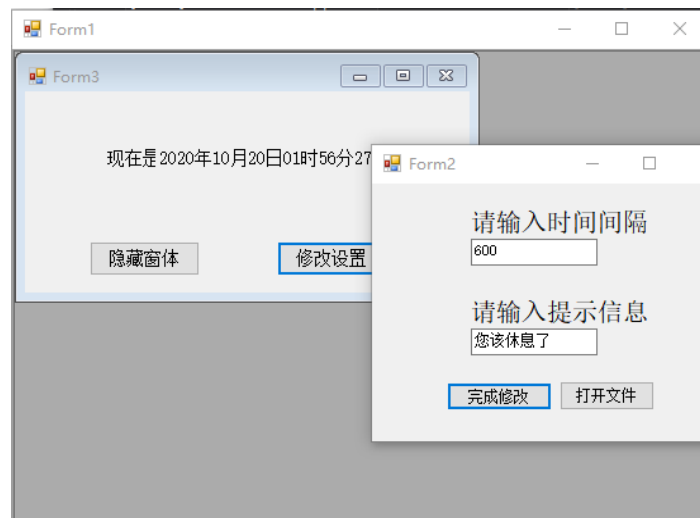
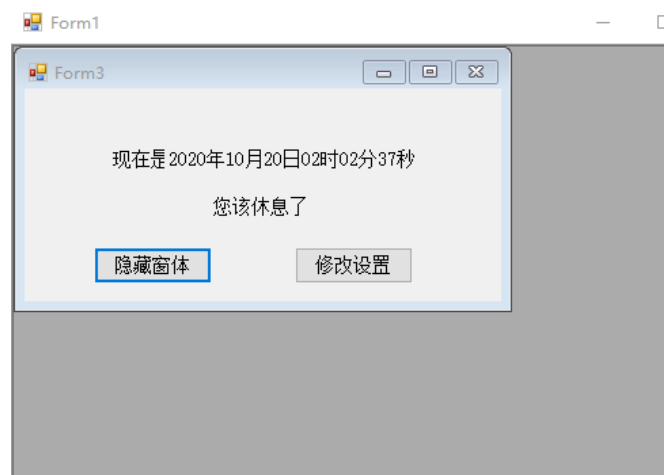


一、 编一个定时提醒的程序，每隔 x 秒就显示提示信息（如：“您该休息了”）。

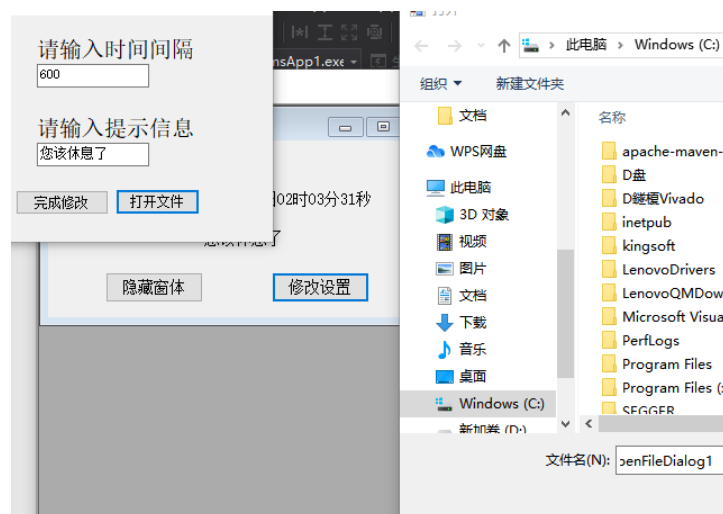
1.点击修改设置，跳出修改设置框



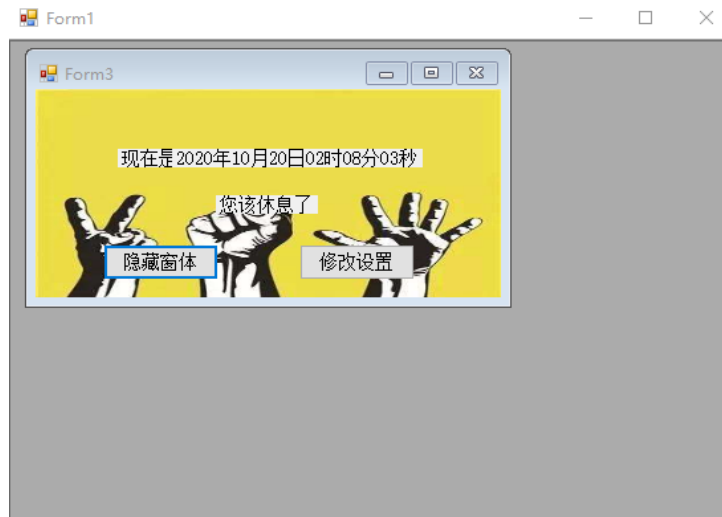
2.点击完成修改，开始倒计时，到点后会提示“您该休息了”




