

Answer on Question #38691, Programming, C#

```
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;
using System.IO;

namespace Task75112//Question #38691
{
    public partial class Form1 : Form
    {
        string urgent = "";
        bool urg = false;

        public Data_Grid products;
        string pass = @"D:\"; //Pass to the file
        List<string> product_name= new List<string>();
        List<int> stock_level = new List<int>();
        List<double> product_price = new List<double>();
        List<string> cell_history = new List<string>();

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            comboBox1.DropDownStyle = ComboBoxStyle.DropDownList;
            products = new Data_Grid(140, "Part Name", "Quantity", "Price","Date");
            products.Columns[1].Width = 80;
            products.Columns[2].Width = 80;
            products.Columns[3].Width = 80;
            products.Location = new Point(20, 20);
            products.Parent = tabControl1.TabPages[0];
            products.Size = new Size(433, 300);
            prodIniz();
            orderIniz();

            // MessageBox.Show(m);
            //Byte[] info = new UTF8Encoding(true).GetBytes(Data); //Convert your data to
Bytes
            // fs.Seek(0, SeekOrigin.End); //Seeking for the last point in the file
            // fs.Write(info, 0, info.Length); // Writing data
        }

        void prodIniz()
        {
            string fileName = "product.txt"; //file Name
            FileStream fs; //object to write data
            string fileName1 = "date.txt"; //file Name
            FileStream fs1; //object to write data
            if (!File.Exists(pass + fileName)) //File exist?
```

```

        {
            fs = new FileStream(pass + fileName, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
        }
        else fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it

        if (!File.Exists(pass + fileName1)) //File exist?
        {
            fs1 = new FileStream(pass + fileName1, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
        }
        else fs1 = new FileStream(pass + fileName1, FileMode.Open,
FileAccess.ReadWrite); // Just open it

        Byte[] info = new Byte[fs.Length];
        fs.Read(info, 0, (int)fs.Length);
        string m = "";
        string m1 = "";
        for (int i = 0; i < info.Length; i++)
        {
            m += (char)info[i];
        }

        info = new Byte[fs1.Length];
        fs1.Read(info, 0, (int)fs1.Length);
        for (int i = 0; i < info.Length; i++)
        {
            m1 += (char)info[i];
        }

        string[] words = m.Split('@');
        if (words.Length > 0)
        {
            for (int i = 0; i < words.Length; i++)
            {
                string[] part = words[i].Split('^');
                if (part.Length != 3) continue;
                products.Rows.Add();
                int row = (products.Rows.Count - 1) < 0 ? 0 : products.Rows.Count -
1;

                product_name.Add(part[0]);
                stock_level.Add(Int32.Parse(part[1]));
                product_price.Add(Double.Parse(part[2]));
                comboBox1.Items.Add(part[0]);
                products.Rows[row].Cells[0].Value = part[0];
                products.Rows[row].Cells[1].Value = part[1];
                products.Rows[row].Cells[2].Value = part[2];
            }
        }

        fs1.Close();
        fs.Close();//Do not forget to close a file

        fileName = "sales.txt";
        string data="";
        if (File.Exists(pass + fileName1)) //File exist?
        {
            fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it
            info = new Byte[fs.Length];
            fs.Read(info, 0, (int)fs.Length);
            for (int i = 0; i < info.Length; i++)
            {

```

```

        data += (char)info[i];
    }
}
else data = "";

DateTime dt;
TimeSpan tm = new TimeSpan(180,0,0,0);
urgent="Urgent Sale Required:\n";
words = m1.Split('@');
if (words.Length > 0)
{
    for (int i = 0; i < words.Length; i++)
    {
        if (words[i] == "") break;
        products.Rows[i].Cells[3].Value = words[i];
        if (DateTime.TryParse(words[i], out dt))
        {
            if (dt + tm <
DateTime.Now&&data.IndexOf(products.Rows[i].Cells[0].Value.ToString())<0)
            {
                urgent += products.Rows[i].Cells[0].Value.ToString() + "\n";
                urg = true;
            }
        }
        else break;
    }
}

}

void orderIniz()
{
    string data = "";

    if (DateTime.Today.Day < 10) data += "0" + DateTime.Today.Day.ToString();
    else data += DateTime.Today.Day.ToString();
    if (DateTime.Today.Month < 10) data += "0" +
DateTime.Today.Month.ToString();
    else data += DateTime.Today.Month.ToString();
    data += DateTime.Today.Year.ToString();

    string fileName = "order.txt"; //file Name
    FileStream fs; //object to write data

    if (File.Exists(pass + fileName))
    {
        fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it
        Byte[] read = new Byte[8];
        fs.Seek(0, SeekOrigin.Begin);
        fs.Read(read, 0, 8);
        string data_n="";
        for(int i=0;i<read.Length;i++) data_n+= (char)read[i];

        if (String.Compare(data, data_n) == 0)
        {
            fs.Seek(0, SeekOrigin.Begin);
            Byte[] b = new Byte[fs.Length];

