

تقديم به همه هموطنان عزیزم

آموزش زبان برنامه نویسی جاوا

گرافیک در جاوا - پکیج Swing

جلسه بیست و دوم

کلاس JLayeredPane

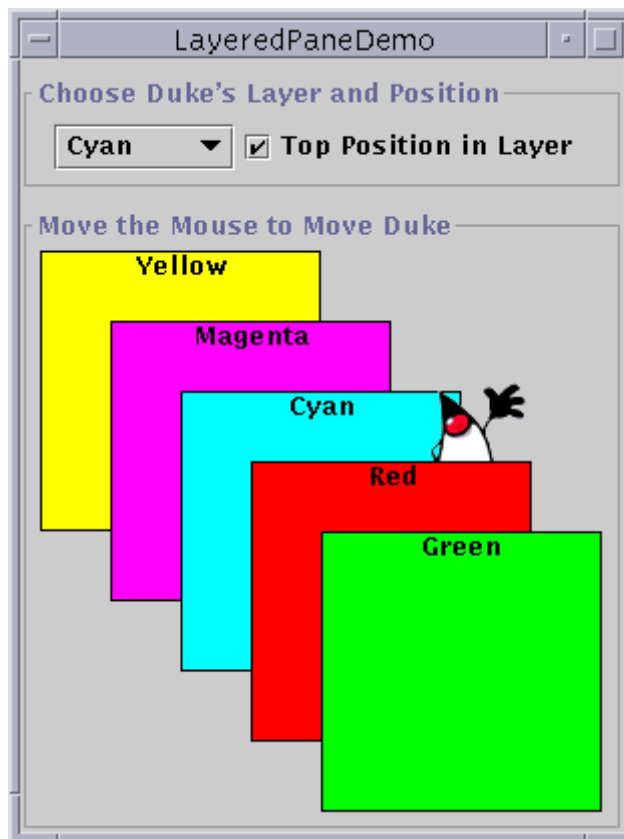
نویسنده: رحمان زارعی

جاوا را ساده، آسان و شیرین بنوشید!!!!



این جلسه آموزشی رایگان است، فروش و ویرایش آن ممنوع و حرام می باشد. اما این کتاب را می توانید همین جور که هست در سایت و شبکه اجتماعی خود به اشتراک بگذارید.

کلاس `JLayeredPane` برای عمق دادن به اجزای گرافیکی در جاوا استفاده می شود. این کلاس برای دادن بُعد سوم به موقعیت قرارگیری اجزای گرافیکی در برنامه استفاده می شود. به بیان دیگر با هم پوشانی و روی هم قرار گرفتن اجزای گرافیکی، عمق دار می شوند. برای درک بهتر از شکل ظاهری کلاس `JLayeredPane` تصویر (۱) را مشاهده کنید:



تصویر (۱)

- همان طور که در تصویر (۱) مشاهده می کنید اجزای گرافیکی با روی هم قرار گرفتن عمق ایجاد می کنند. این کار توسط کلاس `JLayeredPane` انجام می شود.

## • سازنده های پر کاربرد کلاس JPanel :

سازنده	کاربرد
JLayeredPane()	ایجاد یک JPanel جدید

## • متدهای پر کاربرد کلاس JPanel :

متد	کاربرد
int indexOf(Component c)	ایندکس component یا اجزای گرافیکی مشخص را برمیگرداند.
int getLayer(Component c)	ویژگی لایه برای component مشخص را برمیگرداند.
int getPosition(Component c)	موقعیت نسبی component موجود در لایه را برمیگرداند.

- مثال همیشه بهتر از توضیح است، دستورات درون مثال ها رو خوب ببینید.

```
package jLayeredPaneEx;

import javax.swing.*;
import java.awt.*;

public class LayeredPaneExample extends JFrame {
    public LayeredPaneExample() {
        super("LayeredPane Example");
        setSize(400, 400);
        JPanel pane = getLayeredPane();
        // creating buttons
        JButton top = new JButton();
        top.setBackground(Color.white);
        top.setBounds(20, 20, 50, 50);
        JButton middle = new JButton();
        middle.setBackground(Color.red);
        middle.setBounds(40, 40, 50, 50);
        JButton bottom = new JButton();
        bottom.setBackground(Color.cyan);
    }
}
```

