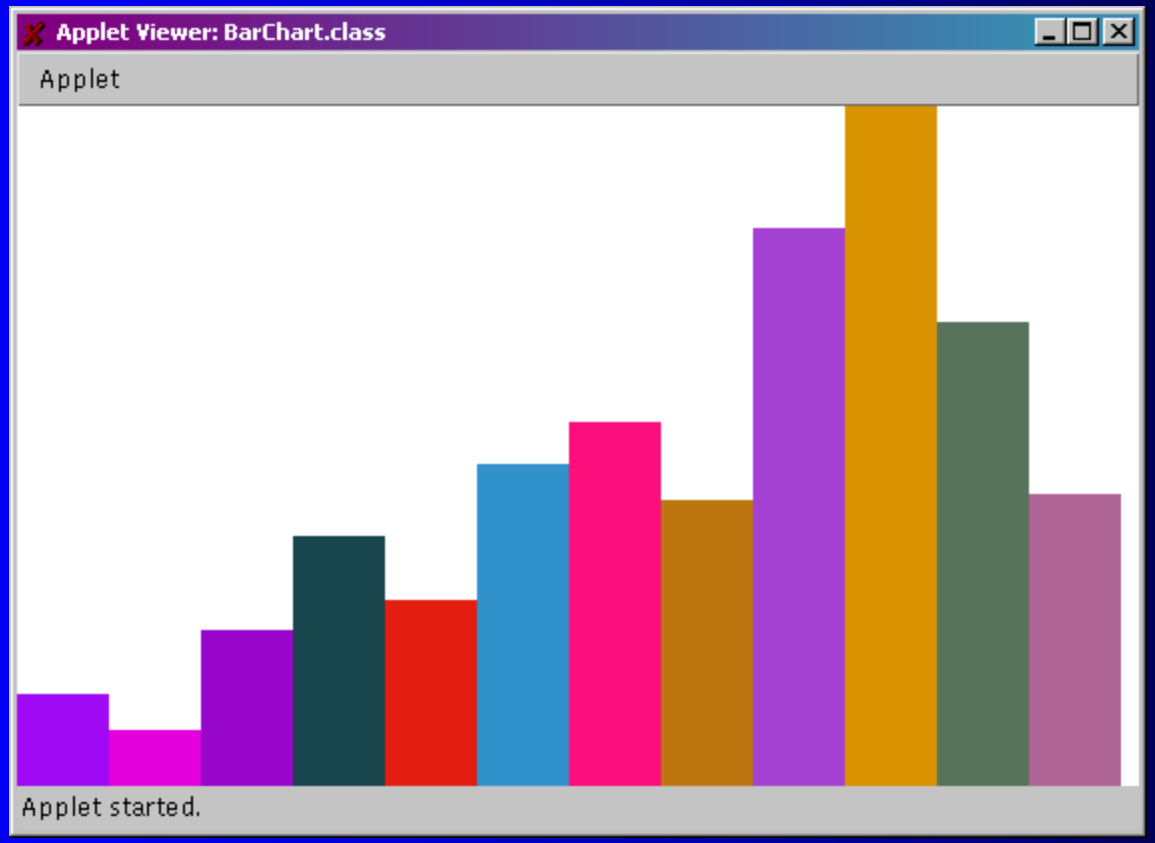
Lecture 35

- Covers
  - Arrays and applets

Bar chart example

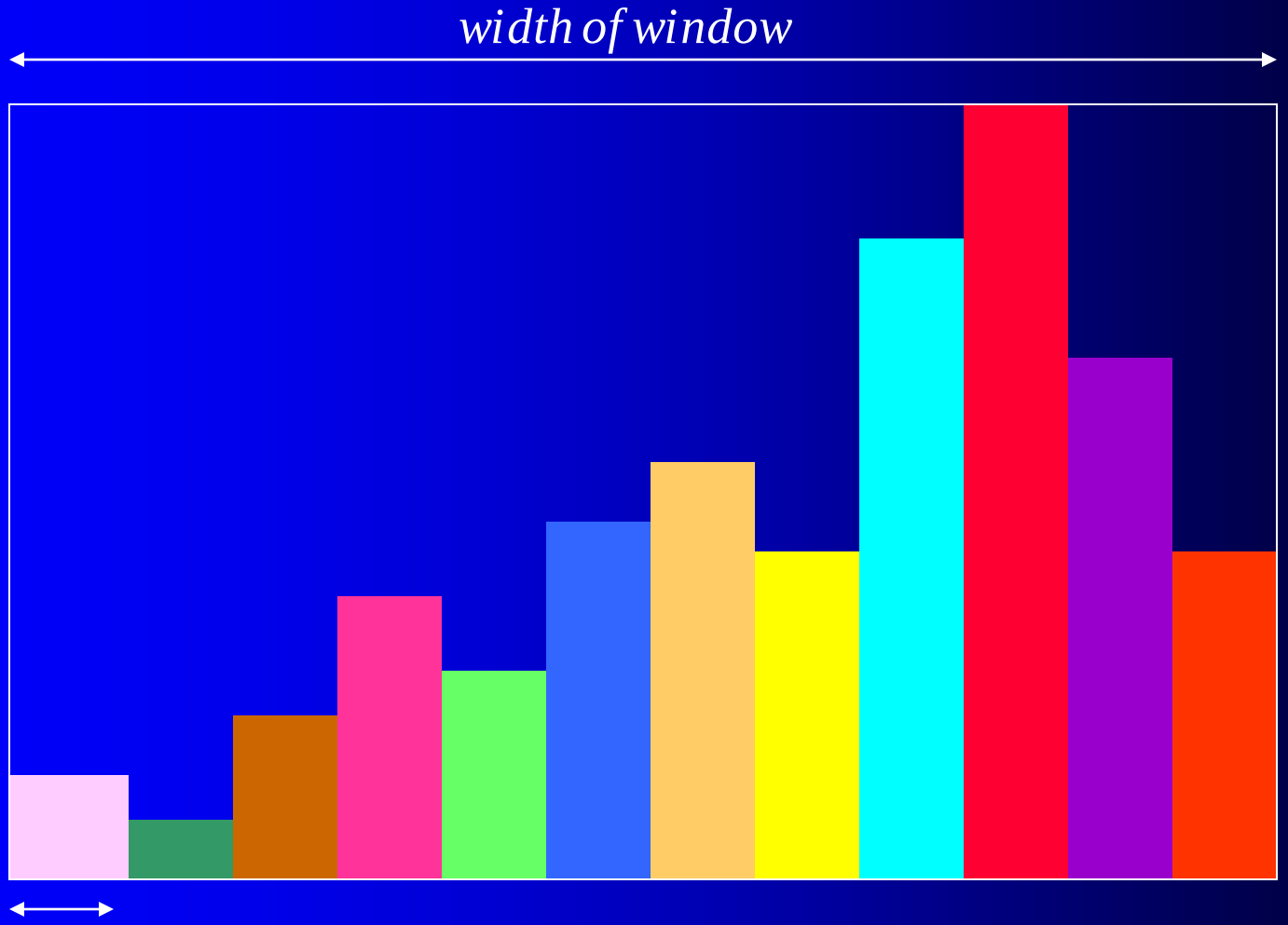
- Write an applet that draws a bar chart given an array of integers



