

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Саратовский государственный технический университет имени Гагарина Ю. А.»

Кафедра
«Прикладные информационные технологии»

ОТЧЕТ
по лабораторной работе №3

Студента группы 62-ПИНФ21
Нефёдова Д. В.

```
using System;
using System.Drawing;
using System.Drawing.Printing;
using System.Text.RegularExpressions;
using System.Windows.Forms;

namespace Lab_3
{
    public partial class PamphletForm : Form
    {
        private String fileName;
        protected Image image;
        public String documentContents;

        public PamphletForm()
        {
            InitializeComponent();
        }

        private bool PhoneNumberIsCorrect()
        {
            Regex phoneRegex = new Regex(@"\+\d{11,11}");
            return phoneRegex.IsMatch(phoneNumberTextBox.Text);
        }

        private bool AddressIsCorrect()
        {
            return addressTextBox.Text.Length > 0;
        }

        private bool NameIsCorrect()
        {
            return nameTextBox.Text.Length > 0;
        }

        private bool ContactPersonIsCorrect()
        {
            return contactPersonTextBox.Text.Length > 0;
        }

        private bool TitleIsCorrect()
        {
            return titleTextBox.Text.Length > 0;
        }

        private bool DateIsCorrect()
        {
            return DateTime.TryParse(dateTextBox.Text, out _);
        }
    }
}
```

```
private void NameTextBox_Leave(object sender, EventArgs e)
{
    if (NameIsCorrect())
        errorProvider.SetError(nameTextBox, String.Empty);
    else
        errorProvider.SetError(nameTextBox, "Name is incorrect.");

    EnablePanel();
}

private void AddressTextBox_Leave(object sender, EventArgs e)
{
    if (AddressIsCorrect())
        errorProvider.SetError(addressTextBox, String.Empty);
    else
        errorProvider.SetError(addressTextBox, "Address is incorrect.");

    EnablePanel();
}

private void TitleTextBox_Leave(object sender, EventArgs e)
{
    if (TitleIsCorrect())
        errorProvider.SetError(titleTextBox, String.Empty);
    else
        errorProvider.SetError(titleTextBox, "Title is incorrect.");

    EnablePanel();
}

private void ContactPersonTextBox_Leave(object sender, EventArgs e)
{
    if (ContactPersonIsCorrect())
        errorProvider.SetError(contactPersonTextBox, String.Empty);
    else
        errorProvider.SetError(contactPersonTextBox, "Contact person info is incorrect.");

    EnablePanel();
}

private void PhoneNumberTextBox_Leave(object sender, EventArgs e)
{
    if (PhoneNumberIsCorrect())
        errorProvider.SetError(phoneNumberTextBox, String.Empty);
    else
        errorProvider.SetError(phoneNumberTextBox, "Phone number is incorrect.");
}
```

```
        EnablePanel();
    }

    private void DateTextBox_Leave(object sender, EventArgs e)
    {
        dateTextBox.Text = DateTime.Now.ToShortDateString();
        EnablePanel();
    }

    private void ColorButton_Click(object sender, EventArgs e)
    {
        if (colorDialog.ShowDialog() == DialogResult.OK)
            this.richTextBox.ForeColor = colorDialog.Color;
    }

    private void OpenButton_Click(object sender, EventArgs e)
    {
        openFileDialog.Filter = "Image Files(*.BMP;*.JPG;*.GIF)|*.BMP;*.JPG;*.GIF|All files(*.*)|
*.*";

        if (openFileDialog.ShowDialog() == DialogResult.OK)
        {
            fileName = openFileDialog.FileName;
            image = Image.FromFile(fileName);
        }
    }

    private void FontButton_Click(object sender, EventArgs e)
    {
        if (fontDialog.ShowDialog() == DialogResult.OK)
            this.richTextBox.Font = fontDialog.Font;
    }

    private void ExitButton_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    protected void PrintDocument1_PrintPage(object sender, PrintPageEventArgs e)
    {
        e.Graphics.DrawImage(image, new Point(0, richTextBox.Font.Height));
    }

    private void PrintButton_Click(object sender, EventArgs e)
    {
        PrintPreviewDialog printDialog = new PrintPreviewDialog();
        PrintDocument printDocument = new PrintDocument();
        printDialog.Document = printDocument;
    }
}
```

```
printDocument.PrintPage += PrintDocument_PrintPage;
DialogResult result = printDialog.ShowDialog();

if (result == DialogResult.OK)
