**Лабораторная работа № 5**

**Тема: Интеграция C# с Microsoft.Office.**

**Цель работы:** Изучить способы создания и манипулирование файлами Microsoft Word и Microsoft Excel используя средства инструментальной среды и язык программирования С#.

**Программное обеспечение:** Windows 7 и выше, Microsoft Office Word 2010 и выше, Visual Studio 2015 и выше.

**Теоретические сведения к выполнению лабораторной работы:**

Для того чтобы провести интеграцию C# c Microsoft.Office в инструментальной среде программирования Visual Studio необходимо добавить к текущему проекту объектную библиотеку MS Word и MS Excel. Для этого перейдите в меню «**Проект**» и выберете команду «**Добавить ссылку**» или в обозревателе решений, найдите пункт «**Ссылки**» и сделайте клик правой клавишей мыши по нему, после выберите «**Добавить ссылку…**».

В открывшемся окне «Добавить ссылку», перейдите на вкладку «COM» и выберете «**Microsoft Word 15.0 Object Library**» и «**Microsoft Excel 15.0 Object Library**» из предложенного списка библиотек (В зависимости от версии Visual Studio версия библиотек может отличаться).

Чтобы облегчим написание программного кода воспользуйтесь добавлением собственной директорией **using**.

using Word = Microsoft.Office.Interop.Word;

using Excel = Microsoft.Office.Interop.Excel;

**Этапы выполнения:**

1. Реализовать внешний вид используя следующие компоненты:

Для реализации данной программы на форму необходимо добавить два Button с названиями:

* Создание нового документа Word;
* Создание нового документа Excel.

1. Код создания элементов в Word и Excel;

Word.Application winword = new Word.Application(); //создаем COM-объект Word

object missing = System.Reflection.Missing.Value;

winword.Visible = false;

//Заголовок документа

winword.Documents.Application.Caption = "Тест";

object missing = System.Reflection.Missing.Value;

//Создание нового документа

Word.Document document = winword.Documents.Add(ref missing, ref missing, ref missing, ref missing);

//добавление новой страницы

//winword.Selection.InsertNewPage();

//Добавление верхнего колонтитула

foreach (Word.Section section in document.Sections)

{

Word.Range headerRange = section.Headers[Word.WdHeaderFooterIndex.wdHeaderFooterPrimary].Range;

headerRange.Fields.Add(headerRange, Word.WdFieldType.wdFieldPage);

headerRange.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

headerRange.Font.ColorIndex = Word.WdColorIndex.wdBlue;

headerRange.Font.Size = 10;

headerRange.Text = "Верхний колонтитул" + Environment.NewLine + "Тест";

}

//Добавление нижнего колонтитула

foreach (Word.Section wordSection in document.Sections)

{

Word.Range footerRange = wordSection.Footers[Word.WdHeaderFooterIndex.wdHeaderFooterPrimary].Range;

//Установка цвета текста

footerRange.Font.ColorIndex = Word.WdColorIndex.wdDarkRed;

//Размер

footerRange.Font.Size = 10;

//Установка расположения по центру

footerRange.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

//Установка текста для вывода в нижнем колонтитуле

footerRange.Text = "Нижний колонтитул" + Environment.NewLine + "Тест";

}

//Добавление текста в документ

document.Content.SetRange(0, 0);

document.Content.Text = "Тест" + Environment.NewLine;

//Добавление текста со стилем Заголовок 1

Word.Paragraph para1 = document.Content.Paragraphs.Add(ref missing);

object styleHeading1 = "Заголовок 1";

para1.Range.set\_Style(styleHeading1);

para1.Range.Text = "Очень много текста";

para1.Range.InsertParagraphAfter();

//Создание таблицы 5х5

Word.Table firstTable = document.Tables.Add(para1.Range, 5, 5, ref missing, ref missing);

firstTable.Borders.Enable = 1;

foreach (Word.Row row in firstTable.Rows)

{

foreach (Word.Cell cell in row.Cells)

{

//Заголовок таблицы

if (cell.RowIndex == 1)

{

cell.Range.Text = "Колонка " + cell.ColumnIndex.ToString();

cell.Range.Font.Bold = 1;

//Задаем шрифт и размер текста

cell.Range.Font.Name = "verdana";

cell.Range.Font.Size = 10;

cell.Shading.BackgroundPatternColor = Word.WdColor.wdColorGray25;