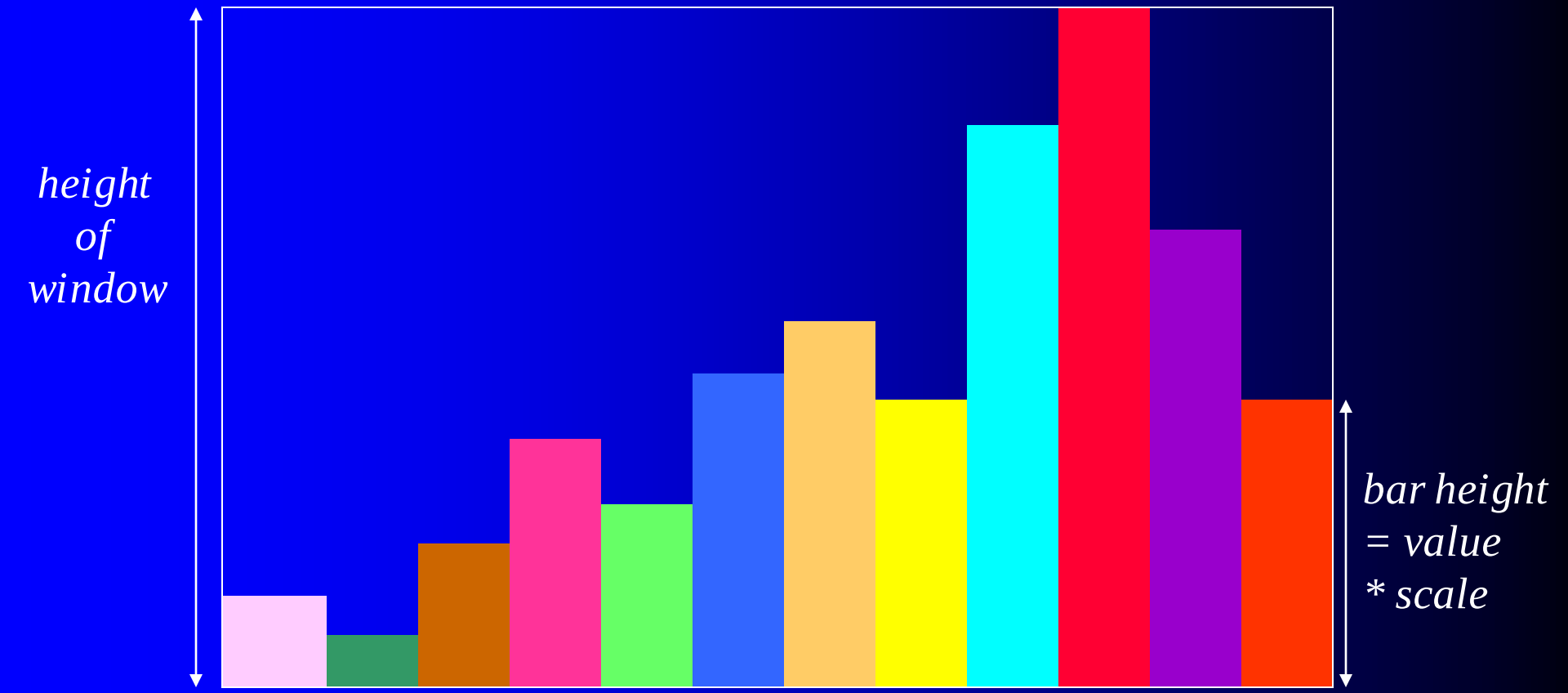
Bar chart example

- Constraints
  - The array may be of any length
  - The bar chart must fill the window as much as possible
  - Each bar is in a random colour