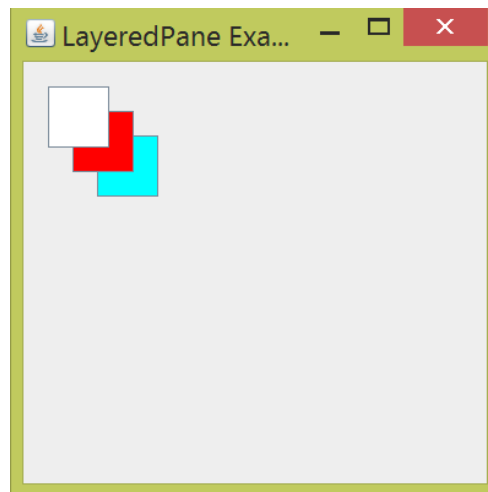
```

        bottom.setBounds(60, 60, 50, 50);
        // adding buttons on pane
        pane.add(bottom, new Integer(1));
        pane.add(middle, new Integer(2));
        pane.add(top, new Integer(3));
        setLayout(null);
        setVisible(true);
    }

    public static void main(String[] args) {
        LayeredPaneExample panel = new LayeredPaneExample();
    }
}

```

خروجی: تصویر (۲)



تصویر (۲)

توضیحات:

```
JLayeredPane pane = getLayeredPane();
```

- با شی ساختن از کلاس JLayeredPane یک لایه ایجاد می کنیم.

```

JButton top = new JButton();
top.setBackground(Color.white);
top.setBounds(20, 20, 50, 50);
JButton middle = new JButton();
middle.setBackground(Color.red);
middle.setBounds(40, 40, 50, 50);
JButton bottom = new JButton();

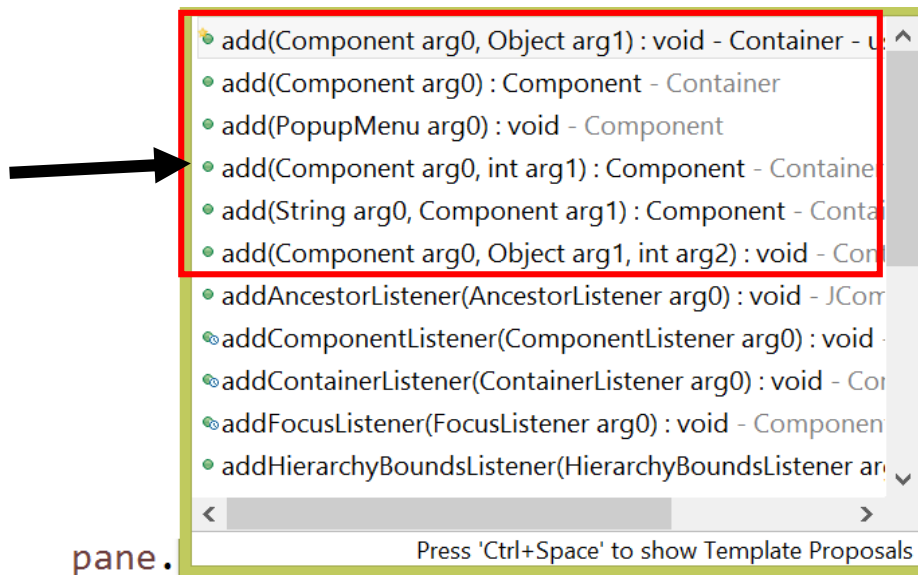
```

```
bottom.setBackground(Color.cyan);
bottom.setBounds(60, 60, 50, 50);
```

- برای برنامه خود چند دکمه button ایجاد کرده و مختصات، ابعاد و رنگ آن ها مشخص کرده ایم.

```
pane.add(bottom, new Integer(1));
```

- با متد add دکمه با نام bottom را به لایه خود اضافه کرده ایم.
- متد add دو پارامتر به عنوان ورودی میگیرد، پارامتر اول از نوع کلاس Component و پارامتر دوم یک عدد صحیح int میگیرد. البته این متد پارامترهای متفاوتی میگیرد که در تصویر (۳) می توانید آنها را مشاهده کنید:



تصویر (۳)

- پارامتر از نوع int متد add برای تعیین موقعیت قرارگیری component (اجزای گرافیکی) ما در لایه استفاده می شود.
- سایر دستورات را در جلسات گذشته بررسی کرده ایم.

پیروز و موفق باشید

سایت آموزش زبان جاوا به زبان ساده، آسان و شیرین!!!

www.JAVAPro.ir

آموزش جاوا SE را با تجربه شخصی و به زبان خودمونی یاد بگیرید!!!!

# بازدید از کانال

# بازدید از سایت

هر روز مفاهیم و مثال های جدید به سایت اضافه می شود برای اطلاع از مطالب جدید روی سایت عضو کانال شوید.

دخل و تصرف ، ویرایش و کپی زدن تمامی آموزش های جاوا لایک به دور از اخلاق حرفه ای ست و حرام می باشد.