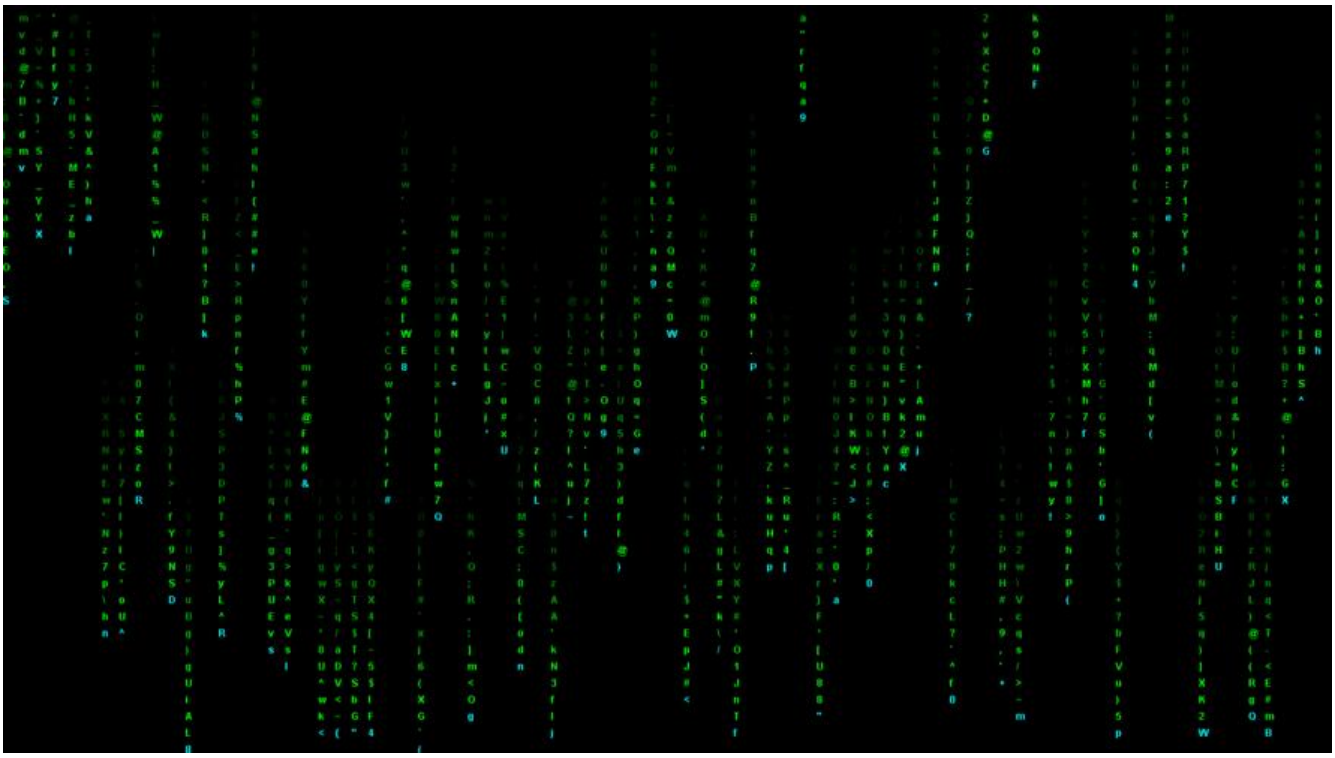
java 代码实现《黑客帝国》酷炫数字雨效果

作者：Stone
原文链接：https://ld246.com/article/1472700604290
来源网站：链滴

直接上效果图：



目前的样子还可以美化，期待其他童鞋效果图0.0

下边是代码

```
package cn.com.entity;

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Cursor;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.Image;
import java.awt.Point;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.awt.image.MemoryImageSource;
import java.util.Random;

import javax.swing.JDialog;
import javax.swing.JPanel;
import javax.swing.Timer;

public class Rain extends JDialog implements ActionListener {

    private static final long serialVersionUID = 1549334578038217282L;
```

```
    private Random random = new Random();
    private Dimension screenSize;
    private JPanel graphicsPanel;
    // 行高,列宽
    private final static int gap = 20;
    // 存放雨点顶部的位置信息(marginTop)
    private int[] posArr;
    // 行数
    private int lines;
    // 列数
    private int columns;

    public Rain() {
        initComponents();
    }

    private void initComponents() {
        setLayout(new BorderLayout());
        graphicsPanel = new GraphicsPanel();
        add(graphicsPanel, BorderLayout.CENTER);
        // 设置光标不可见
        Toolkit defaultToolkit = Toolkit.getDefaultToolkit();
        Image image = defaultToolkit.createImage(new MemoryImageSource(0, 0, null, 0, 0));
        Cursor invisibleCursor = defaultToolkit.createCustomCursor(image, new Point(0, 0), "curs
r");
        setCursor(invisibleCursor);
        // ESC键退出
        KeyPressListener keyPressListener = new KeyPressListener();
        this.addKeyListener(keyPressListener);
        // this.setAlwaysOnTop(true);
        // 去标题栏
        this.setUndecorated(true);
        // 全屏
        this.getGraphicsConfiguration().getDevice().setFullScreenWindow(this);
        this.setDefaultCloseOperation(JDialog.DISPOSE_ON_CLOSE);
        setVisible(true);

        screenSize = Toolkit.getDefaultToolkit().getScreenSize();
        lines = screenSize.height / gap;
        columns = screenSize.width / gap;

        posArr = new int[columns + 1];
        random = new Random();
        for (int i = 0; i < posArr.length; i++) {
            posArr[i] = random.nextInt(lines);
        }

        // 每秒10帧
        new Timer(100, this).start();
    }

    /**
     * @return 随机字符
```

```
 */
private char getChr() {
    return (char) (random.nextInt(94) + 33);
}

@Override
public void actionPerformed(ActionEvent e) {
    graphicsPanel.repaint();
}

private class GraphicsPanel extends JPanel {
    private static final long serialVersionUID = 3043939524168375053L;

    @Override
    public void paint(Graphics g) {
        Graphics2D g2d = (Graphics2D) g;
        g2d.setFont(getFont().deriveFont(Font.BOLD));
        g2d.setColor(Color.BLACK);
        g2d.fillRect(0, 0, screenSize.width, screenSize.height);
        // 当前列
        int currentColumn = 0;
