

# Pertemuan 3

## Oval dan Arc

- Pembentukan Oval
- Pembentukan Arc

# Membangun Bidang “Ovals”

1. Oval digunakan untuk membentuk bidang elips atau lingkaran
2. Bidang **oval** dibangun menggunakan argumen : koordinat x dan y pada sudut atas, dan lebar dan tinggi pada bagian **oval** itu sendiri.
3. Seperti halnya metode yang digunakan pada bidang geometri lainnya, metode **drawOval()** untuk membentuk bidang oval dan metode **fillOval()** digunakan untuk membentuk bidang oval yang berisi terisi

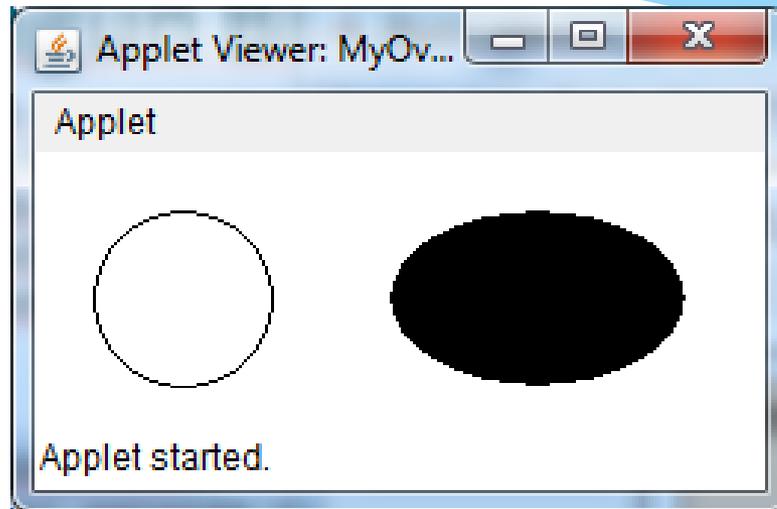
# Membangun Bidang “Ovals”

## Contoh :

```
import java.awt.Graphics;
public class MyOval extends
java.applet.Applet {
    public void paint(Graphics g) {
        g.drawOval(20,20,60,60);
        g.fillOval(120,20,100,60);
    }
}
```

# Membangun Bidang “Ovals”

Hasil :



**Gambar**  
Bidang **lingkaran** (kiri) dan  
Bidang **elips** (kanan)

**Catatan :** jangan karena Anda mengira-ngira bahwa oval itu dibentuk dari beberapa jarak di kiri dan naik keatas dari garis oval itu sendiri. Tetapi anggaplah bahwa oval dibentuk dari sebuah kotak, dengan begitu Anda akan lebih mudah membangun bidang oval ini.

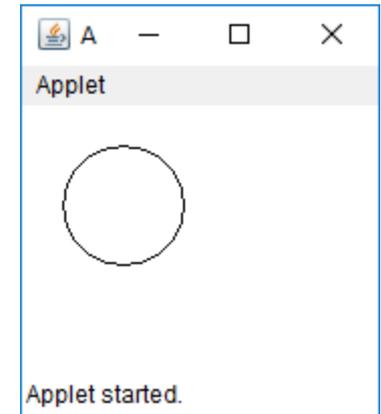
# Pembahasan Oval – Kasus 1

## Kasus 1:

```
import java.awt.Graphics;  
public class MyOval extends java.applet.Applet  
{  
    public void paint(Graphics g) {  
        g.drawOval(20,20,60,60);  
    }  
}
```

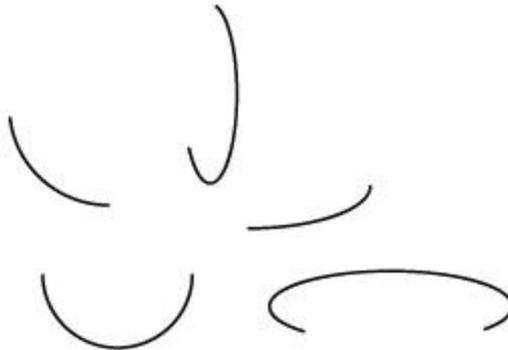
**drawOval(int x, int y, int width, int height)**

Karena nilai  
sama maka  
akan  
linkaran



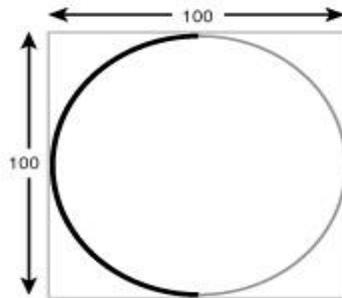
# Membangun Bidang “Arcs”

- \* Metode **drawArc()** memiliki beberapa argumen : sudut awal, lebar dan tinggi, sebuah sudut untuk memulai bentuk arc, sebuah derajat sebagai titik henti.
- \* **drawArc**(int x, int y, int width, int height, int startAngle, int arcAngle)
- \* Hal terpenting yang harus dimengerti mengenai **arc** adalah Anda sebenarnya sedang memformulasi bentuk **arc** sebagai sebuah bidang **oval** dan hanya menggambarkan bagian itu saja

