

## Pertemuan ke-3: Dasar-dasar Grafik 2D

---

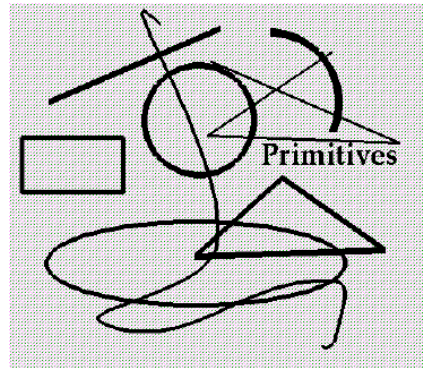


Image Sources:

<http://thewizardofhoz.com/graphics.html>

<http://www.siggraph.org/~rhyne/com97/com97-tut.html>

---

CG1: Universitas Pamulang

## Grafik 2D

---

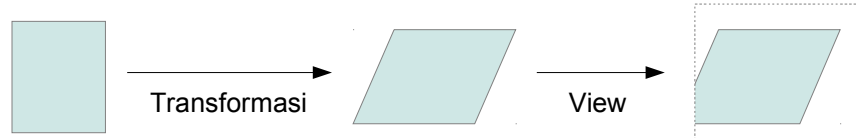
- Memetakan objek-objek kedalam dunia 2 (dua) dimensi.
- Pemodelan dan Proses Rendering relatif lebih mudah dari Grafik 3D (tidak perlu proyeksi, dll)
- Konsep dasar Grafik 2D meliputi:
  - Rendering pipeline
  - Sistem koordinat (object, world, device)
  - Grafik primitif
  - Transformasi
  - Warna
  - Font, dan sebagainya

---

CG1: Universitas Pamulang

# Proses Rendering

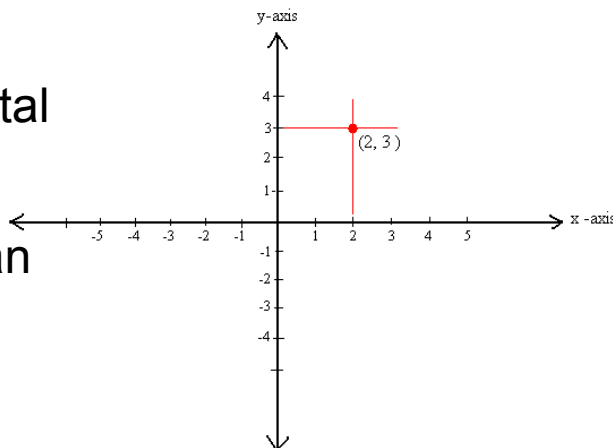
- Dalam Grafik 2D, dunia asal (*virtual world*) dan dunia untuk menampilkan objek (*rendering world*) sama-sama 2D



- Proses rendering relatif sederhana meliputi:
  - Konstruksi objek
  - Transformasi (rotasi, translasi, scaling)
  - Penambahan warna dan properti lain
  - Proses Render di Grafik Device

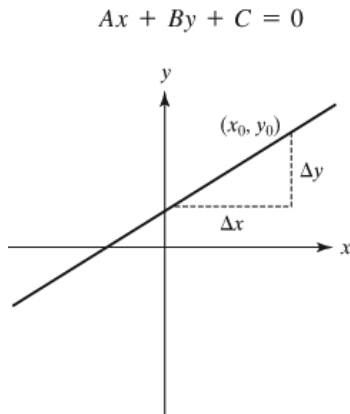
# Sistem koordinat

- Koordinat yang umum adalah koordinat kartesius
- Sumbu x : horisontal
- Sumbu y : vertikal
- Posisi digambarkan sebagai titik (x, y)