//Выравнивание текста в заголовках столбцов по центру

cell.VerticalAlignment =

Word.WdCellVerticalAlignment.wdCellAlignVerticalCenter;

cell.Range.ParagraphFormat.Alignment =

Word.WdParagraphAlignment.wdAlignParagraphCenter;

}

//Значения ячеек

else

{

cell.Range.Text = (cell.RowIndex - 2 + cell.ColumnIndex).ToString();

}

}

}

winword.Visible = true;

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void button2\_Click(object sender, EventArgs e)

{

//Объявляем приложение

Excel.Application ex = new Excel.Application();

//Отобразить Excel

ex.Visible = true;

//Количество листов в рабочей книге

ex.SheetsInNewWorkbook = 2;

//Добавить рабочую книгу

Excel.Workbook workBook = ex.Workbooks.Add(Type.Missing);

//Отключить отображение окон с сообщениями

ex.DisplayAlerts = false;

//Получаем первый лист документа (счет начинается с 1)

Excel.Worksheet sheet = (Excel.Worksheet)ex.Worksheets.get\_Item(1);

//Название листа (вкладки снизу)

sheet.Name = "Отчет за 13.12.2017";

//Пример заполнения ячеек

for (int i = 1; i <= 9; i++)

{

for (int j = 1; j < 9; j++)

sheet.Cells[i, j] = String.Format("Boom {0} {1}", i, j);

}

//Захватываем диапазон ячеек

Excel.Range range1 = sheet.get\_Range(sheet.Cells[1, 1], sheet.Cells[9, 9]);

//Шрифт для диапазона

range1.Cells.Font.Name = "Tahoma";

//Размер шрифта для диапазона

range1.Cells.Font.Size = 10;

//Захватываем другой диапазон ячеек

Excel.Range range2 = sheet.get\_Range(sheet.Cells[1, 1], sheet.Cells[9, 2]);

range2.Cells.Font.Name = "Times New Roman";

//Задаем цвет этого диапазона. Необходимо подключить System.Drawing

range2.Cells.Font.Color = ColorTranslator.ToOle(Color.Green);

//Фоновый цвет

range2.Interior.Color = ColorTranslator.ToOle(Color.FromArgb(0xFF, 0xFF, 0xCC));

//Растановка рамок

range2.Borders.get\_Item(Excel.XlBordersIndex.xlEdgeBottom).LineStyle = Excel.XlLineStyle.xlContinuous;

range2.Borders.get\_Item(Excel.XlBordersIndex.xlEdgeRight).LineStyle = Excel.XlLineStyle.xlContinuous;

range2.Borders.get\_Item(Excel.XlBordersIndex.xlInsideHorizontal).LineStyle = Excel.XlLineStyle.xlContinuous;

range2.Borders.get\_Item(Excel.XlBordersIndex.xlInsideVertical).LineStyle = Excel.XlLineStyle.xlContinuous;

range2.Borders.get\_Item(Excel.XlBordersIndex.xlEdgeTop).LineStyle = Excel.XlLineStyle.xlContinuous;

// Цвет рамки

range2.Borders.Color = ColorTranslator.ToOle(Color.Red);

//Формулы

//Для начала снова получим диапазон ячеек

Excel.Range formulaRange = sheet.get\_Range(sheet.Cells[4, 1], sheet.Cells[9, 1]);

//Далее получим диапазон вида A4:A10 по адресу ячейки ( [4,1]; [9;1] ) описанному выше

string adder = formulaRange.get\_Address(1, 1, Excel.XlReferenceStyle.xlA1, Type.Missing, Type.Missing);

//Теперь в переменной adder у нас хранится строковое значение диапазона ( [4,1]; [9;1] ), то есть A4:A10

//Одна ячейка как диапазон

Excel.Range r = sheet.Cells[10, 1] as Excel.Range;

//Оформления

r.Font.Name = "Times New Roman";

r.Font.Bold = true;

r.Font.Color = ColorTranslator.ToOle(Color.Blue);