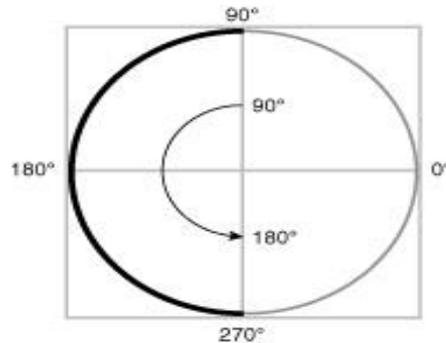


# Membangun Bidang “Arcs”

- ✓ Untuk membangun metode yang membentuk arc, langkah awal adalah membayangkan arc sebagai lingkaran penuh.
- ✓ Kemudian Anda akan menemukan koordinat x dan y, dan titik yang mewakili lebar dan tinggi dari lingkaran tersebut.
- ✓ Keempat nilai tersebut adalah empat argumen yang pertama untuk menggambar menggunakan metode `drawArc()` atau `fillArc()`.

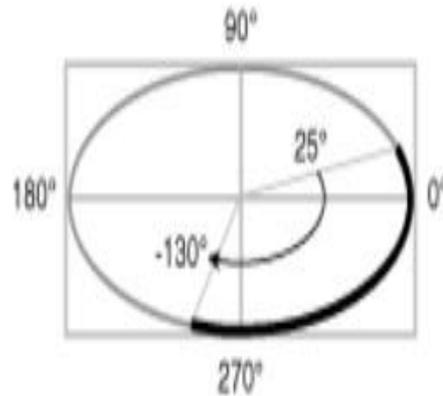


# Membangun Bidang “Arcs”



- ✓ Untuk mendapatkan dua argumen terakhir, sudut mulai lengkung dan dearajat kelengkungan dapat dibayangkan seperti besaran sudut (derajat) putaran berlawanan arah dengan jarum jam.
- ✓ Nol derajat adalah posisi jam 03.00, 90 derajat pada posisi jam 12.00, 180 derajat pada posisi jam 09.00, dan 270 derajat pada posisi jam 06.00.
- ✓ Mulai lengkungan adalah nilai derajat dari bentuk huruf C pada 90 derajat sehingga argumen ke lima adalah 90

# Membangun Bidang “Arcs”



- ✓ Bila kita bermaksud membuat bentuk lengkung C yang berbalikan arah maka kita menyapu 180 derajat ke arah negatif sehingga argumen terakhir menjadi -180 derajat. Gambar berikut ini akan menjelaskan uraian diatas

# Membangun Bidang “Arcs”

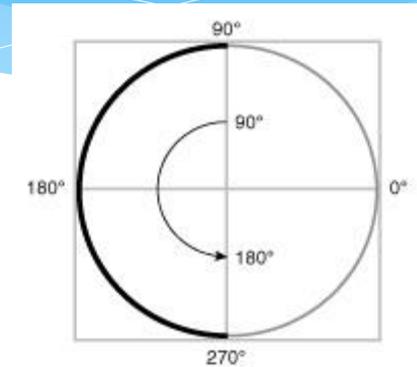
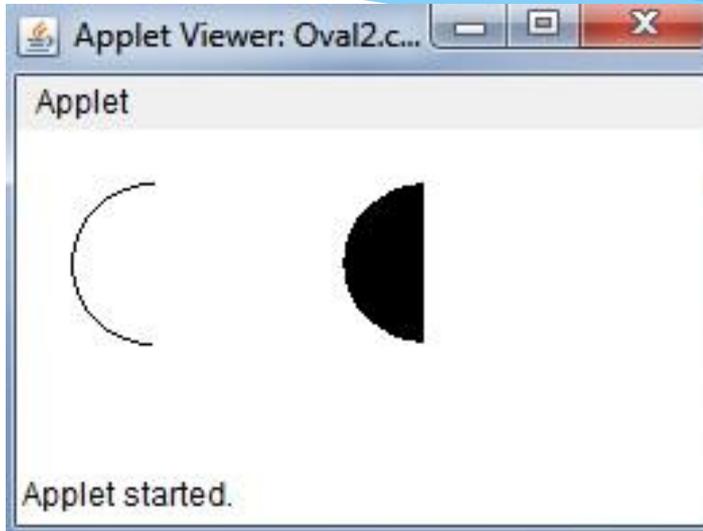
## Contoh 1 :

```
import java.awt.Graphics;

public class MyOvalArc extends
java.applet.Applet {
    public void paint(Graphics g) {
        g.drawArc(20,20,60,60,90,180);
        g.fillArc(120,20,60,60,90,180);
    }
}
```

# Membangun Bidang “Arcs”

Hasil :