```

```

        fs.Read(b, 0, (int)fs.Length);
        data_n = "";
        for (int i = 0; i < b.Length; i++) data_n += (char)b[i];
        textBox9.Text = data_n;
        fs.Close();
        return;
    }
    else { fs.Close(); File.Delete(pass + fileName); }
}

fs = new FileStream(pass + fileName, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
data += "1";
fs.Seek(0, SeekOrigin.Begin);
Byte[] info1 = new UTF8Encoding(true).GetBytes(data); //Convert your data
to Bytes

fs.Write(info1, 0, info1.Length);
fs.Close();//Do not forget to close a file
textBox9.Text = data;
}

//void salesIniz()
//{
//    string fileName = "sales.txt"; //file Name
//    FileStream fs; //object to write data
//    if (!File.Exists(pass + fileName)) //File exist?
//    {
//        return;
//    }
//    else fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it

//    Byte[] inf = new Byte[fs.Length];
//    fs.Read(inf, 0, (int)fs.Length);

//    string s_inf = "";
//    for (int i = 0; i < inf.Length; i++) s_inf += (char)inf[i];

//    string[] mas = s_inf.Split('@');

//    for (int i = 0; i < mas.Length; i++)
//    {
//        string[] m = mas[i].Split('^');
//        if (m.Length != 3) continue;
//        string month = m[0].Substring(2, 2) + "." + m[0].Substring(4, 4);
//        // if (!comboBox2.Items.Contains(month)) comboBox2.Items.Add(month);
//    }

//    fs.Close();
//}

private void button1_Click(object sender, EventArgs e)
{
    DateTime dt;
    if (!(textBox1.Text.Length > 0 && textBox2.Text.Length > 0 &&
textBox3.Text.Length > 0)) return;
    if (!DateTime.TryParse(textBox8.Text, out dt)) return;

    int digit;
    double digit1;
    if (!(Int32.TryParse(textBox2.Text, out digit) &&
Double.TryParse(textBox3.Text, out digit1)))
    {
        MessageBox.Show("Wrong input data");
    }
}

```

```

        return;
    }

    FileStream fs2;

    if (product_name.Contains(textBox1.Text))
    {
        int index = product_name.IndexOf(textBox1.Text);
        product_price[index] = Double.Parse(textBox3.Text);
        stock_level[index] += Int32.Parse(textBox2.Text);
        products.Rows[index].Cells[1].Value = stock_level[index];
        products.Rows[index].Cells[2].Value = product_price[index];
        products.Rows[index].Cells[3].Value = dt.ToShortDateString();

        string data1 = "";
        string fileName1 = "product.txt"; //file Name
        string fileName2 = "date.txt";
        FileStream fs1; //object to write data
        if (File.Exists(pass + fileName1)) //File exist?
        {
            File.Delete(pass + fileName1);
        }
        fs1 = new FileStream(pass + fileName1, FileMode.Create,
        FileAccess.ReadWrite); //Create file for writing

        if (File.Exists(pass + fileName2)) //File exist?
        {
            File.Delete(pass + fileName2);
        }
        fs2 = new FileStream(pass + fileName2, FileMode.Create,
        FileAccess.ReadWrite); //Create file for writing

        for (int i = 0; i < products.Rows.Count; i++)
        {
            data1 += products.Rows[i].Cells[0].Value + "^" +
                products.Rows[i].Cells[1].Value + "^" +
                products.Rows[i].Cells[2].Value + "@";
        }

        Byte[] info1 = new UTF8Encoding(true).GetBytes(data1); //Convert your data
        to Bytes
        fs1.Seek(0, SeekOrigin.Begin);
        fs1.Write(info1, 0, info1.Length);

        info1 = new UTF8Encoding(true).GetBytes(dt.ToShortDateString() + "@");
        fs2.Seek(0, SeekOrigin.Begin);
        fs2.Write(info1, 0, info1.Length);

        fs2.Close();
        fs1.Close();//Do not forget to close a file

        return;
    }

    string data = textBox1.Text + "^"+textBox2.Text+"^"+textBox3.Text+"@";
    string data2 = dt.ToShortDateString() + "@";
    string fileName = "product.txt"; //file Name
    string fileName3 = "date.txt";
    FileStream fs; //object to write data
    if (!File.Exists(pass + fileName)) //File exist?
    {
        fs = new FileStream(pass + fileName, FileMode.Create,
        FileAccess.ReadWrite); //Create file for writing
    }