//Задаем формулу суммы

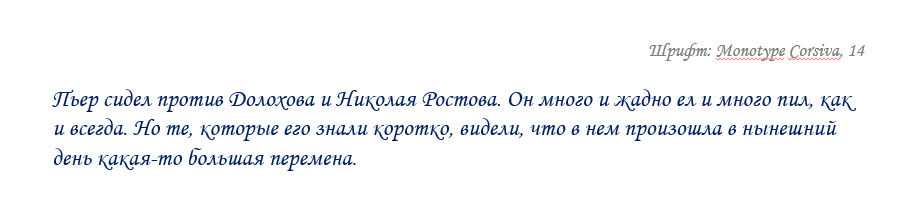
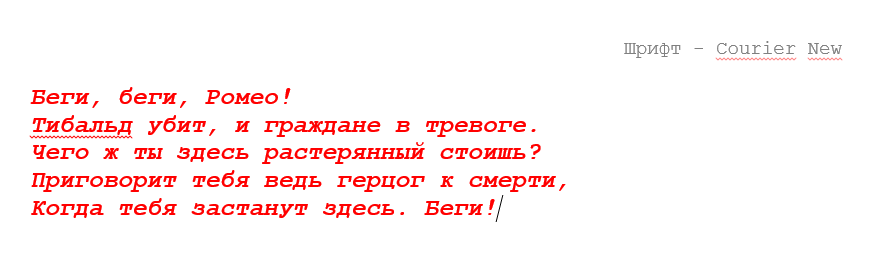
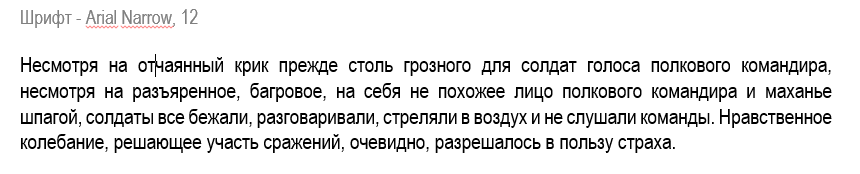
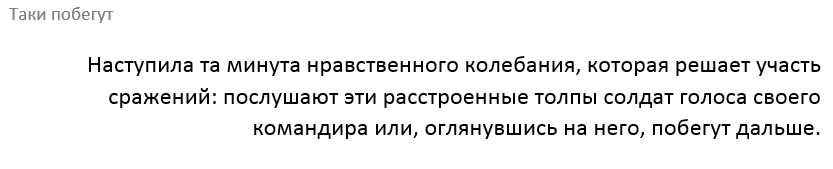
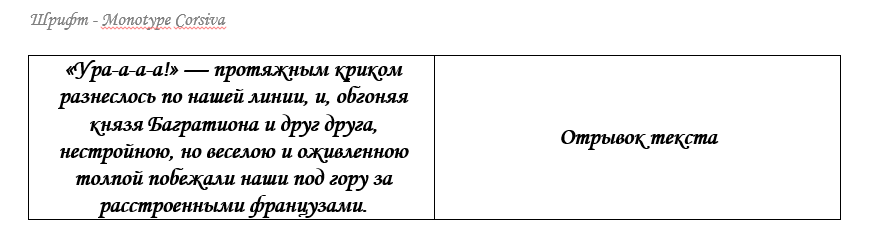
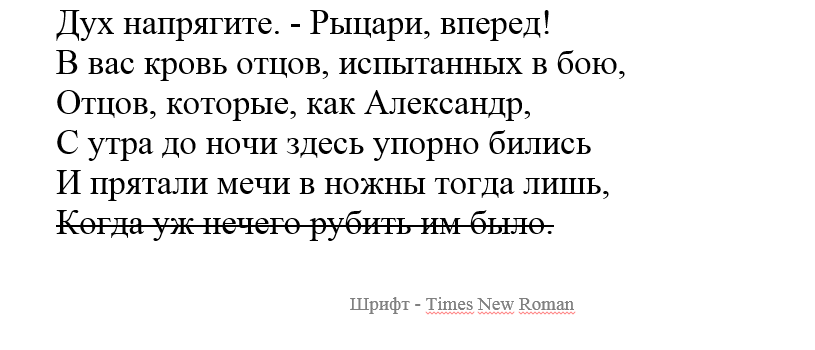
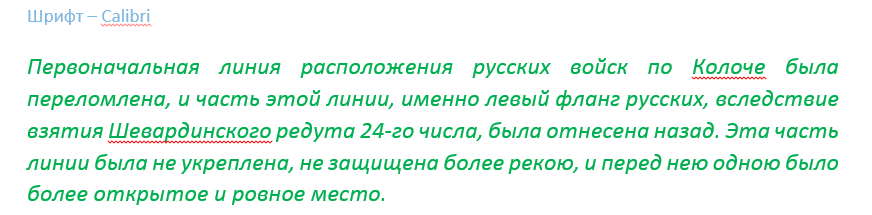