3.打开文件功能

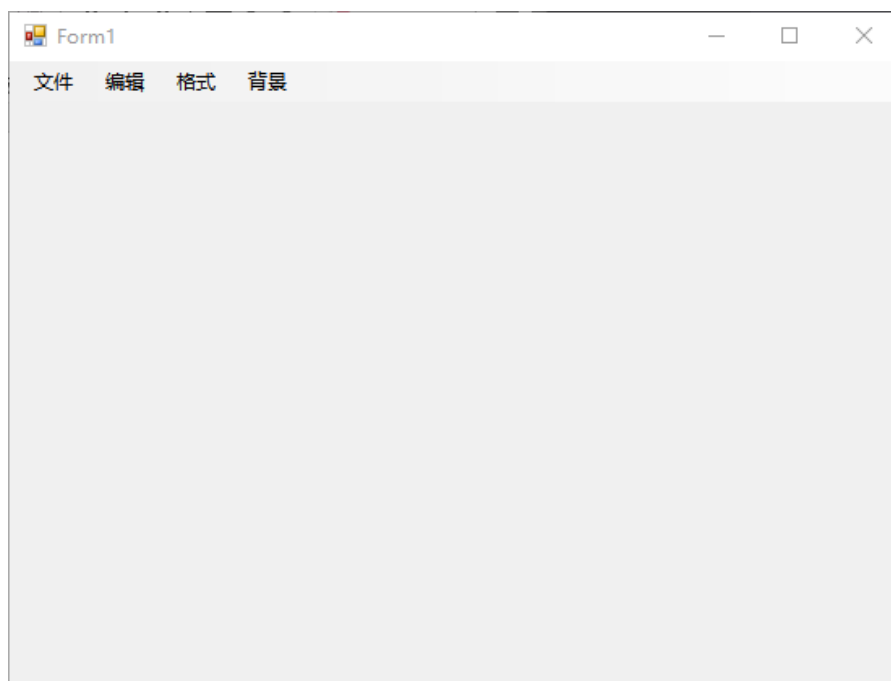


4.更换背景，选择完图片，就可以完成背景更换

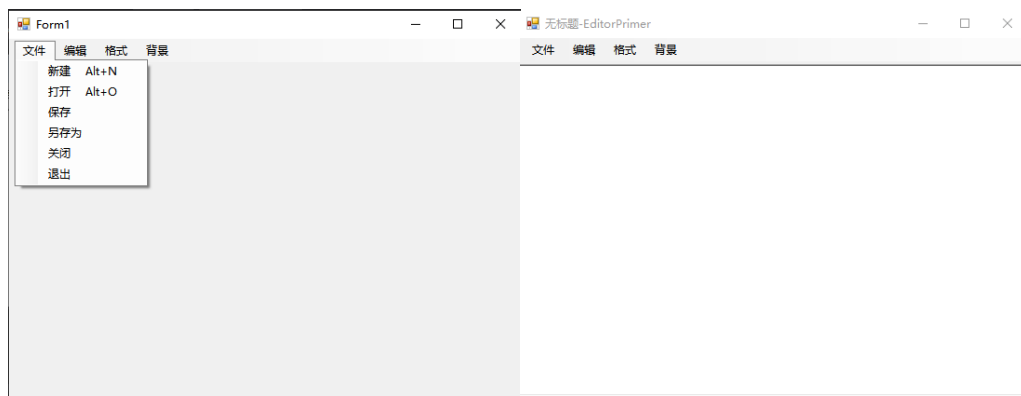


二、实现一个多文档记事本程序，对富文本内容进行编辑。记事本功能：

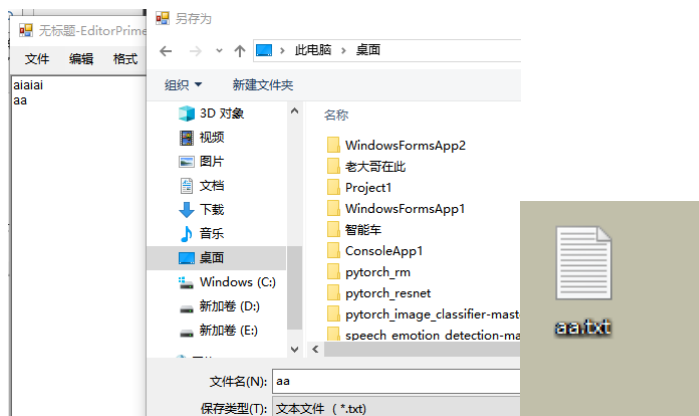
1. 点击 ，得到记事本界面，菜单栏有文件，编辑，格式，背景四个栏目
此时不能在白色区域进行编辑打字（如下图）



2. 点击文件，测试新建功能，点击新建后，可以在富文本框中输入编辑（如下图）

