

```
import java.awt.*;

public class Comet extends Planet
{
    public static final double G = .03;
    public static final double SPEED = .333;
    double xV;
    double yV;
    boolean random = false;
    Color color = null;

    public Comet(double radius, double x, double y)
    {
        super(radius, x, y);
    }

    public Comet()
    {
        super(3, 320, 240);
        double t = Math.random()*2*Math.PI;
        double mag = Math.random()*6;
        x = mag*Math.cos(t);
        y = mag*Math.sin(t);
        random = true;
    }

    public void decelerate()
    {
        xV *= .99;
        yV *= .99;
    }

    public double getMass()
    {
        return (4.0/3.0)*radius*radius*radius*Math.PI;
    }

    public double getDistance(Planet p)
    {
        return Math.pow(Math.pow(p.x-x,2) + Math.pow(p.y-y,2), .5);
    }

    public void move(Planet p)
    {
        if(!contains(p))
        {
            double acceleration = G*p.getMass() / (Math.pow(p.x-x,2) + Math.pow(p.y-y,2));
            xV += acceleration*(p.x-x)/(getDistance(p));
            yV += acceleration*(p.y-y)/(getDistance(p));
        }
    }

    public void move(Direction d)
    {
        xV+= SPEED*d.x;
        yV+= SPEED*d.y;
    }

    public void update()
    {
        x+=xV;
        y+=yV;

        if(Math.sqrt((x-320)*(x-320)+(y-240)*(y-240)) > 1000)
```

```
{
    xV = 0;
    yV = 0;
    double dist = Math.sqrt((x-320)*(x-320)+(y-240)*(y-240));
    double xComp = x-320;
    double yComp = y-240;
    x = (950*xComp/dist)+320;
    y = (950*yComp/dist)+240;
}

public void draw(Graphics g)
{
    if(color == null)
        g.setColor(Color.BLUE);
    else
        g.setColor(color);
    g.fillOval((int)(x-radius+.5), (int)(y-radius+.5), (int)(radius*2+.5),
        (int)(radius*2+.5));
    if(((x > 640) || (y > 480) || (x < 0) || (y < 0)) && !random)
    {
        double dist = Math.sqrt((x-320)*(x-320)+(y-240)*(y-240));
        double xComp = x-320;
        double yComp = y-240;
        g.drawLine((int)(240*(xComp/dist))+320, (int)(240*(yComp/dist))+
            240, (int)(200*(xComp/dist))+320, (int)(200*(yComp/dist))+240);
    }
}
}
```