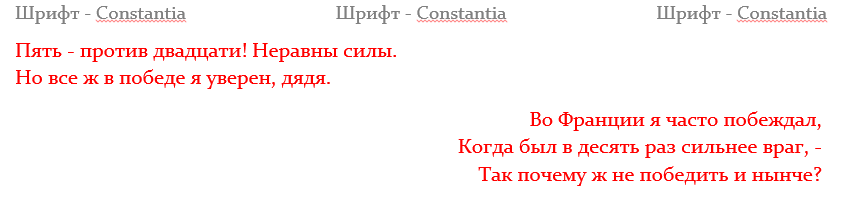
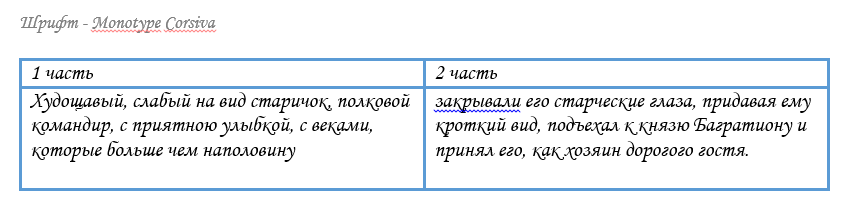
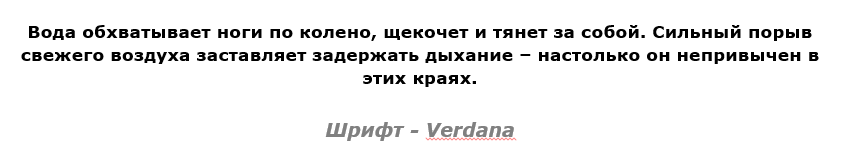
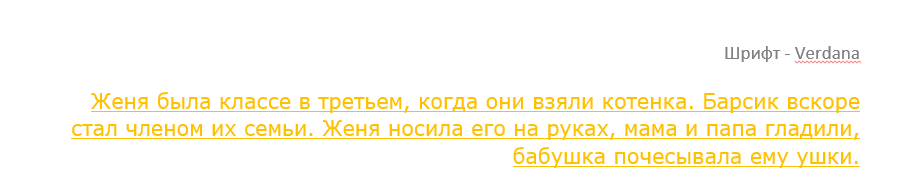
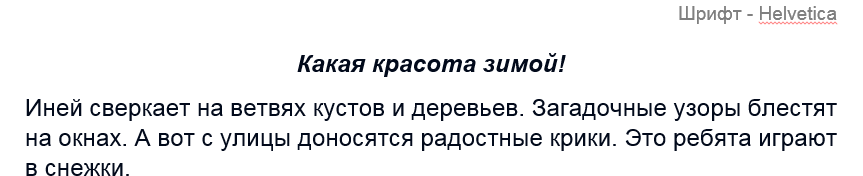
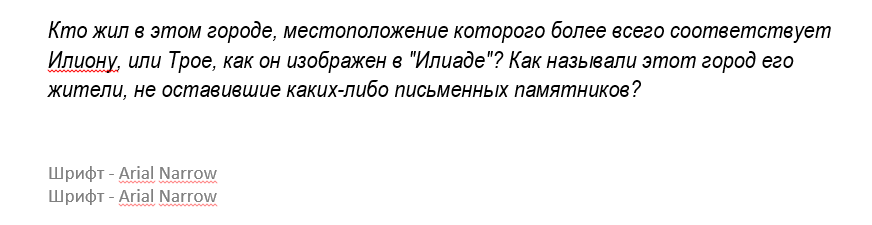
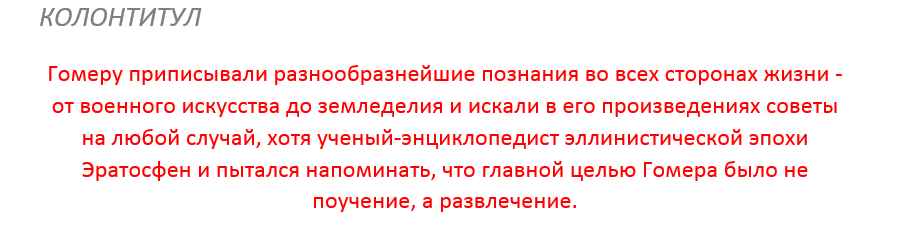
r.Formula = String.Format("=СУММ({0}", adder);