Bar width

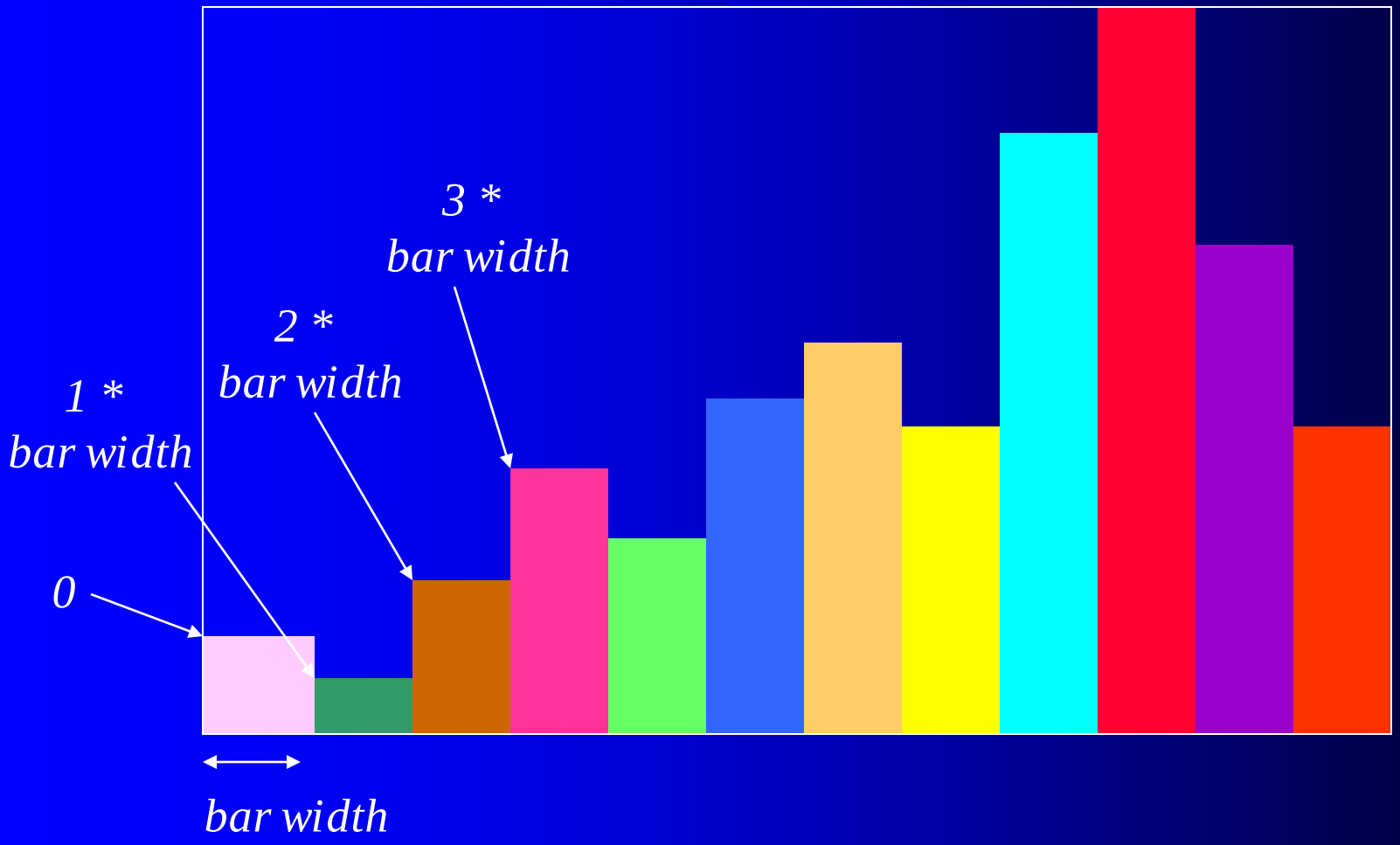


Bar height



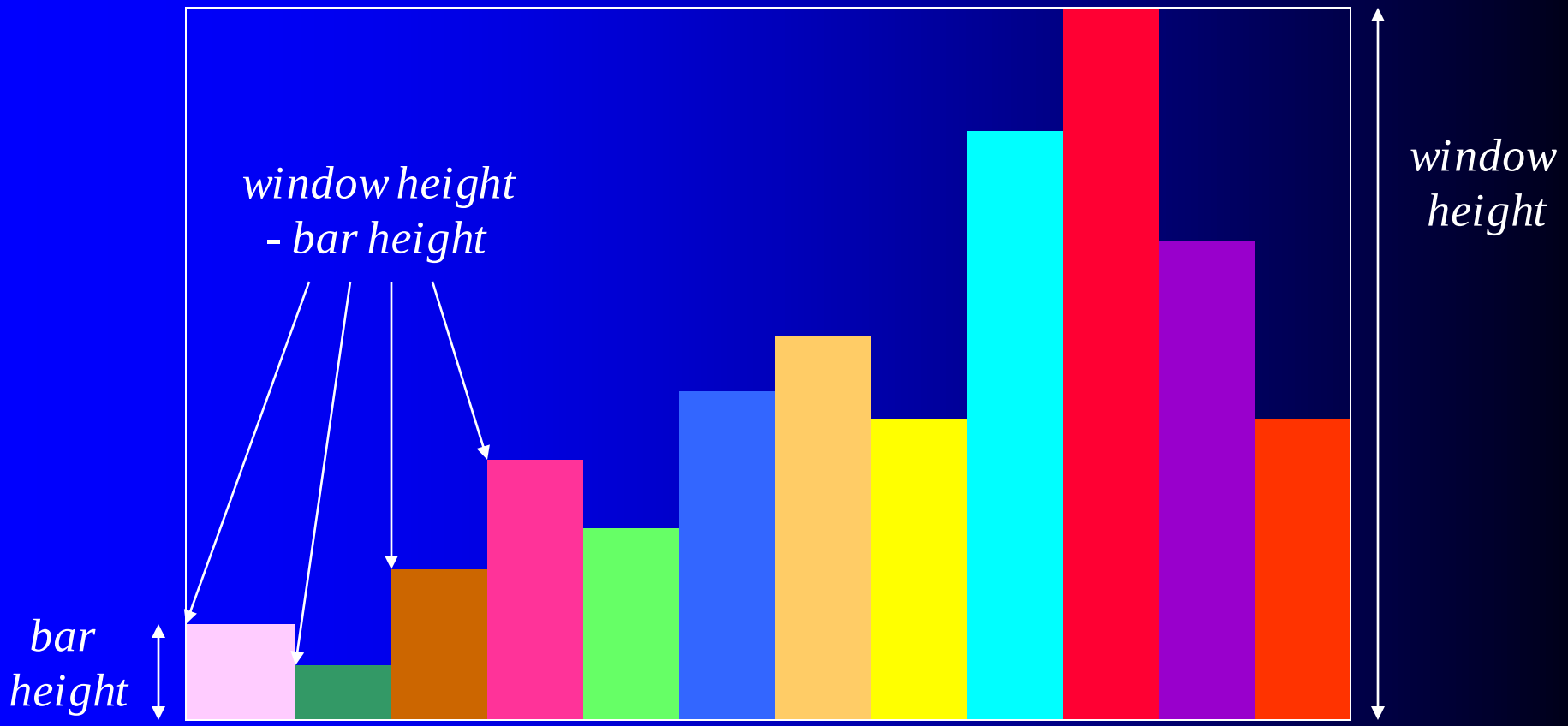
scale = height of window / maximum bar height

x – coord of bar



*x-coord = bar width * bar number (where bar number starts at 0)*

y – coord of bar



y-coord = window height - bar height

Bar chart applet code

```
import java.awt.*;
import java.applet.*;

public class BarChart extends Applet
{
    private int[ ] chartValues;
    private int maximumHeight;

    public void init( )
    {
        int[ ] temp = {13, 8, 22, 35, 26, 45, 51, 40, 78, 95, 65, 41};
        chartValues = temp;
        maximumHeight = maximum( );
        setBackground(Color.white);
    }
```

Bar chart applet code

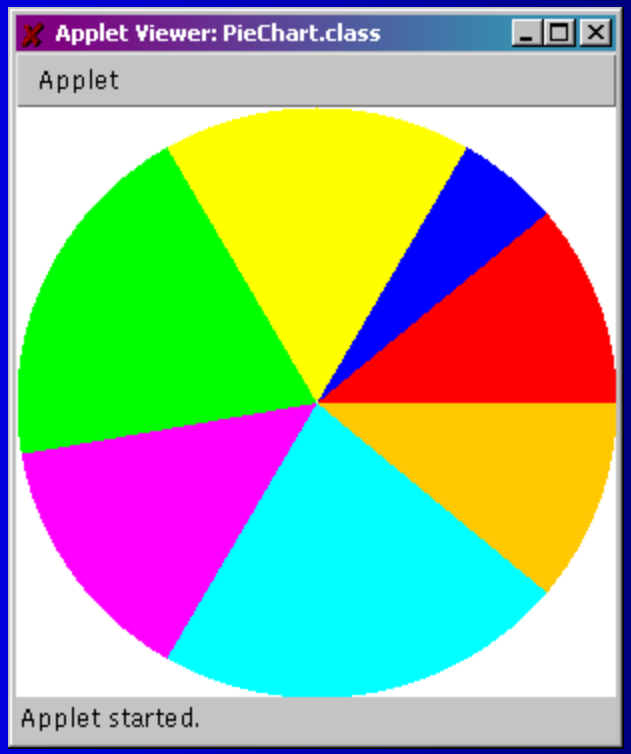
```
public void paint(Graphics page)
{
    int windowHeight = getHeight( );
    int windowWidth = getWidth( );
    int barWidth = windowWidth / chartValues.length;
    double heightScale = (double) windowHeight / maximumHeight;

    for (int i = 0; i < chartValues.length; ++i)
    {
        Color c = randomDrawingColor( );
        page.setColor(c);

        int barHeight = (int) (chartValues[i] * heightScale);
        int xc = i * barWidth;
        int yc = windowHeight - barHeight;

        page.fillRect(xc, yc, barWidth, barHeight);
    }
}
```

Bar chart applet code

```
private int maximum( )
{
    int max = chartValues[0];
    for (int i = 1; i < chartValues.length; ++i)
    {
        if (chartValues[i] > max)
        {
            max = chartValues[i];
        }
    }
    return max;
}

private Color randomDrawingColor( )
{
    int r = (int) (Math.random() * 256);
    int g = (int) (Math.random() * 256);
    int b = (int) (Math.random() * 256);
    return new Color(r,g,b);
}
```

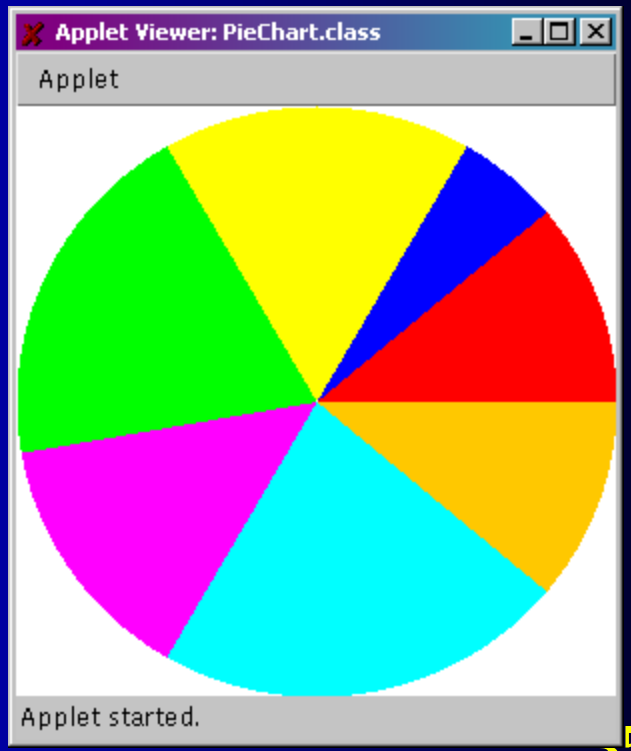
Pie chart example

- Given an array of values, draw a pie chart using the predefined colours



Pie chart example

- The chart should fill up the screen
- Thus, the bounding box for the ovals for all the segments should be
  - start coords: (0,0)
  - height = height of window
  - width = width of window