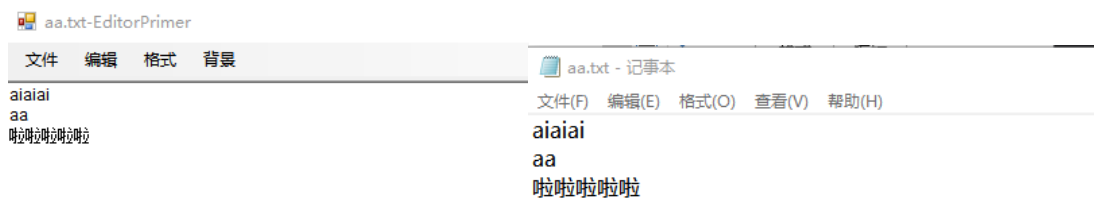


3.随意输入几个字符后，点击另存为功能，弹出另存为界面，此处我保存到了桌面，文件名为 aa.txt（如下图），另存为功能成功



4.测试保存功能，我们随意修改内容，然后点击保存按钮，查看是否保存了，结果确实保存了，保存功能成功（如下图）

若没有进行刚刚的另存为，点击保存按钮，会自动调用另存为按钮的函数，跳出另存为界面

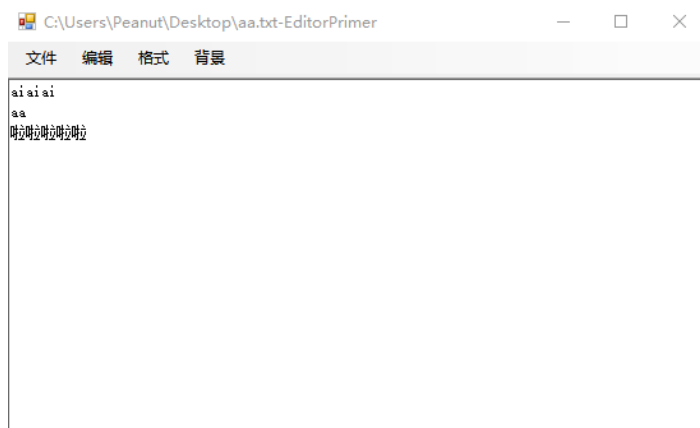
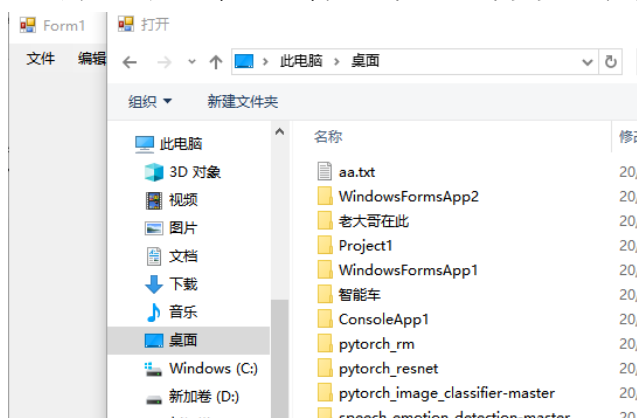


