButtonWidth\)](https://github.com/aterai/java-swing-tips/tree/master/ButtonWidth)



(https://lh5.googleusercontent.com/-B3A8vHPu9_IUmY7hTtdmt/AAAAAAAAAB4s/7NknsHc_vw/s800/ButtonWidth.png)

サンプルコード

[view all \(//github.com/aterai/java-swing-tips/blob/master/ButtonWidth/src/java/example/MainPanel.java\)](https://github.com/aterai/java-swing-tips/blob/master/ButtonWidth/src/java/example/MainPanel.java)

```
private static JComponent createRightAlignButtonBox2(
    final List<JButton> list, final int buttonWidth, int gap) {
    JComponent box = new JPanel() {
        @Override public void updateUI() {
            for (JButton b : list) {
                b.setPreferredSize(null);
            }
            super.updateUI();
            EventQueue.invokeLater(new Runnable() {
                @Override public void run() {
                    int maxHeight = 0;
                    for (JButton b : list) {
                        maxHeight = Math.max(maxHeight, b.getPreferredSize().height);
                    }
                    Dimension d = new Dimension(buttonWidth, maxHeight);
                    for (JButton b : list) {
                        b.setPreferredSize(d);
                    }
                    revalidate();
                }
            });
        }
    };
    box.setLayout(new BoxLayout(box, BoxLayout.X_AXIS));
    box.add(Box.createHorizontalGlue());
    for (JButton b : list) {
        box.add(b);
        box.add(Box.createHorizontalStrut(gap));
    }
    box.setBorder(BorderFactory.createEmptyBorder(gap, 0, gap, 0));
    return box;
}
```

解説

上記のサンプルでは、JButton の高さはデフォルト、幅をその文字列によらずに一定、配置は右寄せで水平にした場合のレイアウト方法をテストしています。

- Default
 - BoxLayout で Box.createHorizontalGlue() を使用して右寄せ
 - JButton の幅は、その文字列の長さに依存

- `getPreferredSize`
 - `Box` を使用して右寄せ
 - `JButton` の幅は、`JButton#setPreferredSize(...)` で固定幅を設定
 - `JButton` の高さは、`JButton#getPreferredSize()` で取得したサイズの高さを使用
 - `LookAndFeel` を変更すると `JButton` の高さが変化するので、その場合は `JButton#updateUI()` をオーバーライドして `JButton#setPreferredSize(...)` を使用する必要がある
- `SpringLayout + Box`
 - [SpringLayoutの使用 \(http://ateraimemo.com/Swing/SpringLayout.html\)](http://ateraimemo.com/Swing/SpringLayout.html)
 - `SpringLayout` で幅指定、`BoxLayout` で右寄せ
 - `SpringLayout.Constraints` で `JButton` の固定幅を指定
 - 親パネルの幅も `SpringLayout.Constraints` で固定し、`Box` で入れ子にして右寄せ
 - `LookAndFeel` を変更すると `JButton` の高さが変化するので、その場合は `JComponent#getPreferredSize(...)` をオーバーライドして親パネルの高さを更新する必要がある
- `SpringLayout`
 - [SpringLayoutの使用 \(http://ateraimemo.com/Swing/SpringLayout.html\)](http://ateraimemo.com/Swing/SpringLayout.html)
 - `SpringLayout` を使用して右寄せ
 - `SpringLayout.Constraints` で `JButton` の幅を固定
 - `LookAndFeel` を変更すると `JButton` の高さが変化するので、その場合は `JComponent#getPreferredSize(...)` をオーバーライドして親パネルの高さを更新する必要がある
- `GridLayout + Box`
 - `GridLayout` で幅指定、`BoxLayout` で右寄せ
 - `GridLayout` ですべての `JButton` のサイズを同じにする
 - `Box` で入れ子にして右寄せ
 - 幅が最大の `JButton` のテキストが変更されると、親の `Box` の幅が変化してしまう

以下、`setPreferredSize()` を使用する際の補足、注意点です。

`JButton` の UI がフォントサイズや文字列の長さから決めたデフォルトサイズを `getPreferredSize()` で取得しています。高さはそのまま利用し、幅だけ一定の値を設定して、新たなデフォルトサイズを作成し、`setPreferredSize()` しています。これで次から `getPreferredSize()` で返ってくる値は、どちらのボタンでも全く同じになります。

この `setPreferredSize(...)` で設定したサイズは、`LookAndFeel` を変更しても残ってしまう (`DimensionUIResource` を使用しても効果がない)ため、`LookAndFeel` がデフォルトで使用する高さを取得したい場合は、`updateUI()` をオーバーライドして、一旦 `JButton#setPreferredSize(null);` を設定して幅を変更した推奨サイズをクリアしてから、`super.updateUI();` で推奨サイズを更新し、その後で固定幅、`LookAndFeel` デフォルトの高さを `setPreferredSize(...)` で設定し直す必要があります。