Pie chart example

Segment	Values	Percentage value	Start angle	Extent
0	20	0.11111	0	0.11111 * 360 = 40
1	10	0.05555	40	0.05555 * 360 = 20
2	30	0.16666	60	0.16666 * 360 = 60
3	35	0.19444	120	0.19444 * 360 = 70
4	25	0.13888	190	0.13888 * 360 = 50
5	40	0.22222	240	0.22222 * 360 = 80
6	20	0.11111	320	0.11111 * 360 = 40

Total: 180

Pie chart example code

```
public class PieChart extends Applet
{
    private int[ ] chartValues = {20, 10, 30, 35, 25, 40, 20};
    private double[ ] percentage;

    private int totalOfChartValues;

    private Color[ ] colors = {Color.red, Color.blue, Color.yellow,
                               Color.green, Color.magenta, Color.cyan,
                               Color.orange, Color.pink};
```

Pie chart example code

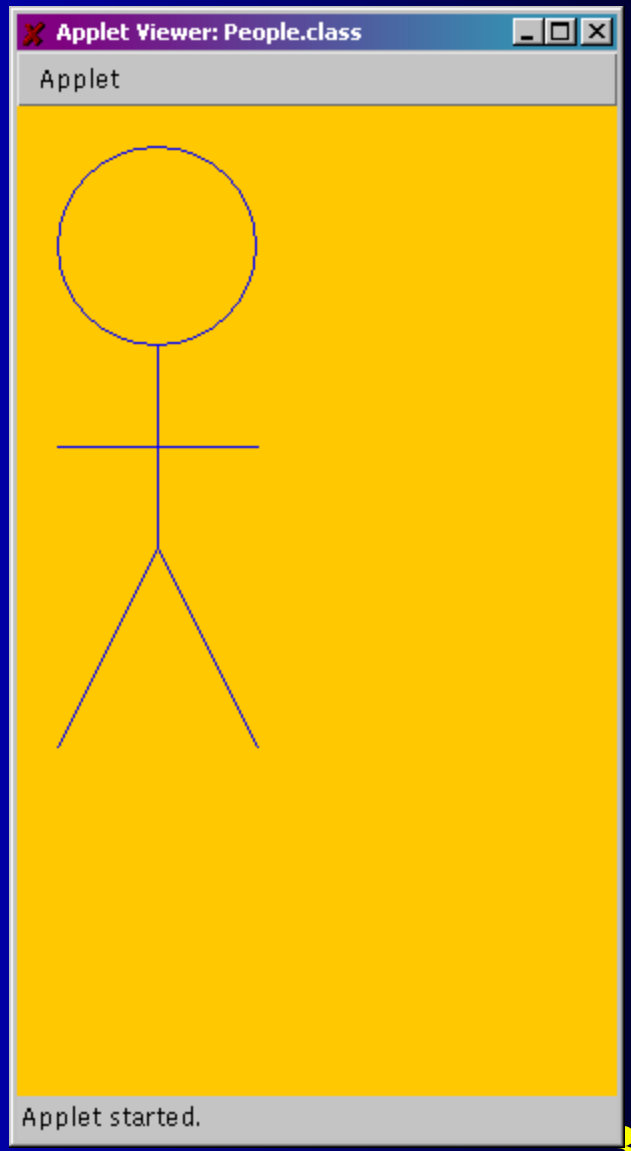
```
public void init( )
{
    setBackground(Color.white);
    totalOfChartValues = 0;
    for (int i = 0; i < chartValues.length; ++i)
    {
        totalOfChartValues += chartValues[i];
    }

    percentage = new double[chartValues.length];
    for (int i = 0; i < chartValues.length; ++i)
    {
        percentage[i] = (double) chartValues[i] / totalOfChartValues;
    }
}
```

Pie chart example code

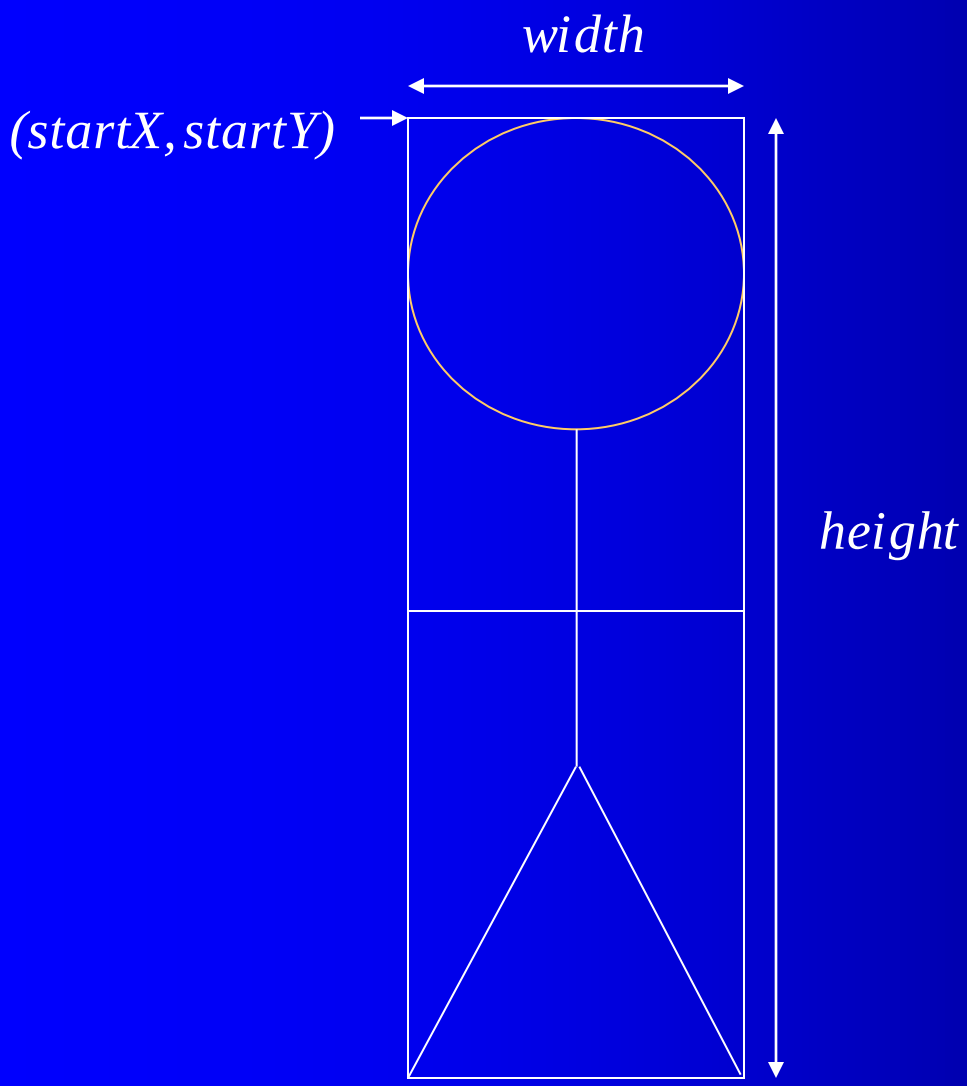
```
public void paint(Graphics page)
{
    int windowHeight = getHeight( );
    int windowWidth = getWidth( );

    int startAngle = 0;

    for (int i = 0; i < chartValues.length; ++i)
    {
        page.setColor(colors[i%colors.length]);

        int extent = (int) (Math.round(percentage[i] * 360));
        page.fillArc(0, 0, windowWidth, windowHeight, startAngle, extent);

        startAngle += extent;
    }
}
```

