

آموزش زبان برنامه نویسی جاوا

گرافیک در جاوا - پکیج Swing

جلسه هفدهم

کلاس JColorChooser

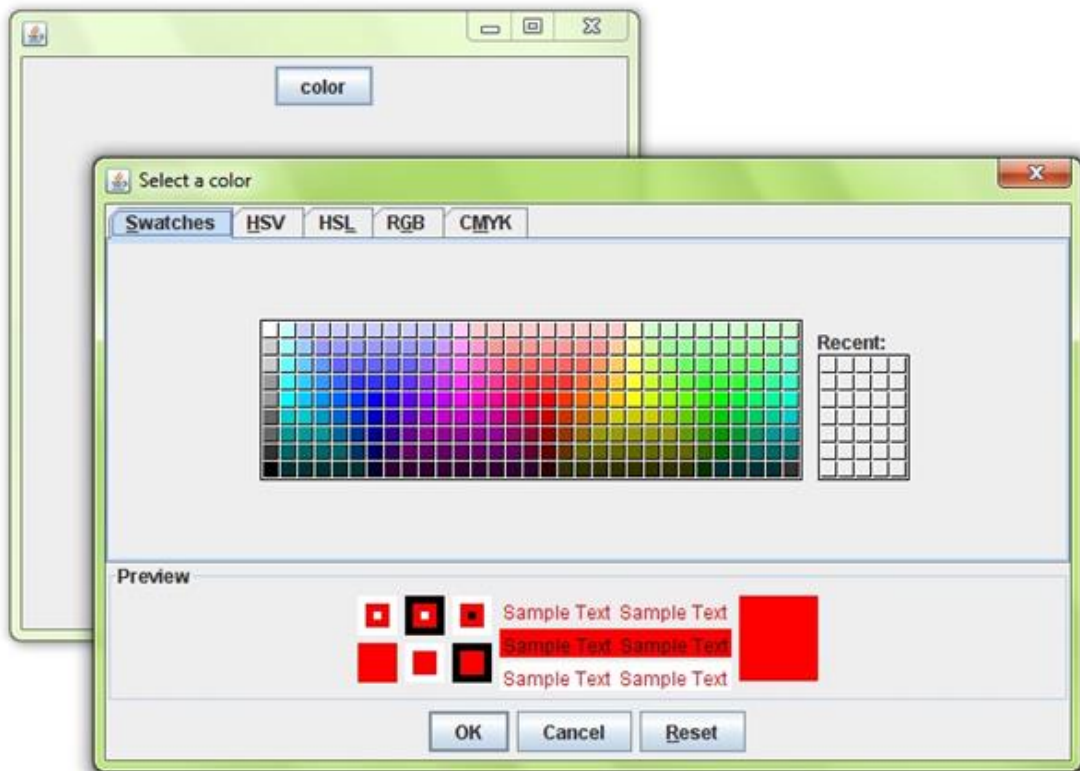
نویسنده: رحمان زارعی

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 JColorChooser همان طور که از اسمش پیداست برای ایجاد یک جعبه رنگ استفاده می شود، کاربر می تواند از میان رنگ ها، رنگ دلخواه خود را انتخاب کند. برای درک بهتر نسبت به این اجزای گرافیکی در جاوا تصویر (۱) را مشاهده کنید:

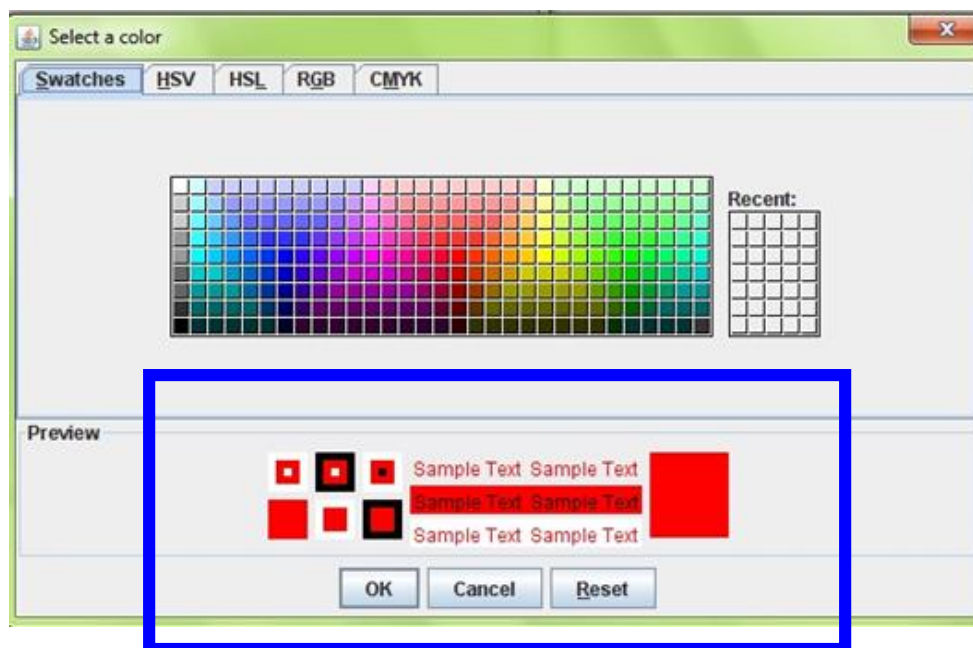


تصویر (۱)

• سازنده های پد کاربرد کلاس JColorChooser :

سازنده	کاربرد
JColorChooser()	ایجاد یک ColorChooser که رنگ پیشفرض انتخاب شده آن سفید می باشد.
JColorChooser(color initialcolor)	ایجاد یک ColorChooser و تعیین رنگ پیشفرض انتخاب شده

- منظور از رنگ پیشفرض انتخاب شده ، رنگی است هنگام اجرای برنامه بین رنگهای موجود در ColorChooser بصورت پیشفرض انتخاب شده است. مثلا در تصویر(۲) رنگ پیشفرض انتخاب شده قرمز می باشد.



تصویر(۲)

• متدهای پر کاربرد کلاس ColorChooser:

void addChooserPanel(AbstractColorChooserPanel panel)	متد
این متد برای افزودن یک رنگ به مجموعه رنگ های موجود در یک ColorChooser کاربرد	
static Color showDialog(Component c, String title, Color initialColor)	متد
این متد برای نمایش جعبه رنگ های موجود در ColorChooser کاربرد دارد. به جای پارامتر C یک شی از اجزای گرافیکی نظیر frame می گذاریم. به جای پارامتر title متن قرار می دهیم که نقش عنوان ColorChooser را بازی خواهد کرد. به جای پارامتر initialColor یک شی از نوع Color می گذاریم که رنگ انتخاب شده پیشفرض ما را تعیین میکند.	

