

FUNDAMENTAL APPLET FOR GRAPHICS

APPLET

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

public class myprogs extends Applet
{
    public void init()
    {
        this.setSize(new Dimension(800,600));
        setBackground(Color.RED);
    }

    public void paint(Graphics g)
    {

    }

}
```

myprogs.java

SWING

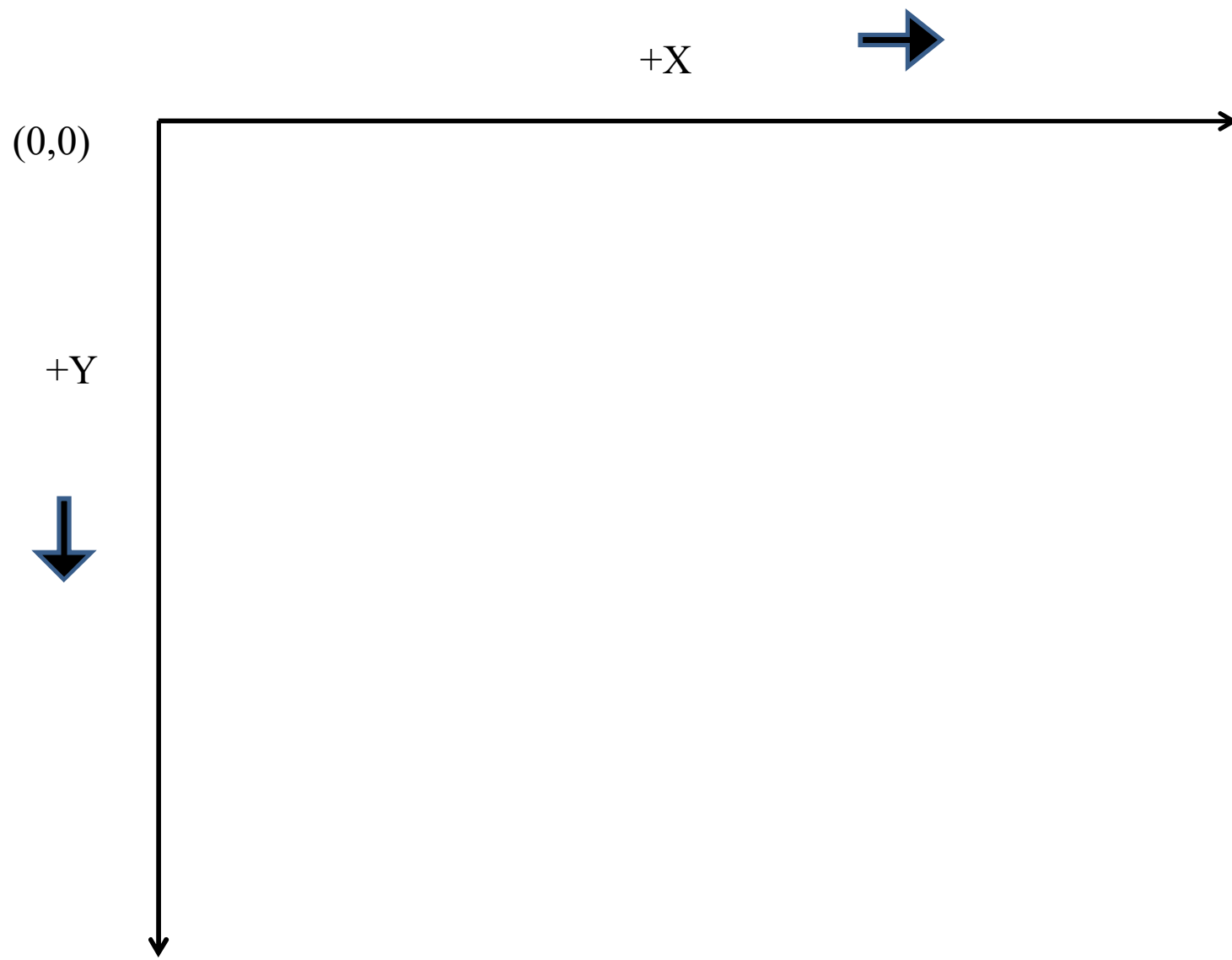
```
import java.awt.*;
import javax.swing.JFrame;