Boy example

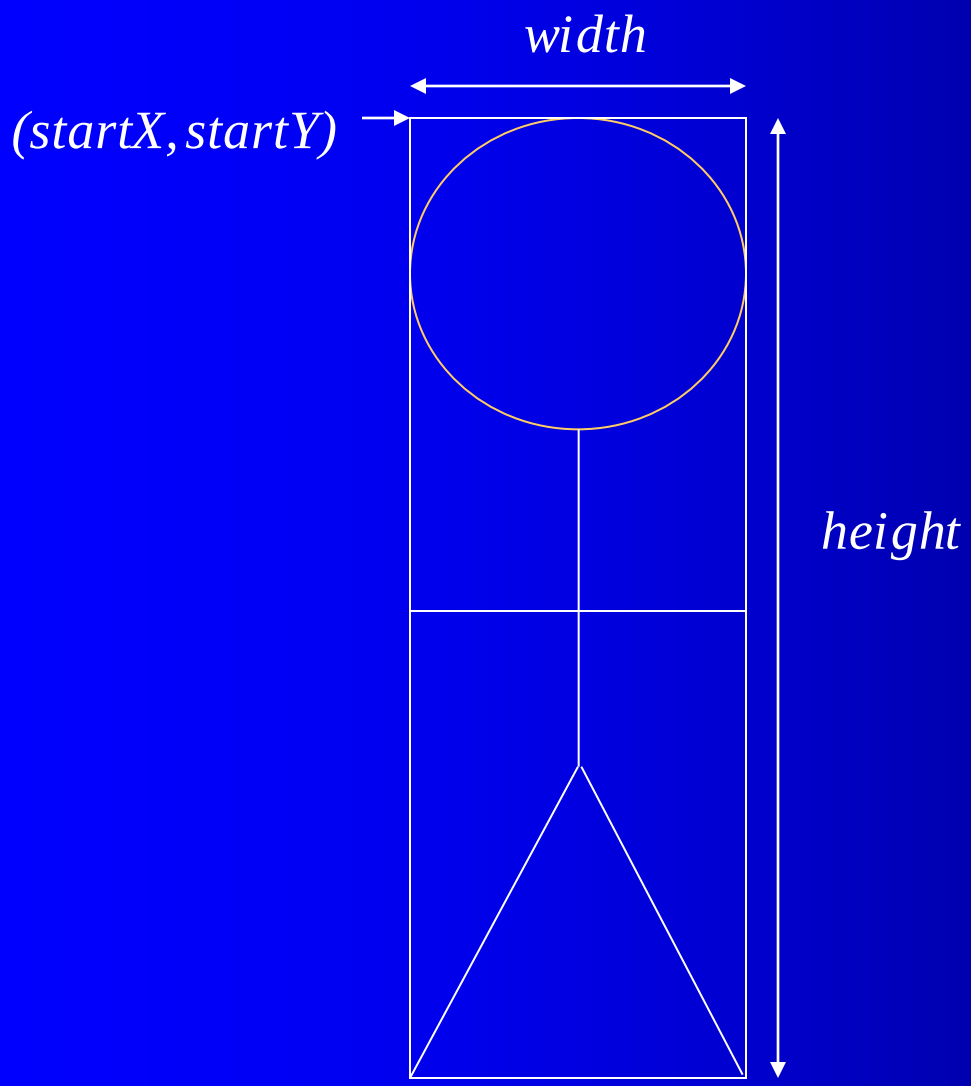
- Write a class Boy that has a draw method that takes a Graphics page and draws on that page a stick figure of a boy