{
    printDocument.Print();
    printDialog.Close();
}
}

private void PrintDocument_PrintPage(object sender, PrintPageEventArgs e)
{
    e.Graphics.DrawString(richTextBox.Text, richTextBox.Font, new
SolidBrush(richTextBox.ForeColor), new Point(0, 0));

    if (openFileDialog.FileName != "openFileDialog")
        e.Graphics.DrawImage(image, new Point(0, richTextBox.Font.Height + 30));
    else
        e.Graphics.DrawImage(Image.FromFile("screevvfa74633nshot.png"), new Point(0,
richTextBox.Font.Height + 30));
}

private void EnablePanel()
{
    if (NameIsCorrect()
        && AddressIsCorrect()
        && TitleIsCorrect()
        && ContactPersonIsCorrect()
        && PhoneNumberIsCorrect()
        && DateIsCorrect())
    {
        panel.Enabled = true;
    }
}

void PrintPageHandler(object sender, PrintPageEventArgs e)
{
    e.Graphics.DrawString(richTextBox.Text, richTextBox.Font, new
SolidBrush(richTextBox.ForeColor), new Point(0, 0));
}
}
}
```

```
using System;
using System.Windows.Forms;

namespace Lab_3
{
    public partial class MainForm : Form
    {
        public int childrenCount = 0;

        public MainForm()
        {
            InitializeComponent();
        }

        private void MainForm_Load(object sender, EventArgs e)
        {
            toolStripStatusLabel1.Text = DateTime.Now.ToLongDateString();
        }

        private void WindowMember_Click(object sender, EventArgs e)
        {
            ToolStripItemCollection items = windowToolStripMenuItem.DropDownItems;

            foreach (ToolStripItem i in items)
                if (i.Equals(sender))
                    foreach (InheritedPamphletForm f in MdiChildren)
                        if (sender.Equals(f.stripItem))
                            f.Activate();
        }

        private void CreateProduct_Click(object sender, EventArgs e)
        {
            CreateChildForm("Name of the product", " Name of the product");
        }

        private void CreateHotel_Click(object sender, EventArgs e)
        {
            CreateChildForm("Name of the hotel", " Name of the hotel");
        }

        private void OrganizationCreate_Click(object sender, EventArgs e)
        {
            CreateChildForm("Name of the organisation", " Name of the organisation");
        }

        private void CreateChildForm(string formName, string windowTitle)
        {
            ToolStripMenuItem newItem = new ToolStripMenuItem(Convert.ToString(++childrenCount) +
```

```
windowTitle);
```

```
        Form f = new InheritedPamphletForm(formName, windowToolStripMenuItem, newItem, this)
        {
            MdiParent = this
        };

        f.Show();

        windowToolStripMenuItem.DropDownItems.Add(newItem);
        newItem.Click += WindowMember_Click;
    }

    private void Cascad_Click(object sender, EventArgs e)
    {
        ShowAllWindows();
        this.LayoutMdi(MdiLayout.Cascade);
    }

    private void Mosaic_Click(object sender, EventArgs e)
    {
        ShowAllWindows();
        LayoutMdi(MdiLayout.TileVertical);
    }

    private void ShowAllWindows()
    {
        foreach (var f in MdiChildren)
        {
            if (f.WindowState == FormWindowState.Minimized)
            {
                f.WindowState = FormWindowState.Normal;
                f.Show();
            }
        }
    }

    private void AlignLeft_Click(object sender, EventArgs e)
    {
        AlignRichTextBox(HorizontalAlignment.Left);
    }

    private void AlignCenter_Click(object sender, EventArgs e)
    {
        AlignRichTextBox(HorizontalAlignment.Center);
    }

    private void AlignRight_Click(object sender, EventArgs e)
```

```
{
    AlignRichTextBox(HorizontalAlignment.Right);
}

private void AlignRichTextBox(HorizontalAlignment alignment)
{
    Form activeChild = this.ActiveMdiChild;

    if (activeChild != null)
    {
        try
        {
            RichTextBox box = (RichTextBox)activeChild.ActiveControl;

            if (box != null)
                box.SelectionAlignment = alignment;
        }
        catch
        {
            MessageBox.Show("You need to select a RichTextBox.");
        }
    }
}

private void Timer_Tick(object sender, EventArgs e)
{
    toolStripStatusLabel2.Text = DateTime.Now.ToLongTimeString();
}
}
```