```

```

        else fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it

        if (!File.Exists(pass + fileName3)) //File exist?
        {
            fs2 = new FileStream(pass + fileName3, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
        }
        else fs2 = new FileStream(pass + fileName3, FileMode.Open,
FileAccess.ReadWrite); // Just open it

Bytes
        Byte[] info = new UTF8Encoding(true).GetBytes(data); //Convert your data to
Bytes
        fs.Seek(0, SeekOrigin.End);
        fs.Write(info, 0, info.Length);
        fs.Close();//Do not forget to close a file

        info = new UTF8Encoding(true).GetBytes(data2); //Convert your data to Bytes
        fs2.Seek(0, SeekOrigin.End);
        fs2.Write(info, 0, info.Length);
        fs2.Close();//Do not forget to close a file

        products.Rows.Add();
        int row = (products.Rows.Count-1) <0?0:products.Rows.Count-1;
        products.Rows[row].Cells[0].Value = textBox1.Text;
        products.Rows[row].Cells[1].Value = textBox2.Text;
        products.Rows[row].Cells[2].Value = textBox3.Text;
        products.Rows[row].Cells[3].Value = dt.ToShortDateString();
        product_name.Add(textBox1.Text);
        comboBox1.Items.Add(textBox1.Text);
        stock_level.Add(Int32.Parse(textBox2.Text));
        product_price.Add(Double.Parse(textBox3.Text));
    }

private void button2_Click(object sender, EventArgs e)
{
    if (comboBox1.SelectedIndex >= 0)
    {
        string data = "";
        string data1 = "";
        string delete = comboBox1.Items[comboBox1.SelectedIndex].ToString();
        //MessageBox.Show(comboBox1.Items[comboBox1.SelectedIndex].ToString());
        product_name.Remove(delete);
        stock_level.RemoveAt(comboBox1.SelectedIndex);
        product_price.Remove(comboBox1.SelectedIndex);
        products.Rows.RemoveAt(comboBox1.SelectedIndex);
        comboBox1.Items.Remove(comboBox1.Items[comboBox1.SelectedIndex]);

        for (int i = 0; i < products.Rows.Count; i++)
        {
            data +=products.Rows[i].Cells[0].Value.ToString()+"^" +
                products.Rows[i].Cells[1].Value.ToString()+"^" +
                products.Rows[i].Cells[2].Value.ToString()+"@";
            data1 += products.Rows[i].Cells[3].Value.ToString() + "@";
        }

        string fileName = "product.txt"; //file Name
        string fileName1 = "date.txt"; //file Name
        FileStream fs; //object to write data
        if (File.Exists(pass + fileName)) //File exist?
        {