Boy example

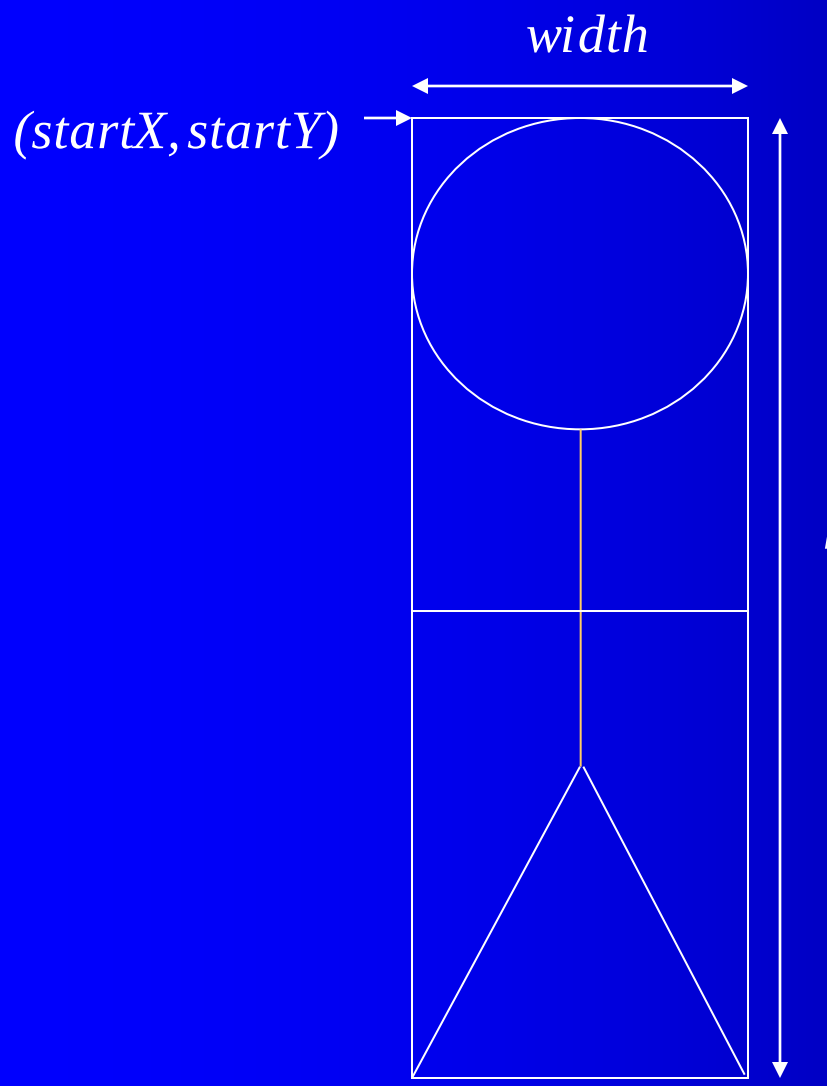


Head



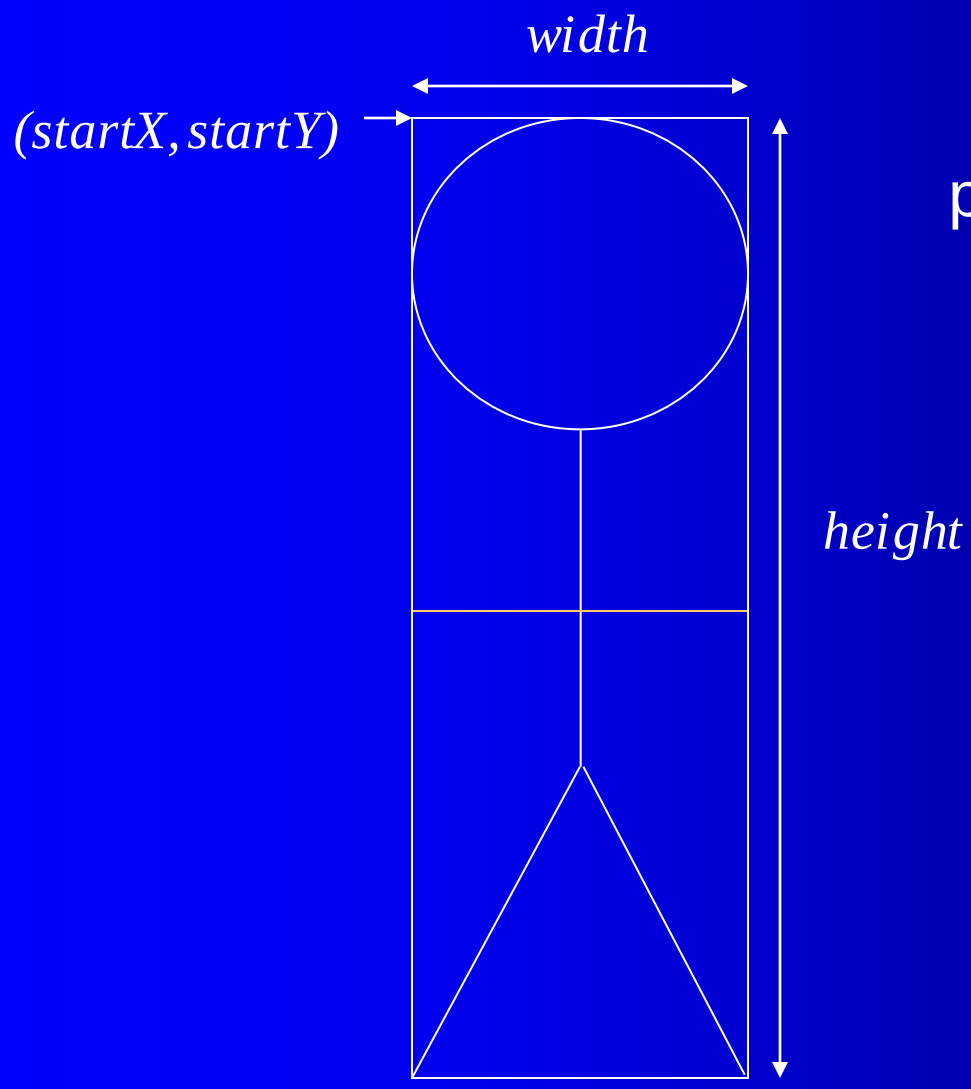
drawOval(startX, startY,
width, height/3);

Body



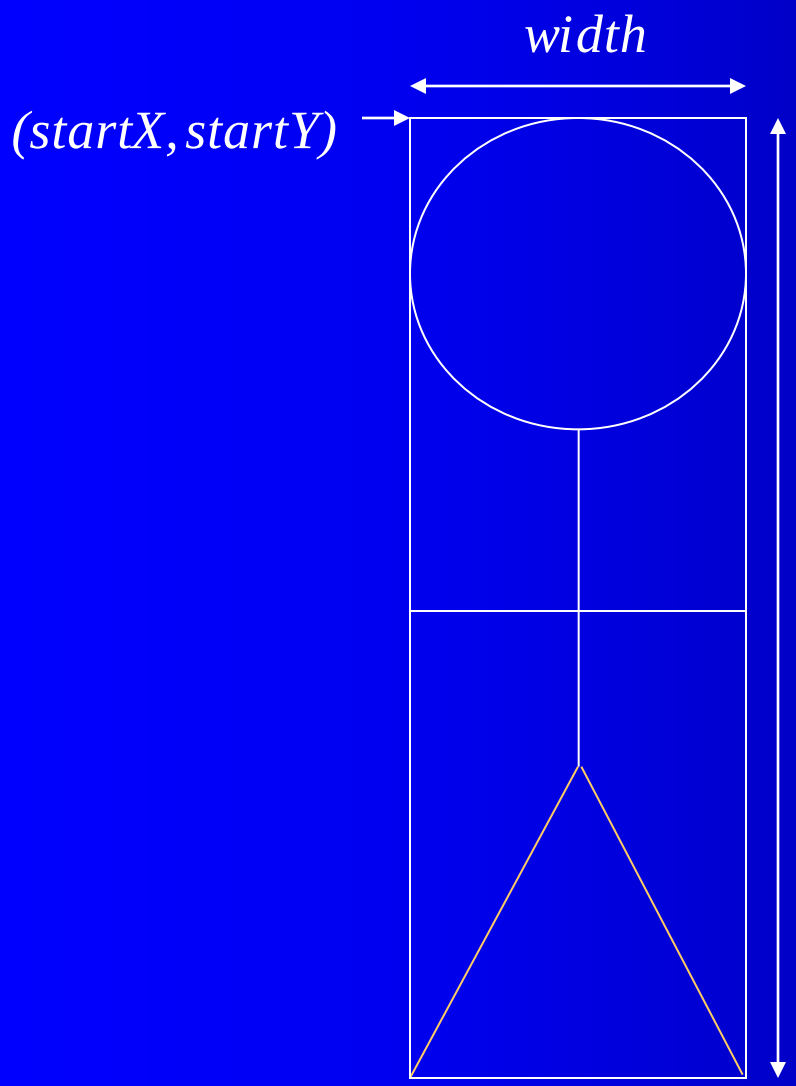

```
page.drawLine(
        startX + width/2,
        startY + height/3,
        startX + width/2,
        startY + height * 2/3);
```

Arms




```
page.drawLine(
        startX,
        startY + height/2,
        startX + width,
        startY + height/2);
```

Legs



page.drawLine(

page.drawLine(

Boy example code

```
import java.awt.*;
import java.applet.*;

public class Boy
{
    private int startX;
    private int startY;
    private int width;
    private int height;
    private Color color;

    public Boy(int x, int y, int w, int h, Color c)
    {
        startX = x;
        startY = y;
        width = w;
        height = h;
        color = c;
    }
```