## Gambar

Bidang arc membentuk huruf C (kiri) dan bidang arc dengan fill

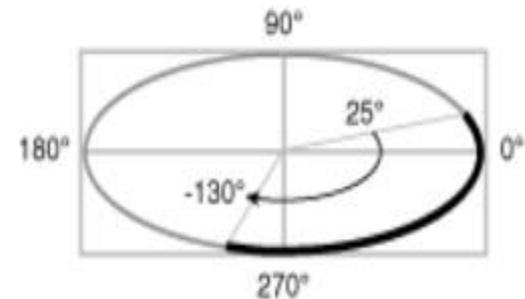
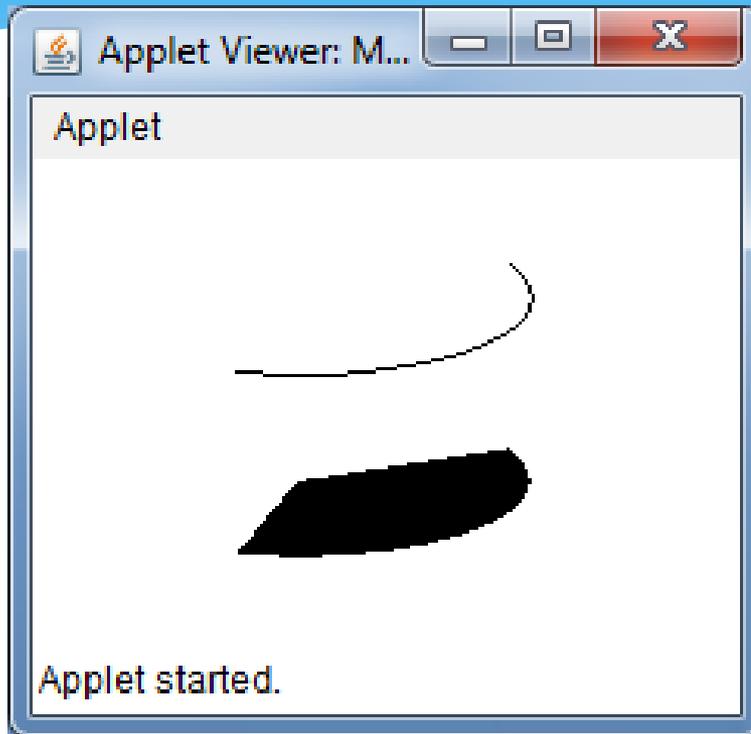
# Membangun Bidang “Arcs”

## **Contoh 2:**

```
import java.awt.Graphics;  
public class MyOval3 extends java.applet.Applet  
{  
    public void paint(Graphics g) {  
        g.drawArc(10,20,150,50,25,-130);  
        g.fillArc(10,80,150,50,25,-130);  
    }  
}
```

# Membangun Bidang “Arcs”

Hasil :



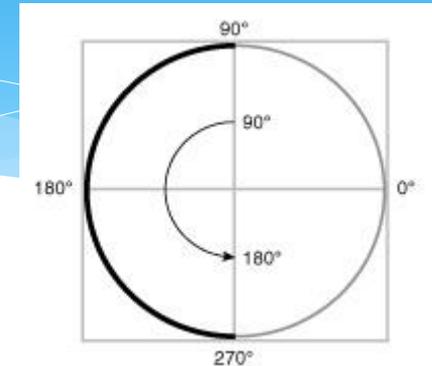
## Gambar

Bidang Arc (kiri) menunjukan hasil dan Bidang Arc (kanan) menunjukan busur arah bidang yang perlihatkan

# Pembahasan Arc – Kasus 1

## Kasus 2:

```
import java.awt.Graphics;  
  
public class MyOvalArc extends  
java.applet.Applet {  
    public void paint(Graphics g) {  
        g.drawArc(20, 20, 60, 60, 90, 180);  
    }  
}
```



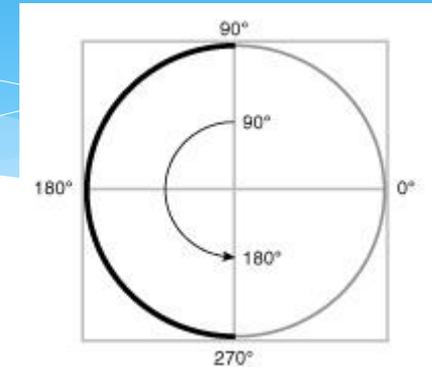
Arah  
putaran  
berlawanan  
jarum jam

**drawArc(int x, int y, int width, int height, int startAngle, int arcAngle)**

# Pembahasan Arc – Kasus 1

## Kasus 2:

```
import java.awt.Graphics;  
  
public class MyOvalArc extends  
java.applet.Applet {  
    public void paint(Graphics g) {  
        g.drawArc(10, 20, 150, 50, 25, -130);  
    }  
}
```



Arah  
putaran  
berlawanan  
jarum jam

**drawArc(int x, int y, int width, int height, int startAngle, int arcAngle)**

# Latihan

## Soal :

1. Buatlah elips Yang Vertikal menggunakan metode `drawOval()`
2. Buatlah seperti bulan sabil menggunakan metode `drawArc`
3. Buat 6 Oval diagonal Kebawah seperti anak tangga

Terima Kasih