```

```

        File.Delete(pass + fileName);
    }
    fs = new FileStream(pass + fileName, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
    Byte[] info = new UTF8Encoding(true).GetBytes(data); //Convert your data
to Bytes
    fs.Seek(0, SeekOrigin.Begin);
    fs.Write(info, 0, info.Length);
    fs.Close();//Do not forget to close a file

    if (File.Exists(pass + fileName1)) //File exist?
    {
        File.Delete(pass + fileName1);
    }
    fs = new FileStream(pass + fileName1, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
    info = new UTF8Encoding(true).GetBytes(data1); //Convert your data to
Bytes
    fs.Seek(0, SeekOrigin.Begin);
    fs.Write(info, 0, info.Length);
    fs.Close();//Do not forget to close a file
    }
}

private void tabControl1_SelectedIndexChanged(object sender, EventArgs e)
{
    switch(tabControl1.SelectedIndex)
    {
        case 0: comboBox1.Location = new Point(460, 260); comboBox1.Parent =
tabControl1.TabPages[0]; break;
        case 1: comboBox1.Location = new Point(20, 110); comboBox1.Parent =
tabControl1.TabPages[1];
            textBox6.Text=textBox5.Text = ""; textBox7.Text= DateTime.Today.Day
+"."+DateTime.Today.Month+"."+DateTime.Today.Year;

            break;
            //case 2: for (int i = 0; i < comboBox2.Items.Count;i++)
comboBox2.Items.RemoveAt(0); salesIniz(); break;
    }
}

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    if (comboBox1.Parent == tabControl1.TabPages[1])
    {
        textBox6.Text = stock_level[comboBox1.SelectedIndex].ToString();
        textBox5.Text = product_price[comboBox1.SelectedIndex].ToString();
        textBox4.Text = textBox10.Text = "";
    }
}

private void textBox4_TextChanged(object sender, EventArgs e)
{
    if (comboBox1.SelectedIndex < 0) return;
    int digit;
    if(Int32.TryParse(textBox4.Text,out
digit)&&digit<=stock_level[comboBox1.SelectedIndex]&&digit>0)
    {
        textBox10.Text = (product_price[comboBox1.SelectedIndex] *
digit).ToString();
        textBox4.BackColor = Color.White;
    }
    else
    {

```

```

        textBox4.BackColor = Color.Red;
        textBox10.Text = "";
    }
}

private void button3_Click(object sender, EventArgs e)
{
    if (comboBox1.SelectedIndex >= 0 && textBox4.BackColor !=
Color.Red&&textBox4.Text.Length>0)
    {
        richTextBox1.Text += "\"" + comboBox1.Text + "\"" + " " + textBox4.Text
+ "ct" + " " + textBox10.Text+"\n";
        cell_history.Add(comboBox1.SelectedIndex + "@" + textBox4.Text + "@" +
textBox10.Text);
        stock_level[comboBox1.SelectedIndex] -= Int32.Parse(textBox4.Text);
        textBox6.Text = stock_level[comboBox1.SelectedIndex].ToString();
        products.Rows[comboBox1.SelectedIndex].Cells[1].Value =
stock_level[comboBox1.SelectedIndex];
        textBox11.Text = (Double.Parse(textBox11.Text) +
Double.Parse(textBox10.Text)).ToString();

        string data1 = "";
        string fileName1 = "product.txt"; //file Name
        FileStream fs1; //object to write data
        if (File.Exists(pass + fileName1)) //File exist?
        {
            File.Delete(pass + fileName1);
        }
        fs1 = new FileStream(pass + fileName1, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing

        for (int i = 0; i < products.Rows.Count; i++)
        {
            data1 += products.Rows[i].Cells[0].Value + "^" +
                products.Rows[i].Cells[1].Value + "^" +
                products.Rows[i].Cells[2].Value + "@";
        }

        Byte[] info1 = new UTF8Encoding(true).GetBytes(data1); //Convert your
data to Bytes
        fs1.Seek(0, SeekOrigin.Begin);
        fs1.Write(info1, 0, info1.Length);
        fs1.Close();//Do not forget to close a file
        textBox4.Text = "";
    }
}

private void button4_Click(object sender, EventArgs e)
{
    if (richTextBox1.Text.Length > 0)
    {
        string[] m = cell_history[cell_history.Count - 1].Split('@');
        if (m.Length != 3) return;
        stock_level[Int32.Parse(m[0])] += Int32.Parse(m[1]);
        products.Rows[Int32.Parse(m[0])].Cells[1].Value =
stock_level[Int32.Parse(m[0])];

        textBox11.Text = (Double.Parse(textBox11.Text) -
Double.Parse(m[2])).ToString();
        string data1 = "";
        string fileName1 = "product.txt"; //file Name
        FileStream fs1; //object to write data
        if (File.Exists(pass + fileName1)) //File exist?
        {
            File.Delete(pass + fileName1);