Boy example code

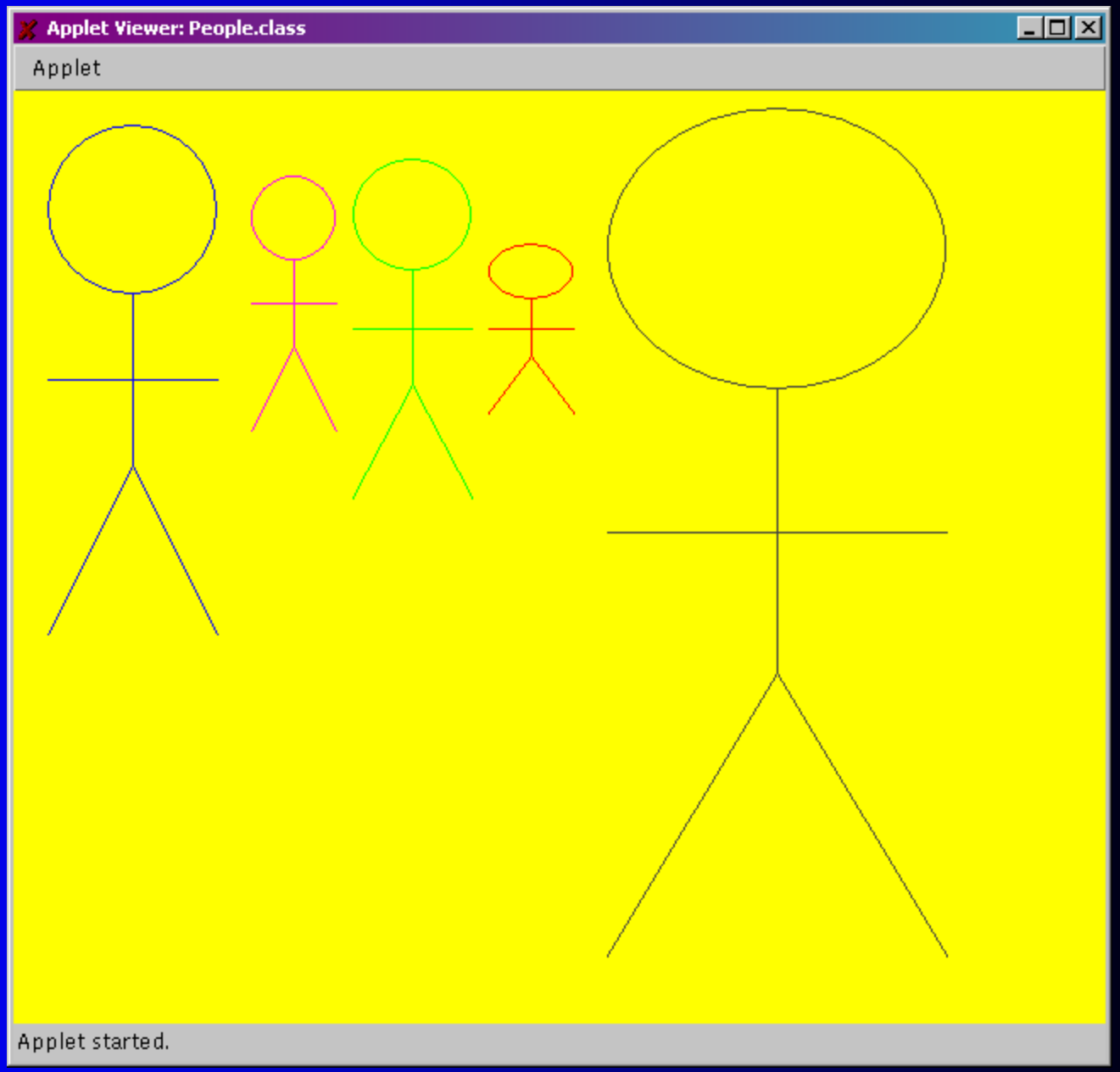
```
public void draw(Graphics page)
{
    page.setColor(color);

    page.drawOval(startX, startY, width, height/3);

    page.drawLine(startX + width/2, startY + height/3,
                  startX + width/2, startY + height * 2/3);

    page.drawLine(startX, startY + height/2,
                  startX + width, startY + height/2);

    page.drawLine(startX + width/2, startY + height * 2/3,
                  startX, startY + height);

    page.drawLine(startX + width/2, startY + height * 2/3,
                  startX + width, startY + height);
  }
}
```

Applet using the Boy class

```
import java.awt.*;
import java.applet.*;

public class People extends Applet
{
    private Boy b;

    public void init( )
    {
        setBackground(Color.orange);
        b = new Boy(20, 20, 100, 300, Color.blue);
    }

    public void paint(Graphics page)
    {
        b.draw(page);
    }
}
```

Boys example

- Change the People applet so that it prints out a number of Boys stored in an array



Boys example

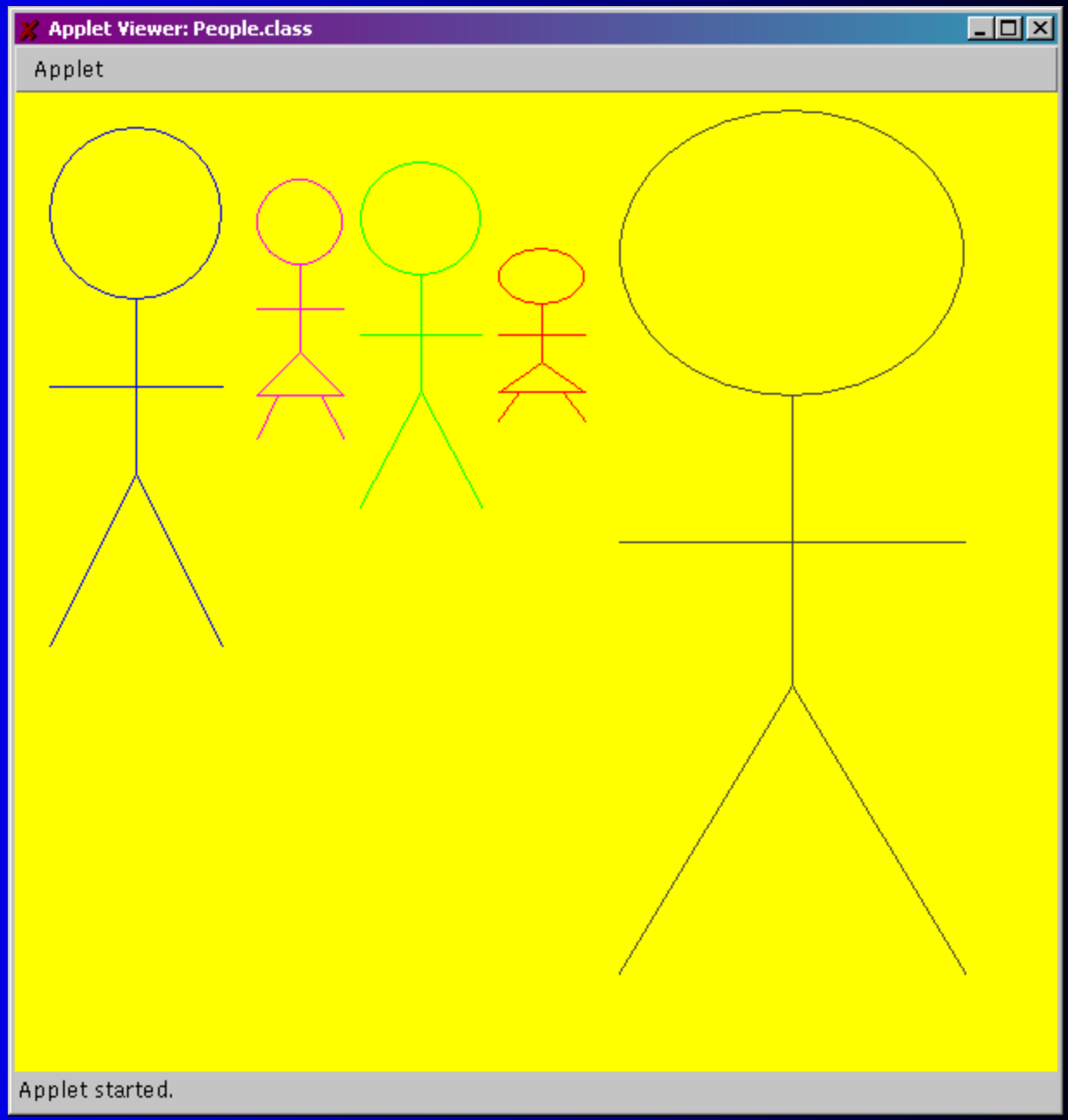
```
import java.awt.*;
import java.applet.*;

public class People extends Applet
{
    private Boy[ ] boys;

    public void init( )
    {
        setBackground(Color.yellow);
        boys = new Boy[5];
        boys[0] = new Boy( 20,  20, 100, 300, Color.blue);
        boys[1] = new Boy(140,  50,  50, 150, Color.magenta);
        boys[2] = new Boy(200,  40,  70, 200, Color.green);
        boys[3] = new Boy(280,  90,  50, 100, Color.red);
        boys[4] = new Boy(350,  10, 200, 500, Color.darkGray);
    }
```