        for (int x = 0; x < screenSize.width; x += gap) {
            int endPos = posArr[currentColumn];
            g2d.setColor(Color.CYAN);
            g2d.drawString(String.valueOf(getChr()), x, endPos * gap);
            int cg = 0;
            for (int j = endPos - 15; j < endPos; j++) {
                // 颜色渐变
                cg += 20;
                if (cg > 255) {
                    cg = 255;
                }
                g2d.setColor(new Color(0, cg, 0));
                g2d.drawString(String.valueOf(getChr()), x, j * gap);
            }
            // 每放完一帧，当前列上雨点的位置随机下移1~5行
            posArr[currentColumn] += random.nextInt(5);
            // 当雨点位置超过屏幕高度时，重新产生一个随机位置
            if (posArr[currentColumn] * gap > getHeight()) {
                posArr[currentColumn] = random.nextInt(lines);
            }
            currentColumn++;
        }
    }
}

private class KeyPressListener extends KeyAdapter {
    @Override
    public void keyPressed(KeyEvent e) {
        if (e.getKeyCode() == KeyEvent.VK_ESCAPE) {
            System.exit(0);
        }
    }
}
```

```
    public static void main(String[] args) {
        new Rain();
    }

}
```

这是html版的

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<style type="text/css">
canvas {
    display: block;
}
</style>
</head>
<body>
    <canvas id="canvas"></canvas>
    <script>
        function $(id) {
            return document.getElementById(id);
        }
        //画布 的大小设置
        var mywindow = window.screen;
        var canvas = $("canvas");
        canvas.width = mywindow.width;
        canvas.height = mywindow.height;
        //要输出的信息
        var str = "0123456789";
        str = str.split("");
        var fontSize = 16;//输出字体的大小
        //每列显示多少个信息
        var cols = canvas.width / fontSize;
        //数组，统计下落的位置
        var drops = [];
        for (var i = 0; i < cols; i++) {
            drops[i] = 1;
        }
        var ctx = canvas.getContext("2d");
        function draw() {
            /**
             *这的黑客帝国：
             * 获取页面的 大小 包括宽度和高度
             * 用cols获取能够宽度加载列数
             * drops加载每一列的位置
             * drops[2]=10 2为第二排的 top为10
             */
            ctx.fillStyle = "rgba(0,0,0,0.05)";
            ctx.fillRect(0, 0, canvas.width, canvas.height);
            ctx.fillStyle = "green";
```

```
        ctx.font = fontSize + "px arial";
        for (var i = 0; i < cols; i++) {
            var text = str[Math.floor(Math.random() * (str.length))];
            // console.info("x-"+i*fontSize);
            // console.info("y-"+drops[i]*fontSize);
            ctx.fillText(text, i * fontSize, drops[i] * fontSize);
            if (drops[i] * fontSize > canvas.height || Math.random() > 0.95)
                drops[i] = 0;//把位置恢复到最上面
                //控制下落的位置
            drops[i]++;
        }
    }
    setInterval(draw, 33);
  </script>
</body>
</html>
```