مثال:

```
package javalike;

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

public class ColorChooserExample extends JFrame implements ActionListener {
    JButton b;

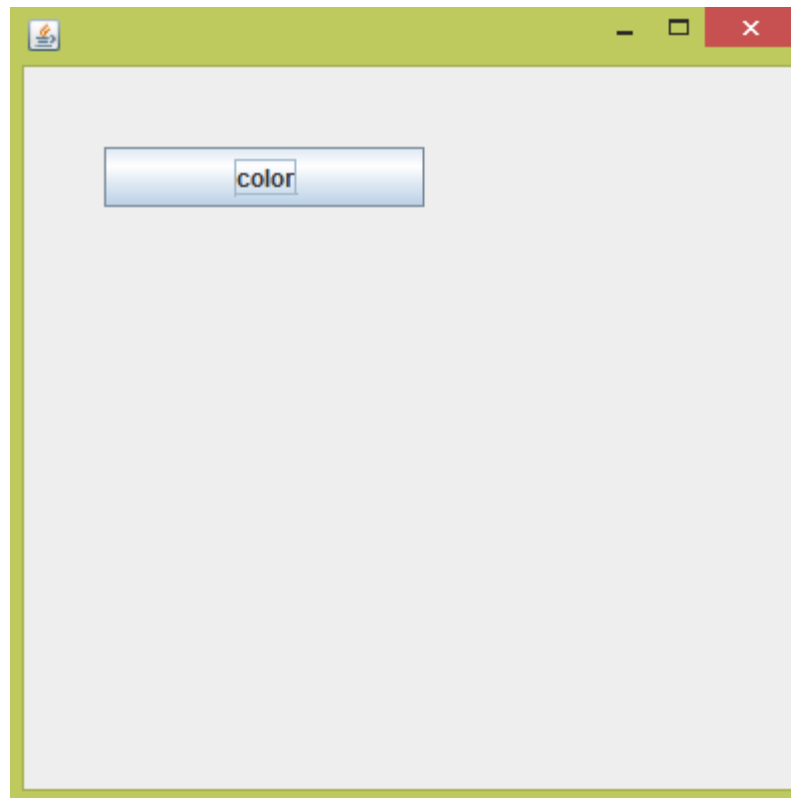
    ColorChooserExample() {
        b = new JButton("color");
        b.addActionListener(this);
        b.setBounds(40, 40, 160, 30);
        add(b);
        setLayout(null);
        setSize(400, 400);
        setVisible(true);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }

    public void actionPerformed(ActionEvent e) {
        Color initialcolor = Color.YELLOW;
    }
}
```

```
Color color = JColorChooser.showDialog(this, "Select a color",
                                     initialcolor);
    b.setBackground(color);
}

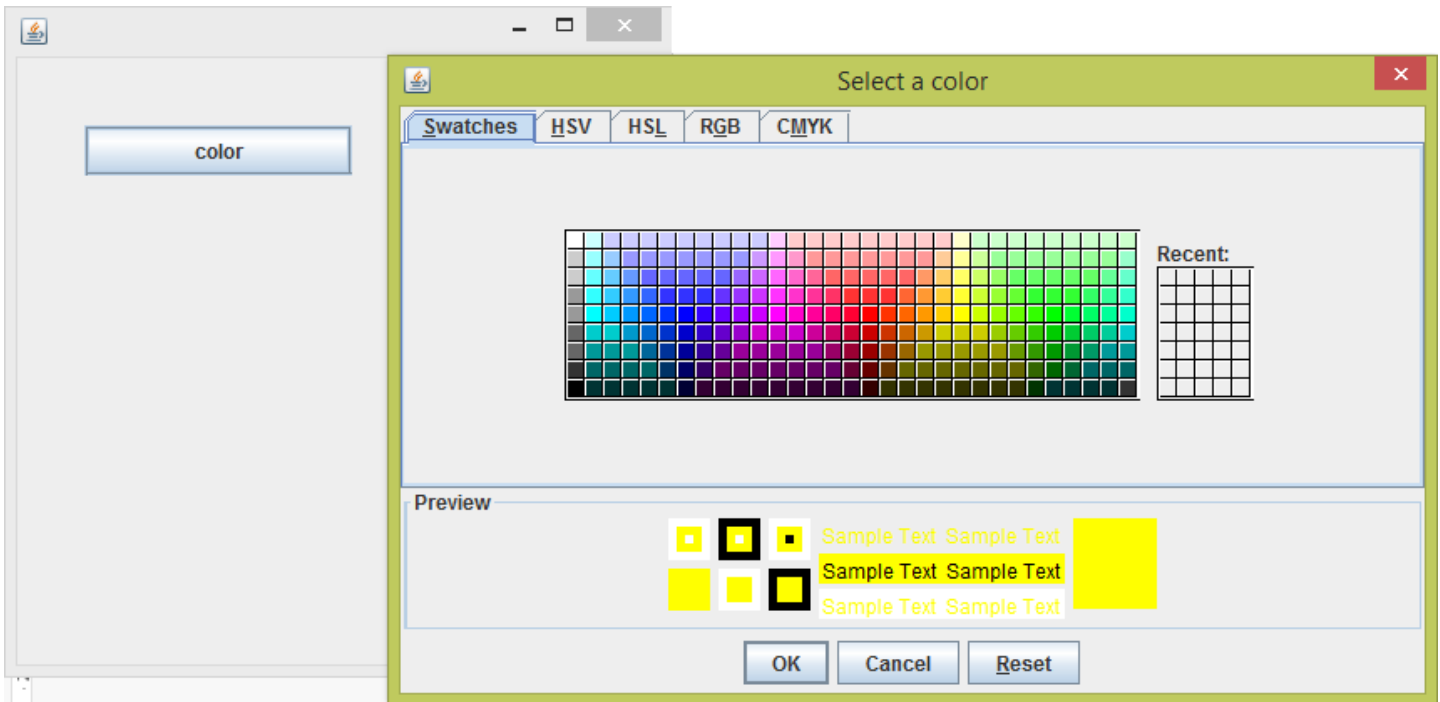
public static void main(String[] args) {
    ColorChooserExample ch = new ColorChooserExample();
}
}
```