5.测试关闭按钮，点击关闭，首先进行保存；然后富文本框隐藏，无法进行输入

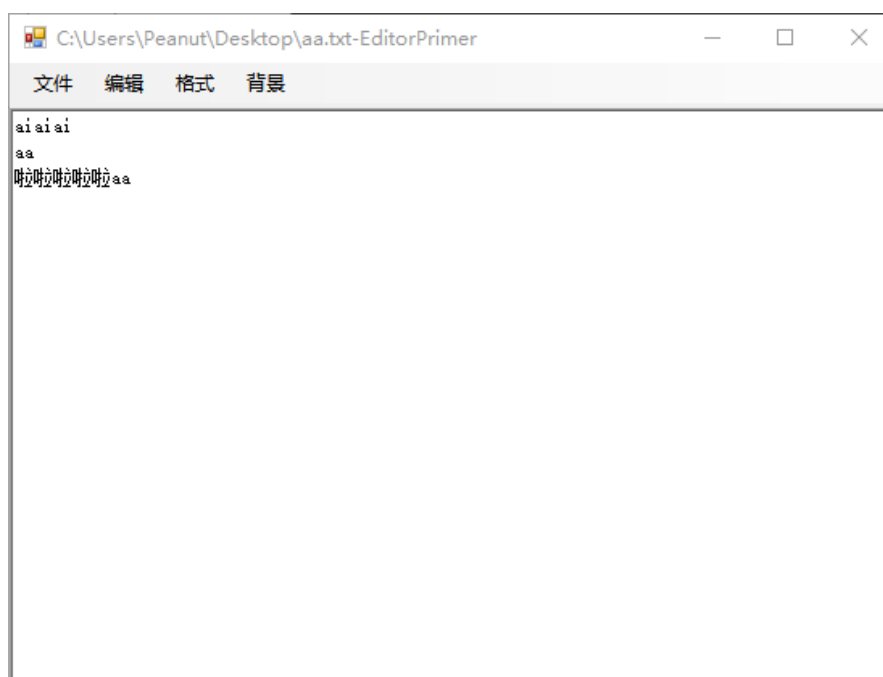
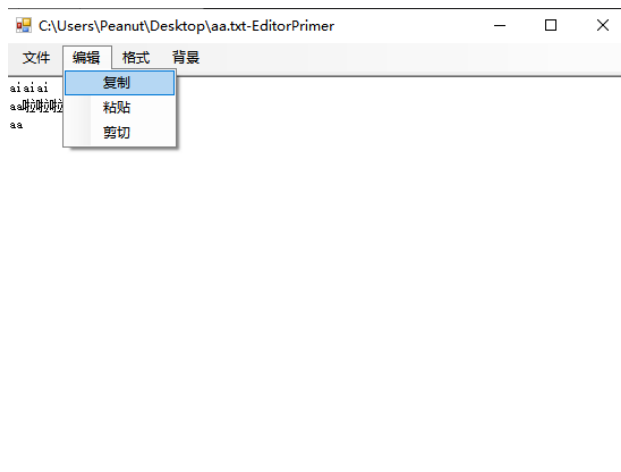


