



Die Klasse Graphics

Die Klasse **Graphics** modelliert einen grundlegenden graphischen Kontext für Component-Objekte und ermöglicht durch einfache Operationen Zeichnungen zu erstellen.

Die **Graphics**-Objekte sind Erweiterungen der abstrakten Klasse **Graphics**, die durch das Ausgabe-Objekt erst instanziert werden.

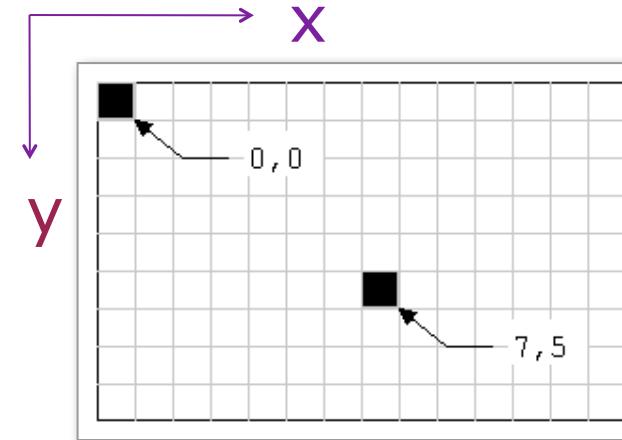
Die **Graphics**-Objekte speichern zusätzlich den aktuellen Zustand des Ausgabegerätes.

z.B. **Hintergrundfarbe**
Zeichenfarbe
Font
u.s.w.



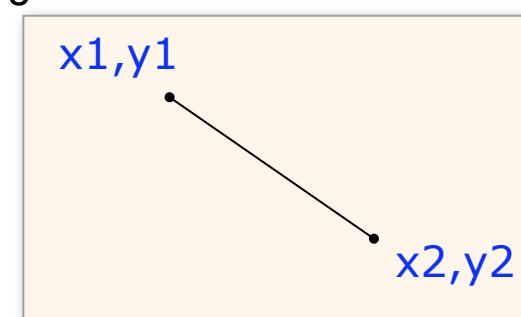
Zeichenoperationen der Klasse Graphics

Koordinatensystem



`drawLine (int x1, int y1, int x2, int y2);`

Diese Methode zeichnet eine Linie von dem Punkt (x_1, y_1) zu dem Punkt (x_2, y_2)



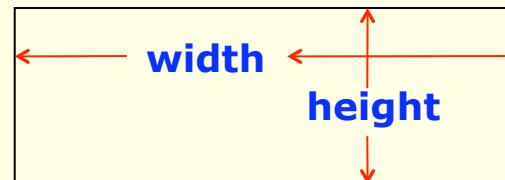


`drawRect (x1, y1, width, height);`

Diese Methode zeichnet ein Rechteck

0, 0

x1,y1



Zeichenoperationen....

`fillRect (x1, y1, width, height);`

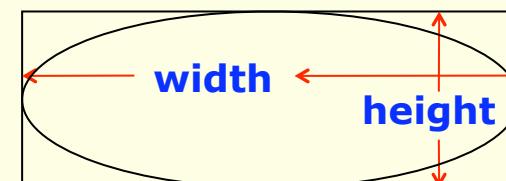
füllt das Rechteck mit der aktuellen Zeichenfarbe

`drawOval (x1, y1, width, height);`

Diese Methode zeichnet ein Oval

0, 0

x1,y1



`fillOval(x1, y1, width, height);`

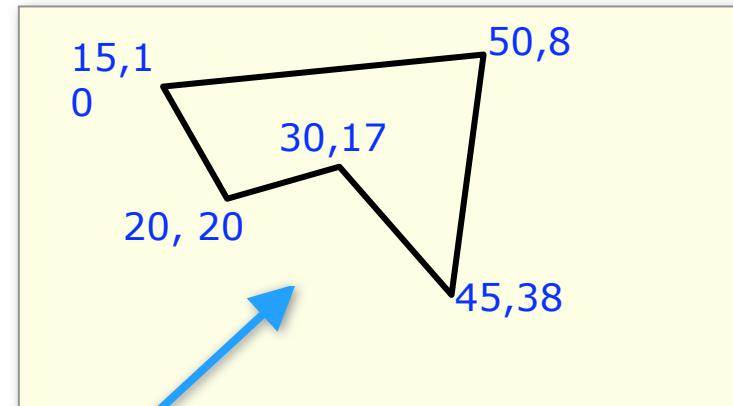
füllt das Oval mit der

aktuellen Zeichenfarbe



Zeichenoperationen der Klasse Graphics

```
int x[] = { 15, 20, 30, 45, 50, 15 };
int y[] = { 10, 20, 17, 38, 8, 10 };
int points = x.length;
Polygon poly = new Polygon(x, y, points);
poly.addPoint(60, 65);
```



```
drawPolygon( poly );
```

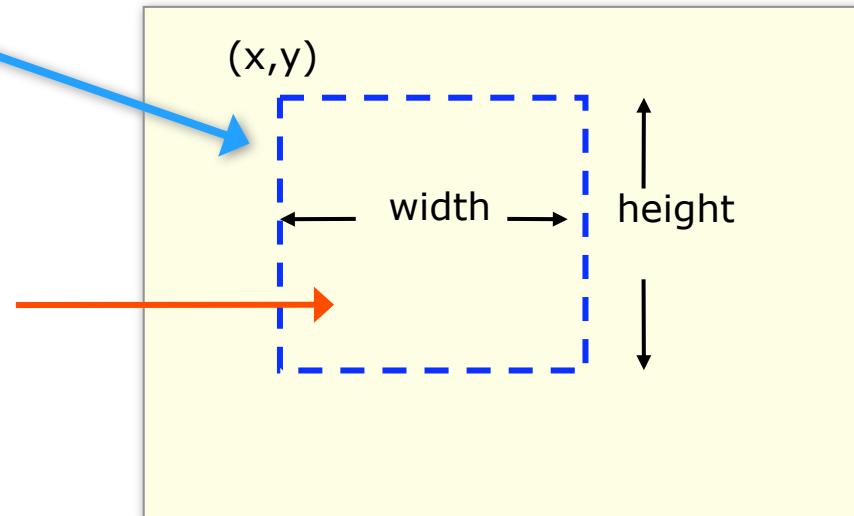
Die Methode `fillPolygon(poly)` schließt das Polygon automatisch, ohne dass ein übereinstimmender Anfangs- und Endpunkt nötig ist.



Zeichenoperationen der Klasse Graphics

`setClip (x, y, width, height);`

Nur ein Bereich wird für
weitere Maloperationen
sichtbar.



`drawString(String str, int x, int y)`

Ein Text wird ab der (x,y) -Position geschrieben