خروجی: ابتدا هنگام اجرای برنامه خروجی بصورت تصویر (۳) می باشد:



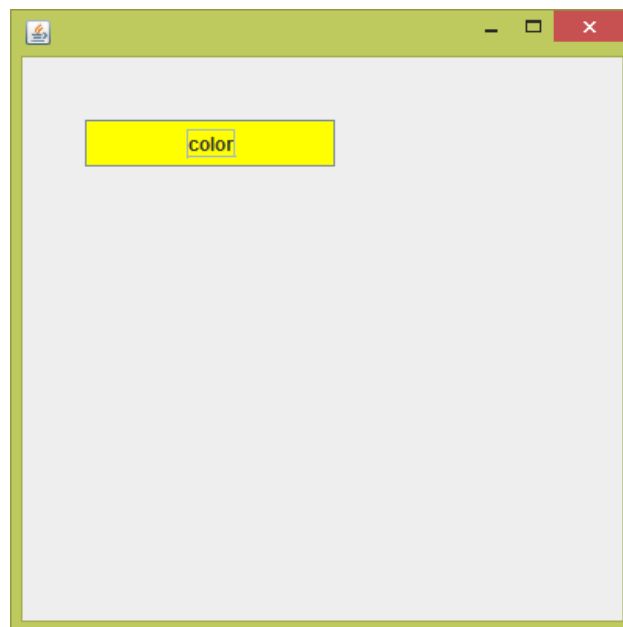
تصویر (۳)

- بعد از کلیک کردن روی دکمه **color** خروجی بصورت تصویر (۴) خواهد شد:



تصویر (۴)

- همان طور که در تصویر (۴) مشاهده می کنید یک جعبه رنگی نمایش داده شده که رنگ انتخاب شده پیشفرض آن زرد می باشد. این جعبه رنگی از بخش های مختلفی برای انتخاب رنگ تشکیل شده است. فرض کنید رنگ مورد نظر ما همین زرد است ، حال اگر دکمه **OK** را فشار دهیم خروجی بصورت تصویر (۵) خواهد شد:



تصویر (۵)

- خوب با زدن دکمه ok رنگ دکمه button ما به زرد تغییر می کند.

توضیحات:

```

1. b = new JButton("color");
2. b.addActionListener(this);
3. b.setBounds(40, 40, 160, 30);
4. add(b);
5. setLayout(null);
6. setSize(400, 400);
7. setVisible(true);
8. setDefaultCloseOperation(EXIT_ON_CLOSE);

```

۱. ایجاد یک دکمه button با عنوان color

۲. در جلسات قبل در مورد این متد زیاد صحبت کردیم، برای افزودن یک ActionListener به دکمه button استفاده می شود.