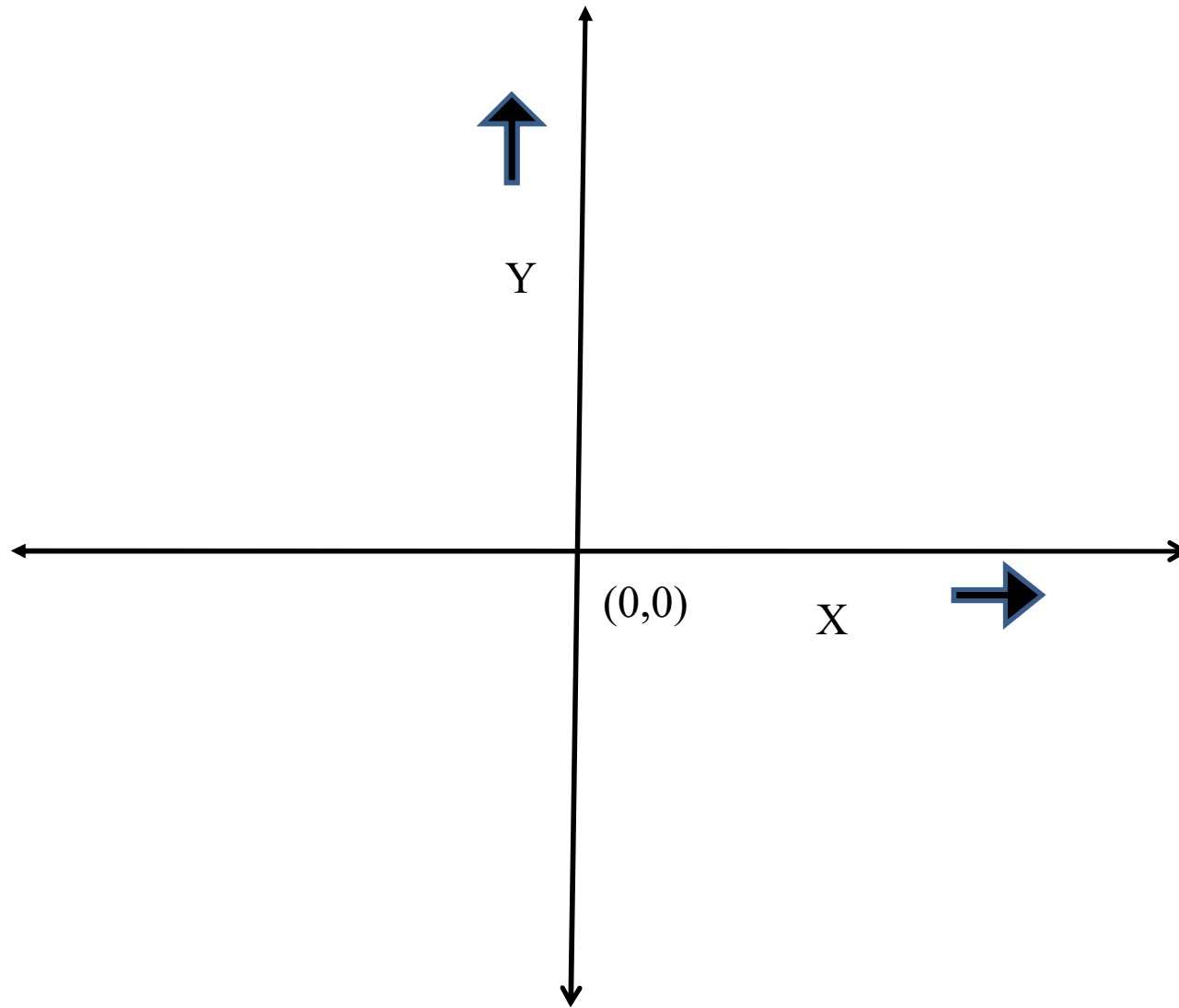
public class myprogs extends Canvas{

    public void paint(Graphics g)
    {
        setBackground(Color.RED);
    }

    public static void main(String[] args)
    {
        myprogs m=new myprogs();
        JFrame f=new JFrame();
        f.add(m);
        f.setSize(800,600);
        f.setVisible(true);
    }
}
```

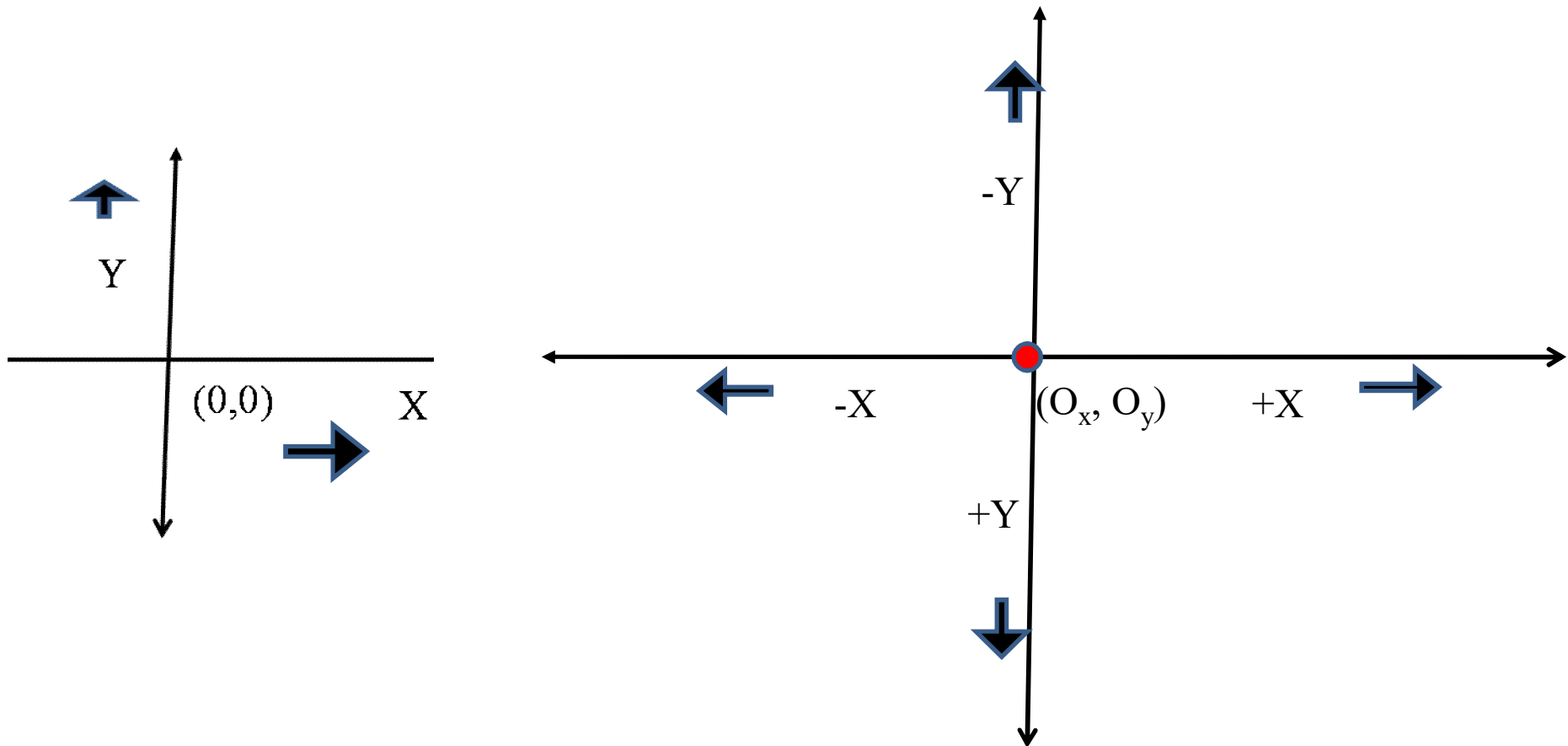
myprogs.java





To shift origin at the centre of the screen

```
originX=(getX()+getWidth())/2;  
originY=(getY()+getHeight())/2;
```



COLOUR

To set a particular colour

```
public void paint(Graphics g)
{
    g.setColor(Color.green);
}
```

To set a particular colour using colour code

```
Color mycolor=new Color(110,210,33);
g.setColor(mycolor);
```

To read current set colour

```
Color c=g.getColor();

int k1=c.getRed();
int k2=c.getGreen();
int k3=c.getBlue();
```

To draw a line

```
g.setColor(Color.green);  
g.drawLine(originX+x1,originY+y1,originX+x2,originY+y2);
```

To draw a point

```
g.setColor(Color.green);  
g.drawLine(originX+x1,originY+y1, originX+x1,originY+y1);
```

To draw a rectangle

```
g.drawRect(originX ,originY, width, height);
```


To draw a circle

```
g.drawOval(originX,originY,30,30);
```

To draw an oval

```
g.drawOval(originX,originY,80,30);
```

```
fillRect(int xTopLeft, int yTopLeft, int width, int height);
```

```
fillOval(int xTopLeft, int yTopLeft, int width, int height);
```

```
fillArc(int xTopLeft, int yTopLeft, int width, int height,  
int startAngle, int arcAngle);
```

```
fill3DRect(int xTopLeft, int, yTopLeft, int width, int height,  
boolean raised);
```

```
fillRoundRect(int xTopLeft, int yTopLeft, int width, int height,  
int arcWidth, int arcHeight);
```

```
fillPolygon(int[] xPoints, int[] yPoints, int numPoint);
```

To write text

```
g.drawString("Apple",originX ,originY);
```

To write text with varying font size and colour

```
Font stringFont = new Font( "Times New Roman",4, 18 );  
g.setFont( stringFont );  
g.setColor(Color.RED);  
g.drawString("Apple",50 ,50);
```