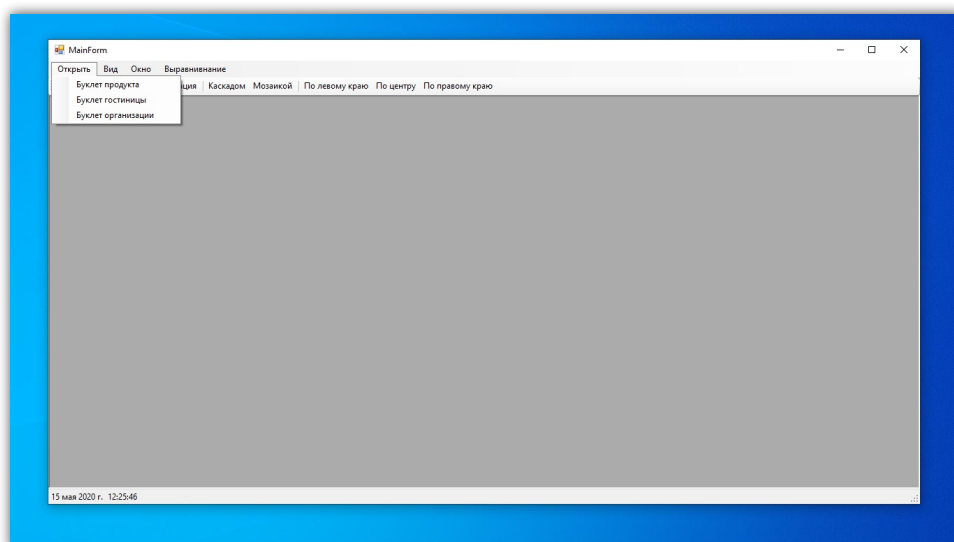



Рис. 1. Пункт меню «Открыть».

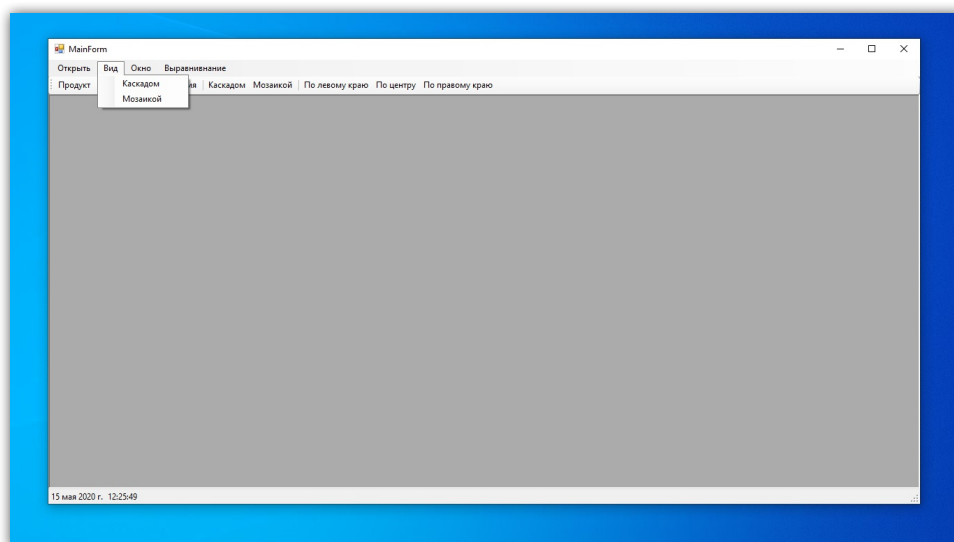


Рис. 2. Пункт меню «Вид».