۳. تعیین مختصات و ابعاد دکمه button

۴. افزودن دکمه button به فریم

۵. چون از طرح بندی خاصی استفاده نکردیم مقدار این متد را null قرار داده ایم.

۶. تعیین اندازه فریم

۷. برای نمایش فریم و تمام اجزای گرافیکی این متد را صدا زده ایم.

۸. این دستور را نیز در جلسات گذشته بررسی کردیم، این دستور ثابت برای بستن برنامه با دکمه close یا همان دکمه ضربدر قرمز رنگ گوشه فریم. اگر از این دستور استفاده نکنیم با زدن دکمه ضربدر قرمز رنگ برنامه به طور کامل بسته نمی شود.

```

public void actionPerformed(ActionEvent e) {
    Color initialcolor = Color.YELLOW;
    Color color = JColorChooser.showDialog(this, "Select a color",
    initialcolor);
    b.setBackground(color);
}

```

- متد actionPerformed درون اینترفیس ActionListener قرار دارد. وظیفه این متد دریافت رویدادهای مربوط به فشردن یا کلیک کردن روی اجزای گرافیکی نظیر button ها، menu ها و... می باشد. وقتی رویدادی رخ میدهد این متد صدا زده می شود و دستورات درون آن اجرا می شود.

```
Color initialcolor = Color.YELLOW;
```

- ایجاد یک شی از کلاس `Color` و مقداردهی اولیه آن. رنگ آن را زرد (`yellow`) قرار داده ایم.
- شی که از نوع کلاس `Color` تعریف می شود رنگ ها را در خود ذخیره می کند و برای رنگی کردن اجزای گرافیکی و... استفاده می شود.

```
Color color = JColorChooser.showDialog(this, "Select a color",
```

- با استفاده از متد `showDialog` جعبه رنگ ما نمایش داده می شود.
- متد `showDialog` یک متد استاتیک درون کلاس `JColorChooser` است که با کلاس `JColorChooser` آن را صدا زده ایم.
- جعبه رنگ نمایش داده شده ، عنوان آن "Select a color" و رنگ پیش فرض انتخاب شده آن شی `initialcolor` که از نوع کلاس `Color` هستش و مقدار درون آن رنگ `yellow` می باشد.
- کلمه کلیدی `this` به شی از کلاس `JFrame` که ارث برده است اشاره دارد. چون کلاس ما کلاس `JFrame` را به ارث برده است ، پس فرزند `JFrame` حساب میشه و به تمام ویژگی ها و رفتارهای `JFrame` دسترسی پیدا می کند. در نتیجه شی از کلاس `JFrame` رو می توانیم در پارامتر اول متد `showDialog` که اشیای اجزای گرافیکی حق دارند قرار بگیرند، جایگزین کنیم.
- ما در جعبه رنگی که نمایش داده شده رنگ مورد نظر را انتخاب می کنیم و با زدن دکمه `ok` متد `showDialog` مقدار رنگ انتخاب شده را برای ما برمیگرداند ، که در شی `color` از نوع کلاس `Color` هستش ذخیره می کنیم.