```
Font stringFont = new Font( "Times New Roman",4, 18 );
```

```
1-----BOLD
```

```
2-----ITALIC
```

```
3-----BOLD ITALIC
```

```
4-----REGULAR
```

MOUSE INTERACTIVE APPLET

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

public class myprogs extends Applet implements
MouseListener, MouseMotionListener
{

    public void init();
    public void mouseEntered(MouseEvent m);
    public void mouseExited(MouseEvent m);
    public void mousePressed(MouseEvent m);
    public void mouseReleased(MouseEvent m);
    public void mouseMoved(MouseEvent m);
    public void mouseDragged(MouseEvent m);
    public void paint(Graphics g);

}
```

```
public void init()  
{  
    addMouseListener(this);  
    addMouseMotionListener(this);  
}
```

```
public void mouseEntered(MouseEvent m)  
{  
    //...YOUR ACTION  
    repaint();  
}
```

```
public void mouseExited(MouseEvent m)  
{  
    //...YOUR ACTION  
    repaint();  
}
```

```
public void mousePressed(MouseEvent m)
{
    int x = (int) m.getX();
    int y = (int) m.getY();

    switch(m.getModifiers())
    {
        case InputEvent.BUTTON1_MASK:
        {
            System.out.println("That's the LEFT button");
            break;
        }
        case InputEvent.BUTTON2_MASK:
        {
            System.out.println("That's the MIDDLE button");
            break;
        }
        case InputEvent.BUTTON3_MASK:
        {
            System.out.println("That's the RIGHT button");
            break;
        }
    }
    repaint();
}
```

KEYBOARD INTERACTIVE APPLET


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

public class kbp extends Applet implements KeyListener
{

    public void init();
    public void keyPressed(KeyEvent k);
    public void keyReleased(KeyEvent k);
    public void keyTyped(KeyEvent k);
    public void paint(Graphics g);
}
```

```
public void init()  
{  
addKeyListener(this);  
requestFocus();  
}
```

```
public void keyReleased(KeyEvent k)  
{  
System.out.println("Key Up");  
}
```

```
public void keyTyped(KeyEvent k)  
{  
System.out.println("You have typed"+": "+k.getKeyChar());  
repaint();  
}
```

```
public void keyPressed(KeyEvent k)
{
    System.out.println("KeyDown");
    int key=k.getKeyCode();
    switch(key)
    {
        case KeyEvent.VK_UP:
            System.out.println("Move to Up");
            break;

        case KeyEvent.VK_DOWN:
            System.out.println("Move to Down");
            break;

        case KeyEvent.VK_LEFT:
            System.out.println("Move to Left");
            break;

        case KeyEvent.VK_RIGHT:
            System.out.println("Move to Right");
            break;
    }
    repaint();
}
```

```
public void paint(Graphics g)
{

}
```

**CREATING COMPLICATED SHAPE AS AN
OBJECT**

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

public class newshape
{
    private int x1,x2,h,w;
    Graphics g1;

    newshape(int a1,int a2,int a3,int a4,Graphics g)
    {
        g1=g; x1=a1; x2=a2; h=a3; w=a4;
    }
    public void setVisible()
    {
    }
    public void setInvisible()
    {
    }

    public void color_change(Color c)
    {
    }
}
```

```
public void paint(Graphics g)
{
    newshape n=new newshape(10,10,50,50,g);
    n.setVisible();
}
```

1D array

```
newshape n[]=new newshape [3];
n[0]=new newshape(10,10,50,50,g);
n[0].setVisible();
```

2D array

```
newshape n[][]=new newshape [3][5];
n[0][0] =new newshape(10,10,50,50,g);
n[0][0].setVisible();
```

INTRODUCING DELAY


```
public void paint(Graphics g)
{
    try
    {
        Thread.sleep(300); //millisecond

        //...YOUR LOGIC
    }

    catch (InterruptedException ie)
    {
        ie.printStackTrace();
    }
}
```

TAKING INPUT (using Textfield)

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

```
String temp ="45";  
textField1.setText(temp);
```

```
TextField textField1;
```

```
textField1 = new TextField("Hello");  
add(textField1);
```

```
String temp =textField1.getText();  
System.out.println(temp);
```

PLACING BUTTON

(using Button)

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

public class AnAppletWithButtons extends Applet
implements ActionListener {
    public void init() {
        button1 = new Button("Button 1");
        add(button1);
        button1.addActionListener(this);

        button2 = new Button("Button 2");
        add(button2);
        button2.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e)
    {
        if (e.getSource() == button1)
            System.out.println("Button 1 was pressed");
        else
            System.out.println("Button 2 was pressed");
    }

    Button button1, button2;
}
```

ADDING IMAGE

```
Image img;  
MediaTracker tr;  
  
public void init()  
{  
    img=getImage(getCodeBase(),"myimage.jpg");  
    tr=new MediaTracker(this);  
}  
  
public void paint(Graphics g)  
{  
    tr.addImage(img,0);  
    g.drawImage(img, 0, 0, this);  
}
```

END OF CHAPTER