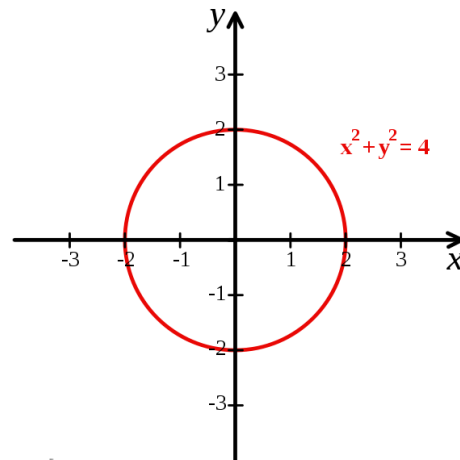


# Contoh Objek Geometri

- Garis



- Lingkaran



- Elipse:  $\frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1$

CG1: Universitas Pamulang

## Ruang (*space*)

- Dalam komputer grafik ada 3 macam ruang
  - Ruang objek (*object space*)  
Sistem koordinat yang diasosiasikan secara lokal pada masing-masing objek
  - Ruang dunia (*world space*)  
Sistem koordinat umum yang digunakan untuk menempatkan objek atau model
  - Ruang devais (*device space*)  
Sistem koordinat yang digunakan oleh devais tertentu (layar monitor, printer, dsb)

CG1: Universitas Pamulang

# Sistem koordinat pada Java

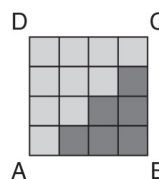
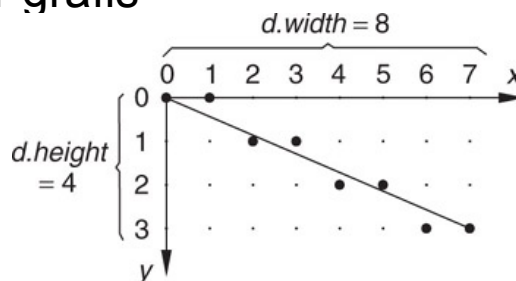
- Berikut adalah contoh sistem koordinat devais. Pojok kiri atas adalah titik (0,0), sumbu x positif ke kanan, sumbu y positif ke kiri.
- Dalam Java, *defaultnya*, *world space* dan *device space* sama.



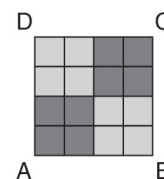
CG1: Universitas Pamulang

## Konsep Pixel

- Piksel adalah representasi sebuah titik terkecil dalam sebuah gambar grafis
- Piksel kecil --> kualitas grafik bagus
- Contoh objek:
  - Garis
  - Segitiga
  - Segiempat



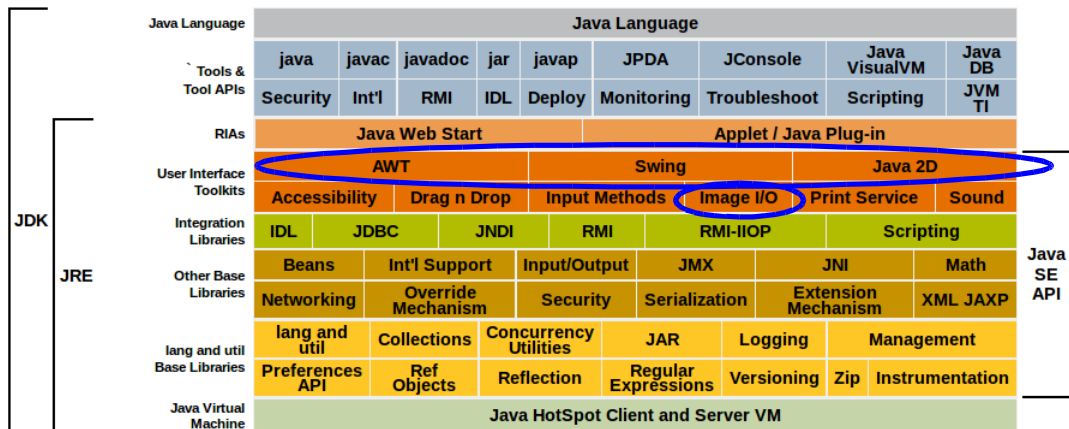
(a)



(b)