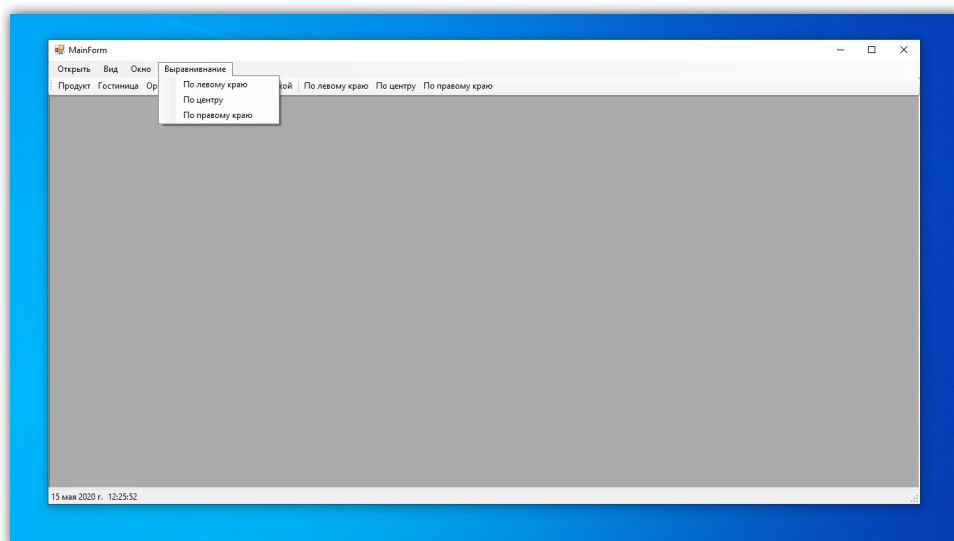


Рис. 3. Пункт меню «Выравнивание».

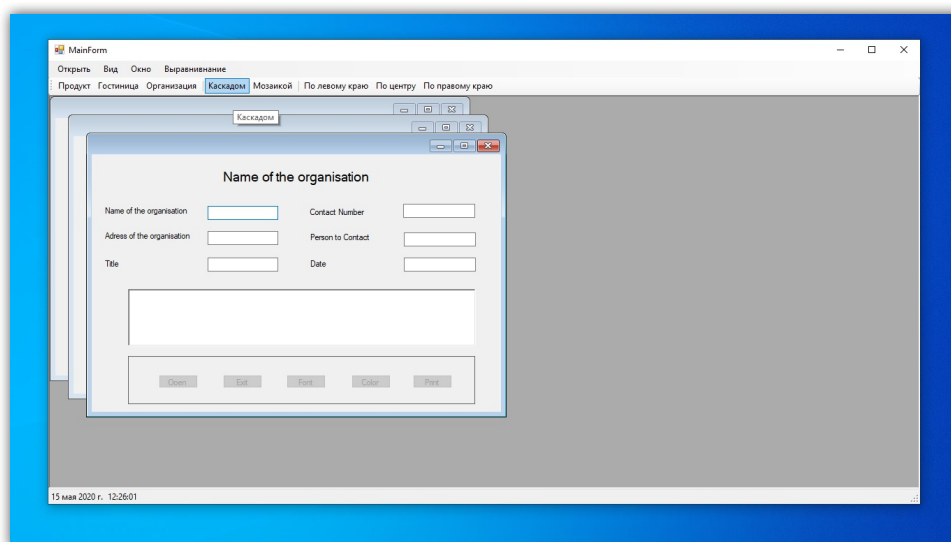


Рис. 4. Расположение окон каскадом.