6.测试退出功能，点击退出，首先将文件保存，然后退出程序

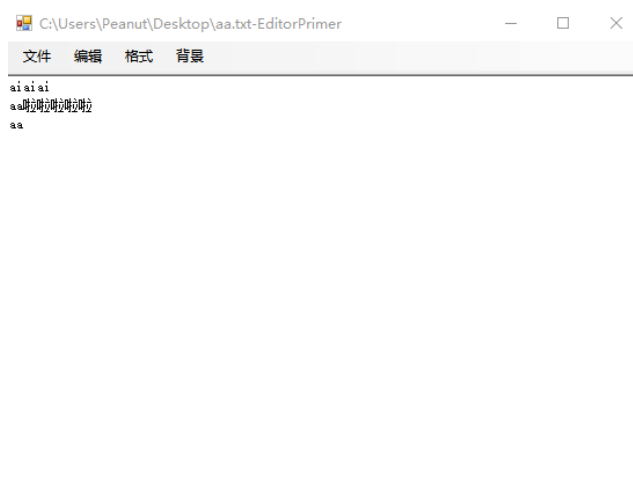
7.测试打开功能（如下图），选择刚刚关闭的文件，得刚刚所输入的结果



8.测试复制粘贴功能（如下图），选住所要复制的内容，点击编辑-复制，然后在想粘贴的地方，点击编辑-粘贴，



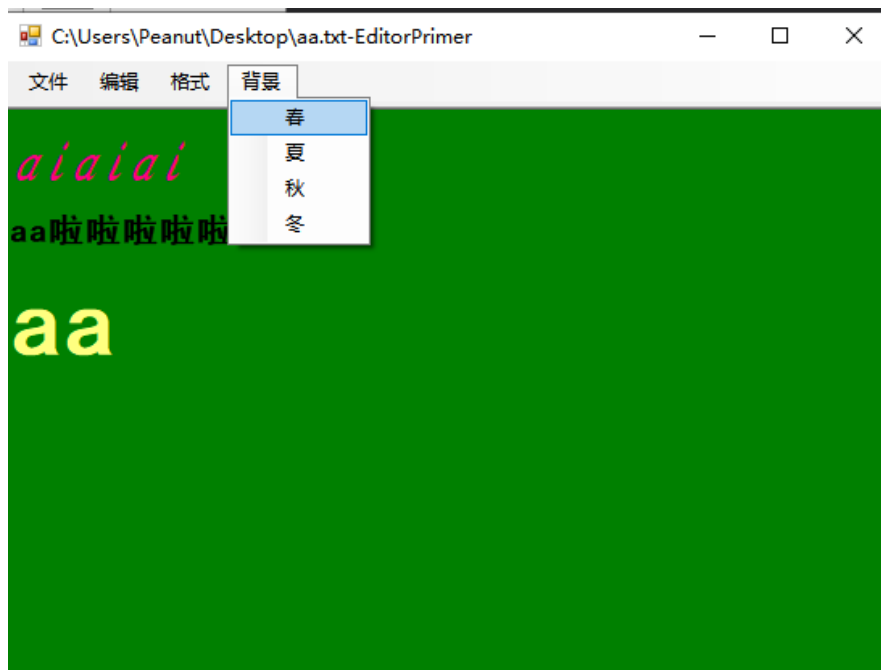
9. 测试剪切粘贴功能（如下图），选住所要剪切的内容，点击编辑-剪切，然后在想粘贴的地方，点击编辑-粘贴，



10. 测试格式功能（效果如下），选住想要编辑的文字，点击字体，颜色



11.测试背景功能（如下）春为绿色，夏为红，秋为黄，冬为浅灰







源代码:

1.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e) //完成
修改
        {
            //点击此选项
            this.Close();
        }
    }
}
```



```

    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void Form2_Load(object sender, EventArgs e)
    {

    }
    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void label4_Click(object sender, EventArgs e)
    {

    }

    private void openFileDialog1_FileOk(object sender,
CancelEventArgs e)
    {

    }
    string filename = ""; //定义并初始化下文的文件名
    private void button2_Click(object sender, EventArgs e)
    {

        openFileDialog1.FilterIndex = 1; //设
置文件对话框中当前选中文件筛选器的索引
        openFileDialog1.InitialDirectory = "C:\\"; //设
置文件对话框显示的初始目录
        if (openFileDialog1.ShowDialog() ==
System.Windows.Forms.DialogResult.OK)
        {
            //如果点击了确定按钮,更新文件名并向丰富区装载相应文本
            filename = openFileDialog1.FileName;
            Form1.BackgroundImage =
Image.FromFile(openFileDialog1.FileName);
        }
    }
}

```

```

        private void textBox2_TextChanged(object sender, EventArgs e)
        {

        }

    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }

        private void label2_Click(object sender, EventArgs e)
        {

        }

        private void button2_Click(object sender, EventArgs e)//修改
        {
            Form2 a = new Form2();

            a.Show();
        }

        private void label3_Click(object sender, EventArgs e)
        {

        }
    }
}

```

设置

```

        private void timer1_Tick(object sender, EventArgs e)
        {
            label3.Text = DateTime.Now.ToString("yyyy年MM月dd日hh时mm
分ss秒");
        }

        private void button1_Click(object sender, EventArgs e)
        {

            this.Hide();
        }
    }
}

```

2.

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp2
{
    public partial class Form1 : Form
    {
        string filename = "";           //定义并初始化下文的文件名
        public Form1()
        {
            InitializeComponent();
        }

        private void 打开Alt0ToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            richTextBox1.Visible = true;
            //打开文件选项, 调用openFile对话框
            openFileDialog1.Filter = "文本文件 | *.txt";           //设置文件
类型过滤器
            openFileDialog1.FilterIndex = 1;                       //设置文件
对话框中当前选中文件筛选器的索引