`getPreferredSize()` で得られる値を使用するかどうかは、レイアウトマネージャーによって異なりますが、水平方向にコンポーネントを並べる `BoxLayout` の場合は以下のようになるため、パネルをリサイズしても、ボタンのサイズはどちらも同じで変化しません。

- 幅: 推奨サイズ(`getPreferredSize` メソッド)から取得した値
- 高さ: 各コンポーネントの推奨サイズ(`getPreferredSize` メソッド)で得られた中からもっとも大きな値
 - 上記のサンプルで `button1.setPreferredSize(new Dimension(100, 0));` としても結果は同じ

例えば、各ボタンを格納した親フレームを `pack()` する前に `JButton#getPreferredSize()` ではなく、`JButton#getSize()` でサイズを取得すると、`[width=0,height=0]` が帰ってきます。コンポーネントが表示されている場合、`getSize()` で得られるサイズは、その実際に表示されているサイズ(レイアウトマネージャーが決める)になります。

以下は `JLabel` を `getPreferredSize()` した場合の例です。初期状態(`preferredSize` が `null`)の場合は、`JLabel` の `UI` がサイズを計算しています。

```
JLabel l = new JLabel();
l.setText("a"); //preferredSizeがnullの場合、UIがサイズを計算
//l.getPreferredSize() -> Dimension[width=6, height=13]

l.setText("aaaa"); //JLabelの場合、Fontサイズと文字列の長さなどで決まる
//l.getPreferredSize() -> Dimension[width=24, height=13]

l.setText("<html>aa<br>aa");
//l.getPreferredSize() -> Dimension[width=12, height=26]

l.setPreferredSize(new Dimension(10, 10)); //preferredSizeを設定した場合
//l.getPreferredSize() -> Dimension[width=10, height=10]

l.setPreferredSize(null); //preferredSizeをnullに戻した場合
//l.getPreferredSize() -> Dimension[width=12, height=26]
```

参考リンク

- [JComboBoxなどの幅をカラム数で指定 \(http://ateraimemo.com/Swing/SetColumns.html\)](http://ateraimemo.com/Swing/SetColumns.html)

- JComboBox 、 JTextField などのコンポーネントでは、カラム数で幅を指定することが可能
 - JDK 1.5.0 : カラム数で幅を指定すると、コンポーネントによってサイズや余白などが微妙に異なる
 - JDK 1.6.0 以上: LookAndFeel が同じなら、カラム数での幅指定で、どのコンポーネントでもほぼ同じサイズになる
- [SpringLayoutの使用 \(http://ateraimemo.com/Swing/SpringLayout.html\)](http://ateraimemo.com/Swing/SpringLayout.html)
- [Santhosh Kumar's Weblog \(http://www.jroller.com/santhosh/entry/how_do_you_layout_command\)](http://www.jroller.com/santhosh/entry/how_do_you_layout_command)
 - 専用のレイアウトマネージャーを作成するサンプル
- [JOptionPaneで使用するボタンのサイズを揃える \(http://ateraimemo.com/Swing/SameSizeButtons.html\)](http://ateraimemo.com/Swing/SameSizeButtons.html)

コメント

- SpringLayout などを使って幅を固定する方法を追加(更新日時は忘れました)。スクリーンショットは未更新。 -- aterai 2013-04-14 (日) 00:36:03
- GridLayout + Box のサンプルを追加、 LookAndFeel を実行中に変更する場合の注意点を追加、スクリーンショットを更新。 -- aterai 2013-10-22 (火) 17:33:58

0 件のコメント Java Swing Tips

 ログイン ▼

♥ オススメする  共有

評価順に並び替え ▼



コメントを投稿する...

最初のコメントをしましょう！

JAVA SWING TIPS の関連記事

JLayerを使ってJProgressBarの色相を変更する

コメント1件 • 2年前 •



TERAI Atsuhiko — 再描画が無限ループするバグを修正。

Java Swing Tips

コメント1件 • 2年前 •



TERAI Atsuhiko — 2017-02-01でlivefyreのコメントが使用できなくなるらしいので、disqusに移行。

Fontの一覧を取得する

2 件のコメント • 3ヶ月前 •



TERAI Atsuhiko — Sorry for the late reply.I used to be interested in JavaFX before, but unfortunately I didn't track that trend recently.

JTreeの余白をドラッグしてスクロール

5 件のコメント • 2年前 •



Ttny Orange — ありがとうございます。(^^)

 更新を受取る  あなたのサイトにDisqusを追加 Disqusを追加 追加  非公開