مثال:

```
package javalike;

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

public class ColorChooserExample extends JFrame implements ActionListener {
    JFrame f;
    JButton b;
    JTextArea ta;

    ColorChooserExample() {
        f = new JFrame("Color Chooser Example.");
        b = new JButton("Pad Color");
        b.setBounds(200, 250, 100, 30);
        ta = new JTextArea();
    }
}
```



```

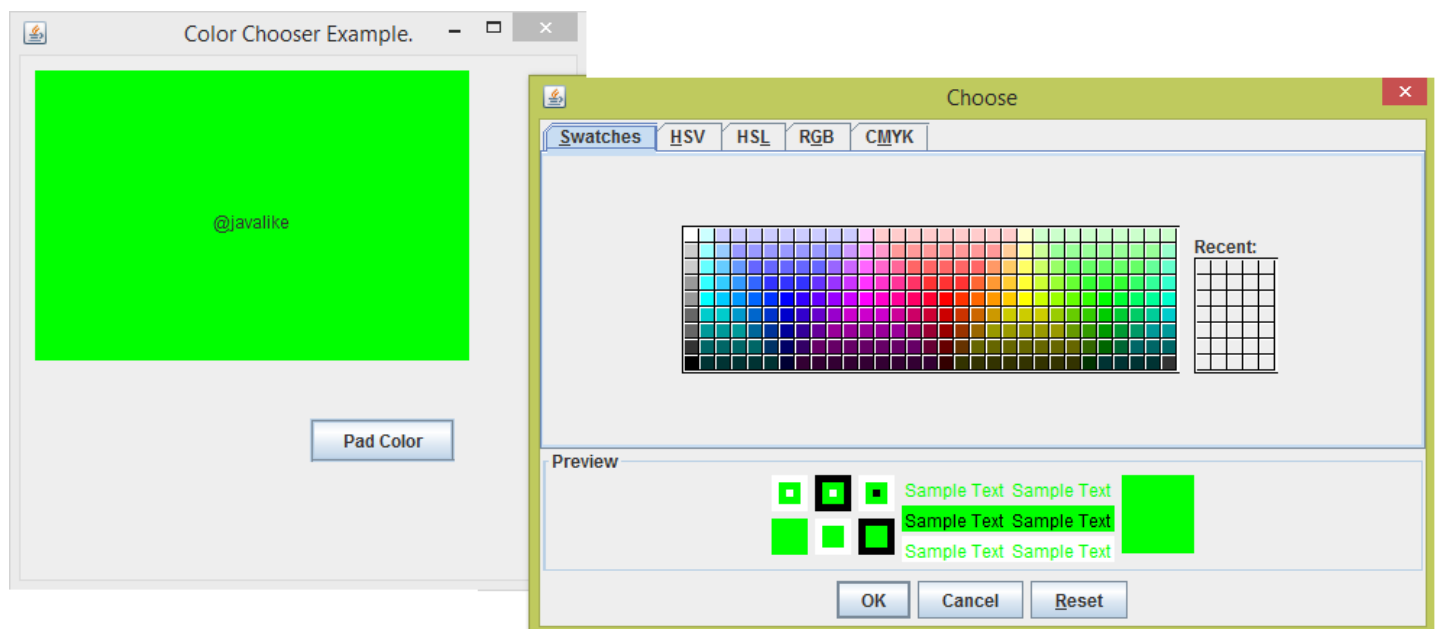
        ta.setBounds(10, 10, 300, 200);
        b.addActionListener(this);
        f.add(b);
        f.add(ta);
        f.setLayout(null);
        f.setSize(400, 400);
        f.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        Color c = JColorChooser.showDialog(this, "Choose", Color.GREEN);
        ta.setBackground(c);
    }

    public static void main(String[] args) {
        new ColorChooserExample();
    }
}

```

خروجی: برنامه هنگام اجرا دارای یک ناحیه متنی `TextArea` و یک دکمه `button` می باشد. ناحیه `TextArea` ابتدا سفید و خالی بود که ما یک متن داخلش نوشتیم و همچنین بعد از کلیک روی دکمه `Color pad` جعبه رنگی ما نمایش داده می شود که رنگ پیشفرض انتخاب شده آن سبز می باشد و بعد از کلیک کردن روی دکمه `OK` موجود در جعبه رنگی ناحیه سفید `TextArea` به سبز تغییر رنگ می دهد. (اگر توضیحات واضح نبود، کاری نداره! کد بالا رو در ایکلیپس یا هر برنامه دیگه اجرا کنید.)



تصویر(۶)

- دستورات مشابه مثال بالا را در این جلسه و جلسات قبل بررسی کرده ایم.

فقط مثال ، مثال و مثال حل کنید!!!!!!

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.