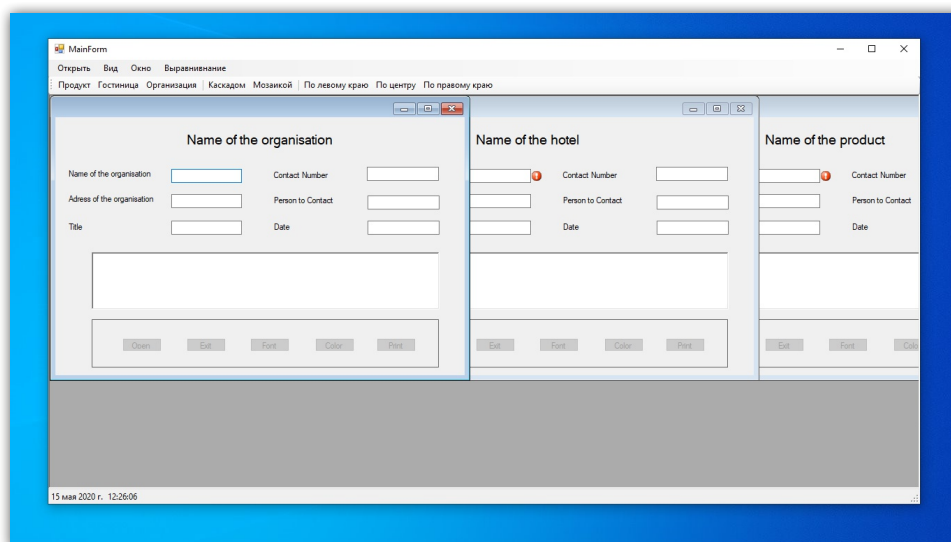


Рис. 5. Расположение окон мозаикой.

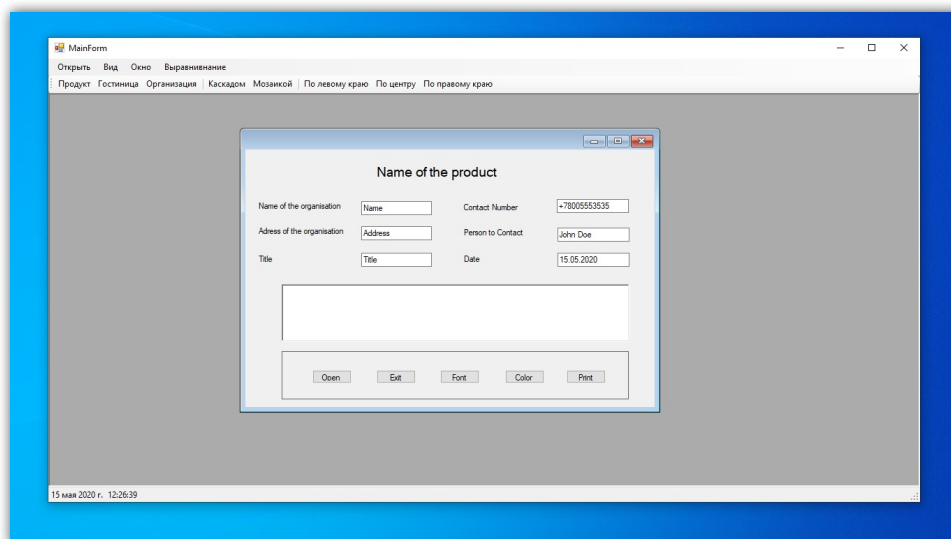


Рис. 6. Заполненная форма продукта.

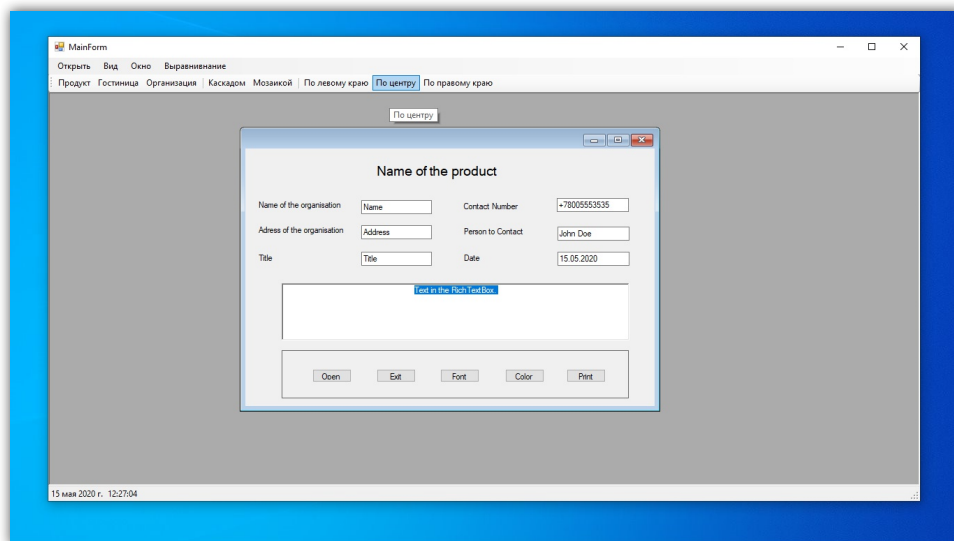


Рис. 7. Выравнивание теста по центру.