```



```

    }
    fs1 = new FileStream(pass + fileName1, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing

    for (int i = 0; i < products.Rows.Count; i++)
    {
        data1 += products.Rows[i].Cells[0].Value + "^" +
            products.Rows[i].Cells[1].Value + "^" +
            products.Rows[i].Cells[2].Value + "@";
    }

    Byte[] info1 = new UTF8Encoding(true).GetBytes(data1); //Convert your
data to Bytes
    fs1.Seek(0, SeekOrigin.Begin);
    fs1.Write(info1, 0, info1.Length);
    fs1.Close();//Do not forget to close a file

    string[] words = richTextBox1.Text.Split('\n');

    richTextBox1.Text = "";

    for (int i = 0; i < words.Length - 2; i++)
    {
        richTextBox1.Text += words[i]+"\\n";
    }

    comboBox1.SelectedIndex = Int32.Parse(m[0]);
    textBox6.Text = stock_level[Int32.Parse(m[0])].ToString();
    cell_history.RemoveAt(cell_history.Count - 1);
}
}

private void button5_Click(object sender, EventArgs e)
{
    if (richTextBox1.Text.Length == 0) return;
    string fileName = "sales.txt"; //file Name
    FileStream fs; //object to write data
    if (!File.Exists(pass + fileName)) //File exist?
    {
        fs = new FileStream(pass + fileName, FileMode.Create,
FileAccess.ReadWrite); //Create file for writing
    }
    else fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it

    Byte[] info1 = new
UTF8Encoding(true).GetBytes(textBox9.Text+"^"+textBox11.Text+"^"+richTextBox1.Text+"@" );
//Convert your data to Bytes
    fs.Seek(0, SeekOrigin.End);
    fs.Write(info1, 0, info1.Length);
    fs.Close();//Do not forget to close a file

    richTextBox1.Text = "";
    string s = textBox9.Text.Substring(8);
    textBox9.Text = textBox9.Text.Substring(0,8)+(Int64.Parse(s) + 1).ToString();

    ////////////////
    fileName = "order.txt"; //file Name
    if (File.Exists(pass + fileName))
    {
        File.Delete(pass + fileName);
    }
    fs = new FileStream(pass + fileName, FileMode.Create, FileAccess.ReadWrite);
//Create file for writing

```

```

        Byte[] info = new UTF8Encoding(true).GetBytes(textBox9.Text); //Convert your
data to Bytes
        fs.Write(info, 0, info.Length);
        fs.Close();//Do not forget to close a file
        textBox11.Text = "0";
        //////////////////////////////////////
        MessageBox.Show("Order was successfully created");
    }

//private void button6_Click(object sender, EventArgs e)
//{
//    if (comboBox2.SelectedIndex >= 0)
//    {

//        List<string> pr_name = new List<string>();
//        List<int> pr_count = new List<int>();

//        richTextBox2.Text = "";
//        string word = comboBox2.Items[comboBox2.SelectedIndex].ToString();
//        string[] m1 = word.Split('.');
//        word = m1[0] + m1[1];
//        double sum = 0;

//        string fileName = "sales.txt"; //file Name
//        FileStream fs; //object to write data
//        if (!File.Exists(pass + fileName)) //File exist?
//        {
//            return;
//        }
//        else fs = new FileStream(pass + fileName, FileMode.Open,
FileAccess.ReadWrite); // Just open it

//        Byte[] inf = new Byte[fs.Length];
//        fs.Read(inf, 0, (int)fs.Length);          string s_inf = "";
//        for (int i = 0; i < inf.Length; i++) s_inf += (char)inf[i];
//        string name,count="";
//        string[] mas = s_inf.Split('@');

//        for (int i = 0; i < mas.Length; i++)
//        {
//            string[] m = mas[i].Split('^');
//            if (m.Length != 3) continue;
//            if (m[0].Substring(2, 6) != word) continue;
//            richTextBox2.Text += m[2];
//            sum += Double.Parse(m[1]);

//            string[] v = m[2].Split('\n');

//            for (int k = 0; k < v.Length; k++)
//            {
//                if (v[k] == "") continue;
//                name = "";
//                for (int j = 1; j < v[k].Length; j++)
//                {
//                    if (v[k][j] == (char)34)
//                    {
//                        count = "";
//                        j += 4;
//                        while (m[2][j] != 'c')
//                        {
//                            count += v[k][j];
//                            j++;
//                        }
//                    }
//                }
//            }
//        }
//    }
//}