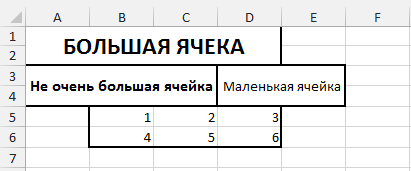
//Добавляем лист в рабочую книгу

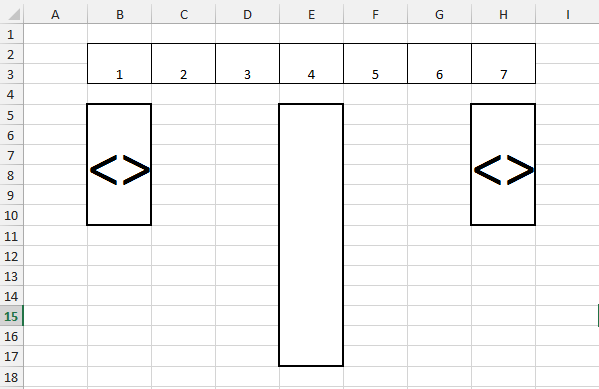
var sh = workBook.Sheets;

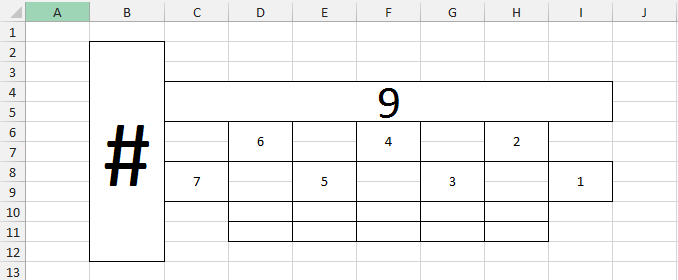
Excel.Worksheet sheetPivot = (Excel.Worksheet)sh.Add(Type.Missing, sh[1], Type.Missing, Type.Missing);

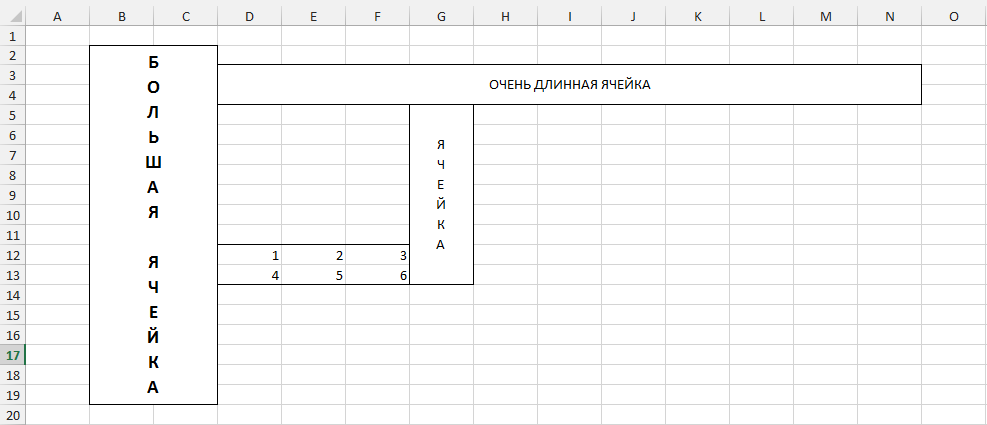
sheetPivot.Name = "Сводная таблица";

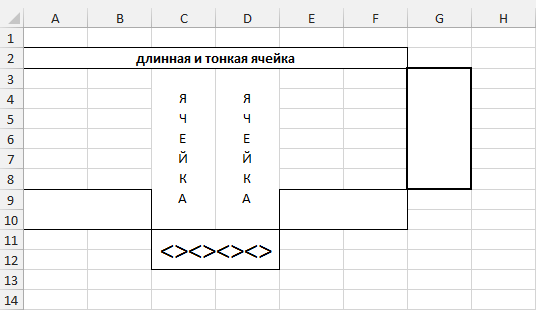
1. Индивидуальное задание, повторите следующие документы (задание необходимо выполнить исходя из номера в журнале). Если не указан шрифт то он по Calibri, обычный текст 14 размера, колонтитул 12 размера:
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 
12. 
13. 
14. 
15. 
16. 
17. Дополнительное задание.
    1. Прокомментировать код;
    2. Выберите любой из этих вариантов Excel и повторите кодом программы.











**Вывод:** При выполнении лабораторной работы, в которой …описываете что делали… были использованы следующие компоненты: …, из них дополнительно изучения требовали: …; для реализации функционала программы необходимо было применить классы: …; дополнительно в лабораторной работе было разобрано (применено, изучено) …. Основные проблемы при выполнении возникли с ….