Boys example

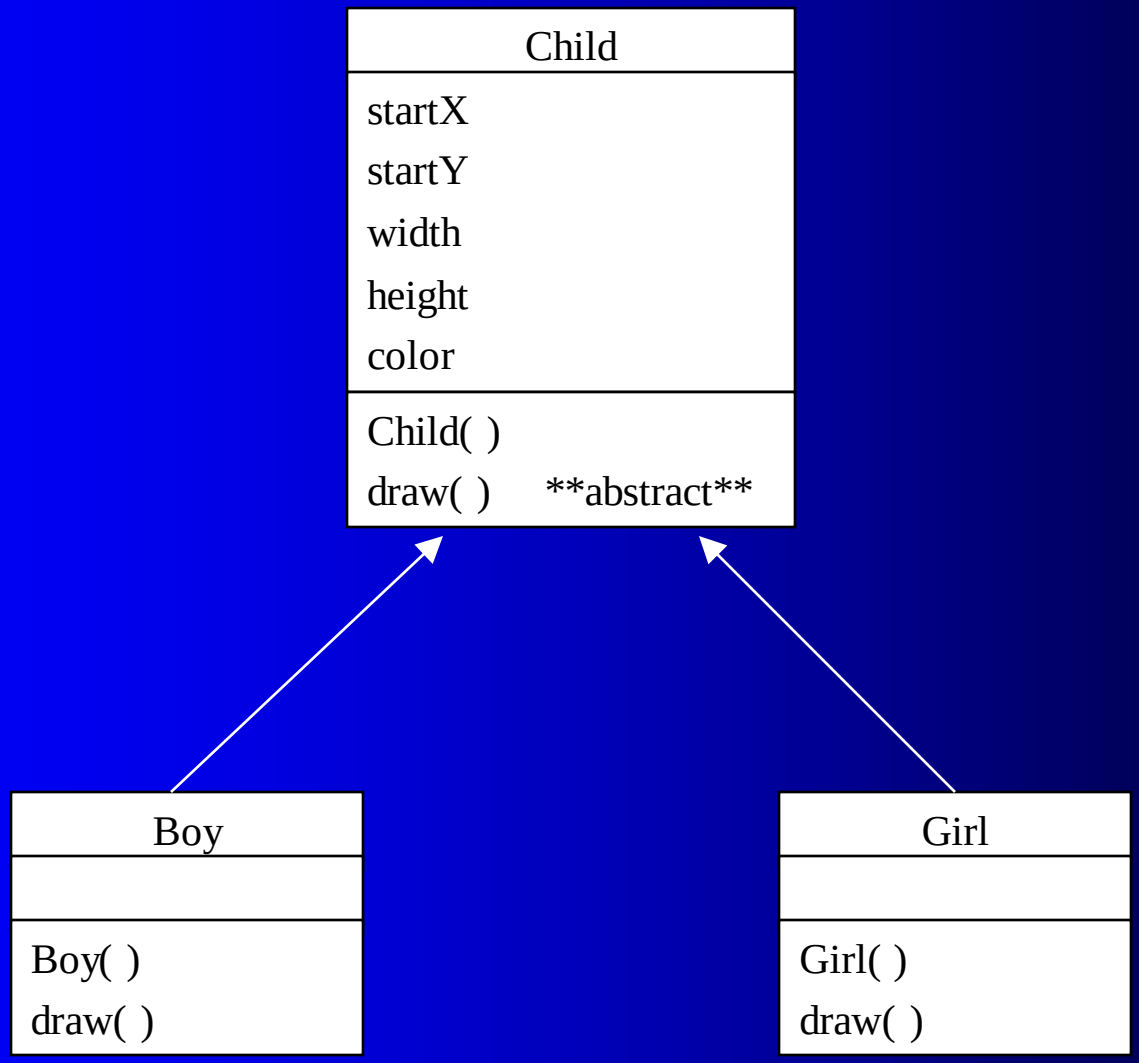
```
    public void paint(Graphics page)
    {
        for (int i = 0; i < boys.length; ++i)
        {
            boys[i].draw(page);
        }
    }
}
```

Children example

- Change the program so it displays both Boys and Girls held in an array of class Child



Model


Child
startX
startY
width
height
color
Child( )
draw( ) **abstract**
Boy
Boy( )
draw( )
Girl
Girl( )
draw( )

Children example – People class

```
import java.awt.*;
import java.applet.*;

public class People extends Applet
{
    private Child[ ] kids;

    public void init( )
    {
        setBackground(Color.yellow);
        kids = new Child[5];
        kids[0] = new Boy( 20,  20, 100, 300, Color.blue);
        kids[1] = new Girl(140,  50,  50, 150, Color.magenta);
        kids[2] = new Boy(200,  40,  70, 200, Color.green);
        kids[3] = new Girl(280,  90,  50, 100, Color.red);
        kids[4] = new Boy(350,  10, 200, 500, Color.darkGray);
    }
```

Children example – People class

```
    public void paint(Graphics page)
    {
        for (int i = 0; i < kids.length; ++i)
        {
            kids[i].draw(page);
        }
    }
}
```

Children example – Child class

```
import java.awt.*;
import java.applet.*;
public abstract class Child
{
    protected int startX;
    protected int startY;
    protected int width;
    protected int height;
    protected Color color;
    public Child(int x, int y, int w, int h, Color c)
    {
        startX = x;
        startY = y;
        width = w;
        height = h;
        color = c;
    }
    public abstract void draw(Graphics page);
}
```

Children example – Boy class

```
import java.awt.*;
import java.applet.*;

public class Boy extends Child
{
    public Boy(int x, int y, int w, int h, Color c)
    {
        super(x, y, w, h, c);
    }
```

Children example – Boy class

```
public void draw(Graphics page)
{
    page.setColor(color);

    page.drawOval(startX, startY, width, height/3);

    page.drawLine(startX + width/2, startY + height/3,
                  startX + width/2, startY + height * 2/3);

    page.drawLine(startX, startY + height/2,
                  startX + width, startY + height/2);

    page.drawLine(startX + width/2, startY + height * 2/3,
                  startX, startY + height);

    page.drawLine(startX + width/2, startY + height * 2/3,
                  startX + width, startY + height);
  }
}
```

Children example – Girl class

```
import java.awt.*;
import java.applet.*;

public class Girl extends Child
{
    public Girl(int x, int y, int w, int h, Color c)
    {
        super(x, y, w, h, c);
    }
```

Children example – Girl class

```
public void draw(Graphics page)
{
    page.setColor(color);
    page.drawOval(startX, startY, width, height/3);
    page.drawLine(startX + width/2, startY + height/3,
                  startX + width/2,   startY + height * 2/3);
    page.drawLine(startX, startY + height/2,
                  startX + width, startY + height/2);

    page.drawLine(startX + width/4, startY + height * 5/6,
                  startX, startY + height);
    page.drawLine(startX + width * 3/4, startY + height * 5/6,
                  startX + width, startY + height);

     page.drawLine(startX + width/2, startY + height * 2/3,
                  startX, startY + height * 5/6);
     page.drawLine(startX + width/2, startY + height * 2/3,
                   startX + width, startY + height * 5/6);
     page.drawLine(startX, startY + height * 5/6,
                   startX + width, startY + height * 5/6);
    }
}
```

Next lecture

- Event handling
- Mouse events
- Animation with the Timer class