```

```

//          break;
//      }
//      name += v[k][j];
//  }

//      if (!pr_name.Contains(name))
//      {
//          pr_name.Add(name);
//          pr_count.Add(Int32.Parse(count));
//      }
//      else
//      {
//          int index = pr_name.IndexOf(name);
//          pr_count[index] += Int32.Parse(count);
//      }

//  }
//  }

//      for (int i = 0; i < 23; i++) richTextBox2.Text += "_";
//      int num=pr_count.Max();
//      int g = pr_count.IndexOf(num);
//      richTextBox2.Text += "\nFast selling product and it`s stock level:\n";
//      bool exist=false;
//      for (int i = 0; i < products.Rows.Count; i++)
//          if (products.Rows[i].Cells[0].Value.ToString() == pr_name[g]) {
exist = true;num=i; break; }
//          string part = exist ?
(products.Rows[num].Cells[1].Value.ToString().ToString() + "ct") : "out of stock";
//          richTextBox2.Text += pr_name[g] + "    " + part;
//          richTextBox2.Text += "\nSlow selling product and it`s stock level:\n";

//          num=pr_count.Min();
//          g = pr_count.IndexOf(num);
//          exist = false;
//          for (int i = 0; i < products.Rows.Count; i++)
//              if (products.Rows[i].Cells[0].Value.ToString() == pr_name[g]) {
exist = true; num = i; break; }
//              part = exist ? (products.Rows[num].Cells[1].Value.ToString() + "ct") :
"out of stock";
//              richTextBox2.Text += pr_name[g] + "    " + part;

//          textBox12.Text = sum.ToString();
//          fs.Close();
//      }
//  }

private void Form1_FormClosing(object sender, FormClosingEventArgs e)
{
    int i = 0;
    while (richTextBox1.Text.Length > 0&&i<100)
    {
        button4_Click(button4, null);
        i++;
    }
}

private void Form1_Activated(object sender, EventArgs e)
{
    if (urg)
    {
        urg = false;
        MessageBox.Show(urgent);
    }
}

```

```

    }
}
}

public class Data_Grid : DataGridView
{

    public Data_Grid(int size = 100, params string[] m)
    {
        int i = 0;
        foreach (var str in m)
        {
            this.Columns.Add(new DataGridViewTextBoxColumn { HeaderText = str });
            this.Columns[i].Width = size;
            i++;
        }

        foreach (DataGridViewColumn column in this.Columns)
        {
            column.SortMode = DataGridViewColumnSortMode.NotSortable;
        }

        this.AllowUserToResizeColumns = false;
        this.AllowUserToResizeRows = false;
        this.ColumnHeadersHeightSizeMode =
DataGridViewColumnHeadersHeightSizeMode.DisableResizing;
        this.RowHeadersWidthSizeMode =
DataGridViewRowHeadersWidthSizeMode.DisableResizing;
        this.AllowUserToAddRows = false;
        this.RowPrePaint += dataGridView_RowPrePaint;
        this.RowHeadersWidth = 50;
    }

    public Data_Grid(Data_Grid obj, int size)
    {
        for (int i = 0; i < obj.Columns.Count; i++)
        {
            this.Columns.Add(new DataGridViewTextBoxColumn { HeaderText =
obj.Columns[i].HeaderText });
            this.Columns[i].Width = size;
        }
        for (int i = 0; i < obj.Rows.Count; i++)
        {
            this.Rows.Add();
            for (int j = 0; j < obj.Rows[i].Cells.Count; j++)
            {
                this.Rows[i].Cells[j].Value = obj.Rows[i].Cells[j].Value;
            }
        }

        foreach (DataGridViewColumn column in this.Columns)
        {
            column.SortMode = DataGridViewColumnSortMode.NotSortable;
        }

        this.AllowUserToResizeColumns = false;
        this.AllowUserToResizeRows = false;
        this.ColumnHeadersHeightSizeMode =
DataGridViewColumnHeadersHeightSizeMode.DisableResizing;
        this.RowHeadersWidthSizeMode =
DataGridViewRowHeadersWidthSizeMode.DisableResizing;
    }
}

```

```
        this.AllowUserToAddRows = false;
        this.RowPrePaint += dataGridView_RowPrePaint;
        this.RowHeadersWidth = 50;
    }

    public static void dataGridView_RowPrePaint(object sender,
DataGridViewRowPrePaintEventArgs e)
    {
        DataGridView dt = (DataGridView)sender;
        object head = dt.Rows[e.RowIndex].HeaderCell.Value;
        if (head == null || !head.Equals((e.RowIndex + 1).ToString()))
        {
            dt.Rows[e.RowIndex].HeaderCell.Value =
                (e.RowIndex + 1).ToString();
        }
    }
}

}
```