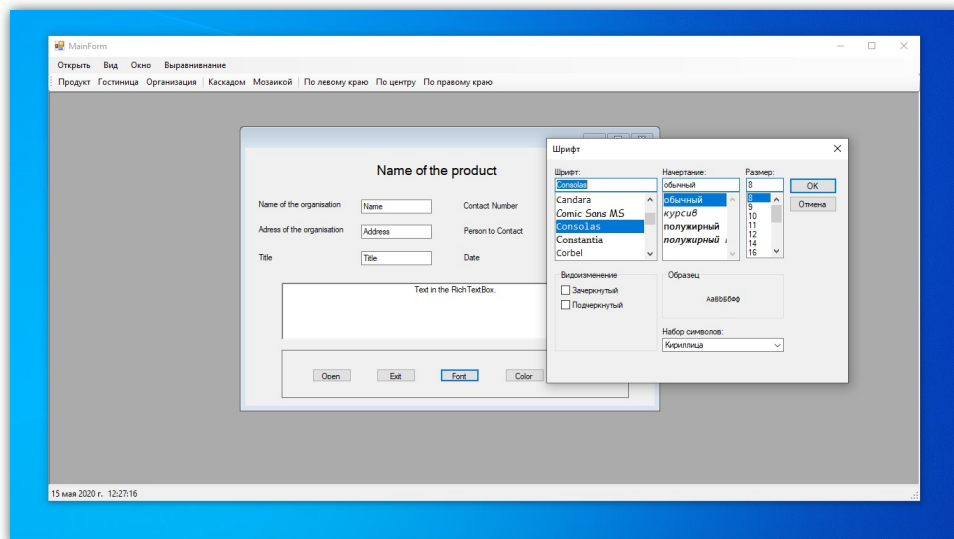


Рис. 8. Диалог выбора шрифта.

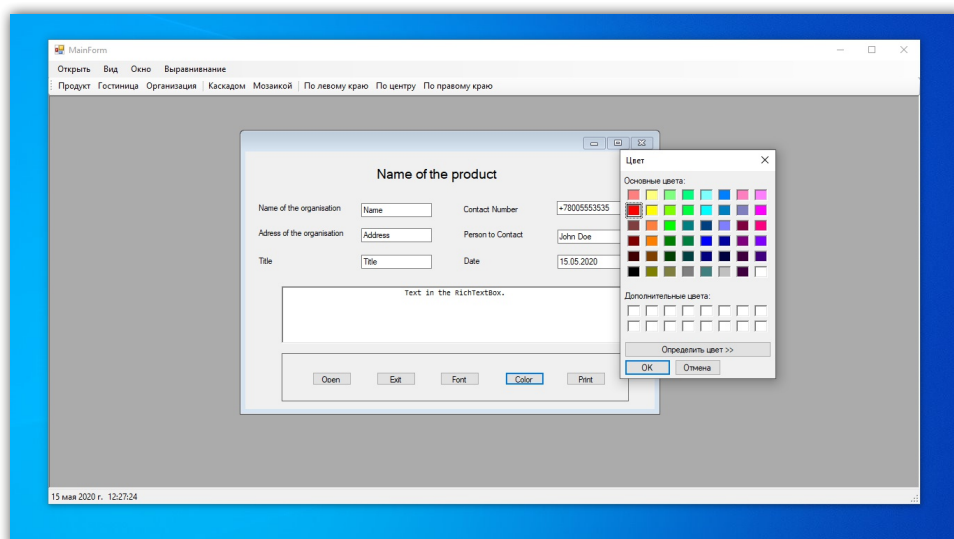


Рис. 9. Диалог выбора цвета.