الجلسة السابعة - برمجة 3

الغاية من الجلسة: التعامل مع الأشكال والألوان في الجافا من خلال رسم وجه مبتسم بسيط Smiley.

هذا الكود يُظهر كيفية رسم وجه مبتسم باللون الأصفر باستخدام Java Swing:

```
class DrawSmiley extends JPanel {

    public void paintComponent(Graphics g) {
        super.paintComponent(g); // draw the face
        g.setColor(Color.YELLOW);
        g.fillOval(10, 10, 200, 200);
        // draw the eyes
        g.setColor(Color.BLACK);
        g.fillOval(55, 65, 30, 30);
        g.fillOval(135, 65, 30, 30);
        // draw the mouth
        g.fillOval(50, 110, 120, 60);
        // "touch up" the mouth into a smile
        g.setColor(Color.YELLOW);
        g.fillRect(50, 110, 120, 30);
        g.fillOval(50, 120, 120, 40);
    }
}
```

```
public class DrawSmileyTest {

    public static void main(String[] args) {
        DrawSmiley panel = new DrawSmiley();
        JFrame application = new JFrame();

        application.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        application.add(panel);
        application.setSize(230, 250);
        application.setVisible(true);
    }
}
```

1. تعريف الكلاس `DrawSmiley`:

- يتم تعريف كلاس يرث من `JPanel`، وهو عبارة عن لوحة تُستخدم لعرض رسوميات مخصصة.

- داخل كلاس `DrawSmiley` طريقة `paintComponent` التي تُستخدم لرسم المكونات داخل الـ `JPanel` وهي override.

2. طريقة `paintComponent`:

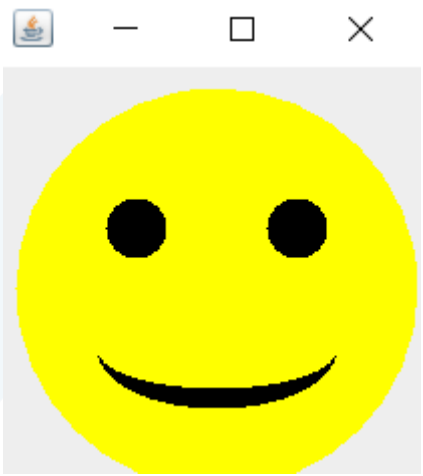
- يتم استدعاء `super.paintComponent(g)` لضمان أن العمليات القياسية للرسم مثل مسح اللوحة تُنفذ بشكل صحيح.
- يتم تعيين لون الرسم إلى اللون الأصفر باستخدام `g.setColor(Color.YELLOW)`.
- يتم استخدام `g.fillOval` لرسم دائرة ملونة تمثل وجه الوجه.
- يُرسم العينين باستخدام دوائر سوداء صغيرة.
- يُرسم الفم باستخدام دائرة بلون أسود.

التعليمات الثلاث الأخيرة في الطريقة painComponent هي لجعل الوجه مبتسماً.

3. الكلاس `DrawSmileyTest`:

- يتم تعريف الكلاس `DrawSmileyTest` الذي يحتوي على الطريقة الرئيسية `main`.
- في الطريقة الرئيسية، يتم إنشاء كائن من `DrawSmiley` واسمه panel.
- يتم إنشاء نافذة (`JFrame`) وتهيئتها لعرض الواجهة application.
- يتم إضافة `DrawSmiley` كمكون إلى النافذة من خلال الطريقة add.
- يتم تحديد حجم النافذة وجعلها مرئية للمستخدم باستخدام setSize و setVisible.

هذا الكود سيقوم بعرض وجه مبتسم باللون الأصفر على نافذة صغيرة عند تشغيله.



سؤال مهم:

ما الفرق بين الـ JPanel والـ JFrame؟

JFrame وJPanel هما جزءان أساسيان في مكتبة Java Swing المستخدمة لتطوير واجهات المستخدم الرسومية (GUI). إليك الفرق بينهما:

1. JFrame:

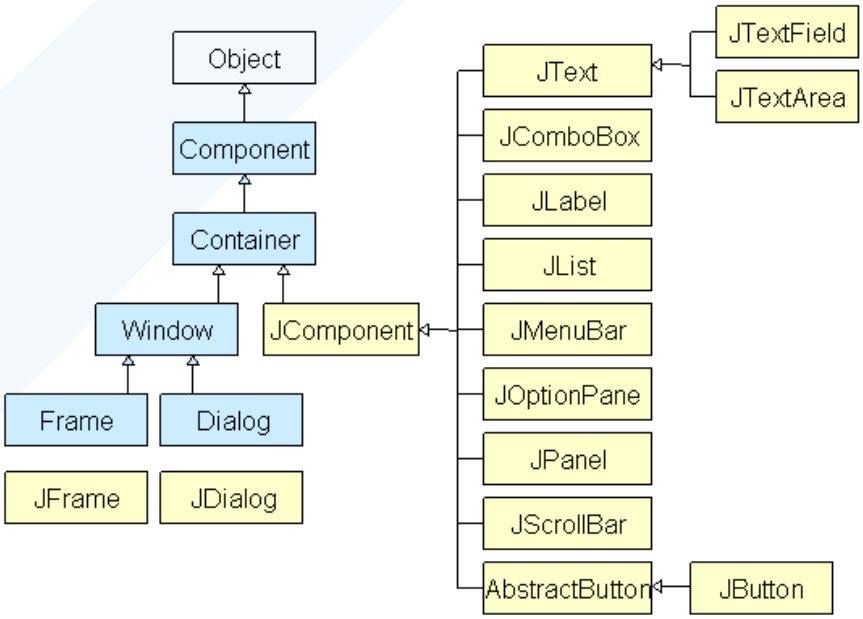
- JFrame هو إطار (Frame) رئيسي يُستخدم لإنشاء نافذة رئيسية لتطبيقك.
- يمكنك إضافة مكونات GUI أخرى إلى JFrame مثل JButton وJTextField وغيرها لبناء تطبيق GUI متكامل.
- يوفر JFrame الإطار الخارجي للتطبيق الذي يحتوي على عناصر التحكم الأساسية مثل الأزرار والقوائم.
- يُستخدم JFrame لتحديد خصائص النافذة مثل الحجم، العنوان، وإغلاق التطبيق.

2. JPanel:

- JPanel هو حاوية (Container) تُستخدم لتنظيم وتجميع مكونات GUI داخلها.
- **يمكن إضافة JPanel إلى JFrame** أو أي إطار آخر لتنظيم المكونات بطريقة منظمة.
- يُمكن استخدام JPanel لتجميع المكونات ذات العلاقة ببعضها البعض في مجموعات منطقية.
- يُستخدم JPanel لإنشاء مساحات لعرض المكونات مثل الأزرار والحقول النصية والرسومات.

بشكل عام، JFrame هو الإطار الرئيسي للتطبيق الذي يحتوي على النافذة والعناصر التحكم الرئيسية، بينما يُستخدم JPanel لتنظيم وتجميع المكونات داخل الإطارات الرئيسية للعرض بشكل منظم ومرتب.

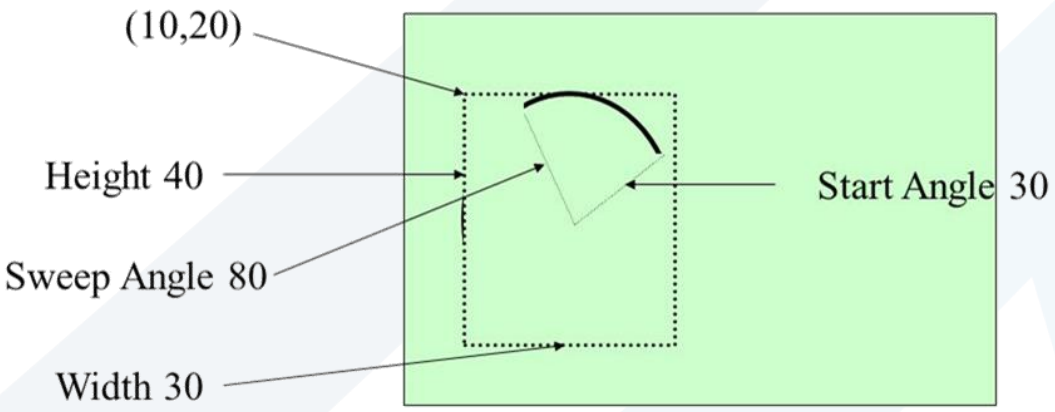
الشكل الآتي يبين أنواع العناصر في Java Swing، لاحظ أن جميع العناصر التي نتعامل معها من نوافذ وحقول وأزرار فإن نمطها الأساس هو Object.



تعليمة رسم قوس هي drawArc حيث نمرر لها الوسطاء الستة الموضحة في الشكل أدناه:

حيث أن المستطيل هو يعبر عن مساحة القوس وهو وهمي غير مرئي.

```
public void paint(Graphics g) {
  g.drawArc(10,20,30,40,30,80);
}
```



إن `fillArc` و `drawArc` هما طرق مختلفة لرسم أقواس (Arcs) في Java Graphics.

1. `fillArc`:

- تستخدم لرسم أقواس ممتلئة (Filled Arcs)، وهي تقاطعات قوسية تمتلئ باللون.
- يمكن استخدامها لرسم أشكال ممتلئة مثل دوائر جزئية (Pie charts) أو تظليلات (Shaded areas).

2. `drawArc`:

- تستخدم لرسم أقواس فقط (Arcs)، وهي مجرد خطوط تمثل الحدود الخارجية للأقواس دون ملء الداخل.
- يمكن استخدامها لرسم أشكال هندسية مثل دوائر جزئية (Pie charts) أو أشكال مجردة بدون تظليل.

نهاية الجلسة