```

```

        openFileDialog1.InitialDirectory = "E:\\";           //设置文件
对话框显示的初始目录
        if (openFileDialog1.ShowDialog() ==
System.Windows.Forms.DialogResult.OK)
        {
            //如果点击了确定按钮,更新文件名并向丰富区装载相应文本
            filename = openFileDialog1.FileName;
            //参数: 需要装载的文件名以及装载的文本格式, 这里指定纯文本
            richTextBox1.LoadFile(filename,
RichTextBoxStreamType.PlainText);
            this.Text = filename + "-EditorPrimer";         //更新窗口
标题栏
        }
    }

    private void 保存ToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        //保存选项实现代码
        if (filename.Length > 0)
        {
            //文件名不是空, 原来已经有此文件则直接保存
            richTextBox1.SaveFile(filename,
RichTextBoxStreamType.PlainText);
        }
        else
        {
            //文件名是空, 属于新建的文件, 调用另存为事件
            新建AltNToolStripMenuItem_Click(sender, e);
        }
    }

    private void 另存为ToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        //另存为选项, 代码与打开文件选项类似
        saveFileDialog1.Filter = "文本文件 | *.txt";         //设置
保存文件类型过滤器
        saveFileDialog1.FilterIndex = 1;                     //文件
对话框中当前选定筛选器的索引
        saveFileDialog1.InitialDirectory = "c:\\";           //保存
文件时的默认目录
        if (saveFileDialog1.ShowDialog() ==
System.Windows.Forms.DialogResult.OK)

```

```

        {
            //点击了保存文件对话框的确定按钮
            filename = saveFileDialog1.FileName; //更新
文件名
            //保存文件, 参数分别是文件的路径path 和文件的类型(这里指定为
纯文本)
            richTextBox1.SaveFile(filename,
RichTextBoxStreamType.PlainText);
            //获取文件保存路径并更新窗口标题栏
            int index = filename.LastIndexOf('\\'); //找到
最后一个斜线的下标索引
            string Text_2 = filename.Substring(index + 1); //去掉
文件路径获取文件名(参数下标索引, 截取长度(不指定直到最后))
            this.Text = Text_2 + "-EditorPrimer"; //更新
标题栏
        }
    }

    private void 新建AltNToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        //新建选项
        richTextBox1.Clear(); //清空丰富文本区
        richTextBox1.Visible = true;
        filename = ""; //开始的文件名为空
        this.Text = "无标题-EditorPrimer"; //初始化文件标题
    }

    private void 关闭ToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        //点击此选项, 保存然后关闭窗口
        保存ToolStripMenuItem_Click(sender, e);
        this.Close();
    }

    private void 退出ToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        //点击此选项, 关闭窗口并退出程序
        this.Close();
        Application.Exit();
    }
}

```

```

private void richTextBox1_TextChanged(object sender, EventArgs e)
{

}

private void openFileDialog1_FileOk(object sender, CancelEventArgs
e)
{

}

private void saveFileDialog1_FileOk(object sender, CancelEventArgs
e)
{

}

private void 复制ToolStripMenuItem_Click(object sender, EventArgs
e)
{
    this.richTextBox1.Copy();

}

private void 粘贴ToolStripMenuItem_Click(object sender, EventArgs
e)
{
    this.richTextBox1.Paste();
}

private void 剪切ToolStripMenuItem_Click(object sender, EventArgs
e)
{
    this.richTextBox1.Cut();
}

private void 字体ToolStripMenuItem_Click(object sender, EventArgs
e)
{
    //修改字体选项响应事件
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        //颜色对话框弹出成功, 修改选中文本的颜色

```

```
        richTextBox1.SelectionFont = fontDialog1.Font;
    }
}

e) private void 颜色ToolStripMenuItem_Click(object sender, EventArgs
{
    //修改颜色选项响应事件
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        //颜色对话框弹出成功，修改选中文本的颜色
        richTextBox1.SelectionColor = colorDialog1.Color;
    }
}

private void 春ToolStripMenuItem_Click(object sender, EventArgs e)
{
    richTextBox1.BackColor = Color.Green;
}

private void 夏ToolStripMenuItem_Click(object sender, EventArgs e)
{
    richTextBox1.BackColor = Color.Red;
}

private void 秋ToolStripMenuItem_Click(object sender, EventArgs e)
{
    richTextBox1.BackColor = Color.Yellow;
}

private void 冬ToolStripMenuItem_Click(object sender, EventArgs e)
{
    richTextBox1.BackColor = Color.LightGray;
}
}
```