CG1: Universitas Pamulang

# Komponen-komponen Java untuk Graphics 2D



CG1: Universitas Pamulang

## Deskripsi masing-masing komponen

- AWT: Abstract Window Toolkit, kumpulan API (fungsi-fungsi) untuk mengimplementasikan user interface berbasis grafik
- Swing : Kumpulan komponen untuk membuat user interface berbasis grafik di Java
- Java2D : Kumpulan komponen untuk membuat grafik 2D dan imaging (pengolahan citra)
- Image I/O : Kumpulan komponen untuk membaca/menulis image dalam berbagai format

CG1: Universitas Pamulang

# Apa yang disediakan oleh Java2D

---

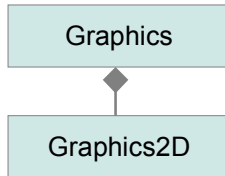
- Model yang sama untuk render pada “Display Devices” dan Printer
- Menyediakan berbagai macam geometri primitif seperti: Kurva, Segiempat, Ellipse dan mekanisme untuk membuat berbagai bentuk geometri
- Fungsi-fungsi untuk melakukan hit deteksi pada shape, text dan image
- Model untuk menggabungkan bagaimana merender objek-objek yang overlap
- Manajemen warna
- Fungsi-fungsi untuk mencetak ke printer
- Pengendalian kualitas objek yang dirender melalui mekanisme “rendering hints”

## Package-package terkait

---

- Berikut adalah package-package yang sering diimport dalam program untuk komputer grafik 2D
- AWT:
  - java.awt, java.awt.color, java.awt.font, java.awt.geom, java.awt.print, java.awt.image
- Swing:
  - javax.swing, javax.swing.event, javax.swing.text
- Image I/O:
  - javax.imageio

# Class yang digunakan



- Graphics menyimpan informasi:
  - Komponen tempat menggambar objek 2D
  - Koordinat untuk rendering dan 'clipping'
  - Warna
  - Font
  - Operasi pixel: XOR atau paint
- Graphics2D
  - Kelas mendasar untuk render objek 2D
  - Render geometri primitif, transformasi koordinat, manajemen warna, dan text layout

## *Class Graphics2D*

- *Graphics2D* merupakan kelas utama untuk render objek 2 (dua) dimensi
- Kelas ini memiliki berbagai *method* yang dapat digunakan untuk membuat objek 2D
- Untuk API selengkapnya dapat dilihat di  
<http://download.oracle.com/javase/7/docs/api/java/awt/Graphics2D.html>
- Pada umumnya objek di-render menggunakan *JPanel*, dengan meng-override *paintComponent*

# Kerangka Class GrafikCanvas

```
1 //import package yang diperlukan
2 import java.awt.*;
3 import java.awt.event.*;
4 import java.awt.geom.*;
5 import javax.swing.*;
6
7 public class GrafikCanvas extends JPanel {
8     public GrafikCanvas() {
9         this.setPreferredSize(new Dimension(400, 150));
10        this.setBackground(Color.WHITE);
11    }
12    public void paintComponent(Graphics g) {
13        super.paintComponent(g);
14        Graphics2D g2 = (Graphics2D)g;
15        // kode render grafik objek
```

CG1: Universitas Pamulang

# Kerangka class Main untuk menampilkan GrafikCanvas

```
1 //import package yang diperlukan
2 import java.awt.*;
3 import java.awt.event.*;
4 import java.awt.geom.*;
5 import javax.swing.*;
6
7 public class Main {
8     public static void main(String[] args) {
9         JFrame frame = new JFrame();
10        frame.setTitle("Aplikasi Komputer Grafik");
11        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
12
13        //tambahkan canvas
14        MainPanel canvas = new GrafikCanvas();
15        frame.getContentPane().add(canvas);
16        frame.pack();
17        frame.setVisible(true);
18    }
19 }
```

CG1: Universitas Pamulang



# Atribut-atribut pada Graphics2D

- Pen: outline dari objek seperti solid, putus-putus, ukuran pena, dsb
- Fill: bagaimana interior dari objek, seperti warna, gradient atau dengan pola
- Composite: bagaimana merender gabungan beberapa objek
- Transform: translasi, rotasi, scalling, shearing
- Clip: membatasi render pada area tertentu
- Font: menampilkan text
- Rendering hint: kecepatan dan kualitas render

