

ЛАБОРАТОРНАЯ РАБОТА №5

Меню, команды, диалоговые окна

1 Цель работы

Получить навыки работы с меню, панелью инструментов и строкой статуса.

2 Постановка задачи

Создать графический редактор. В окне документа, в точке, где пользователь щелкнул левой клавишей мыши, должна появляться графическая фигура, заданная в индивидуальном задании.

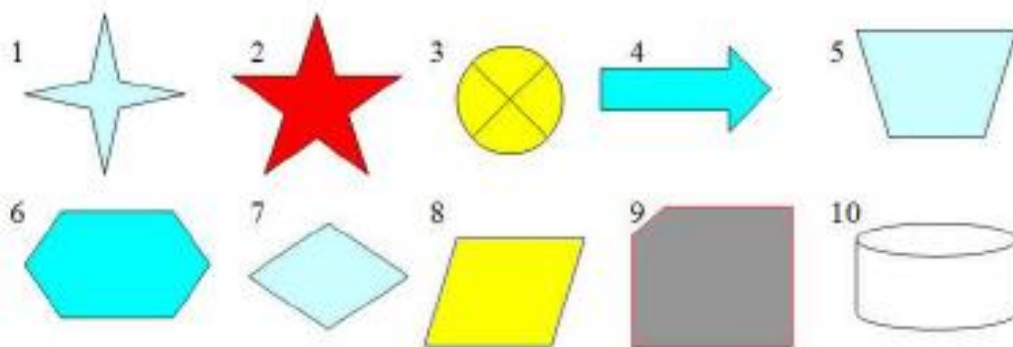
В приложении предусмотреть **диалоговое окно** для выбора толщины линий фигуры, выбора цвета фона и цвета линий фигуры, диалог отображения сведений о программе. Отображение окна диалога реализовать через меню. Продублировать выбор параметров в панели инструментов.

В приложение добавить чтение и запись сформированного пользователем изображения в файл. Файл должен содержать координаты, цвет линий и цвет фона, значение толщины линий. Сохранение файла должно быть доступно **только при наличии фигур в документе**.

Сохранение файла реализовать с помощью механизма **команд** WPF

В заголовке окна выводить название файла, а в панели статуса - координаты мыши.

3 Индивидуальные задания



4 Рекомендации к выполнению задания

4.1 Выбор файла для сохранения данных

Для выбора места сохранения файла используйте стандартный диалог `SaveFileDialog` (необходимо подключить библиотеку `Microsoft.Win32`):

```
SaveFileDialog sfd = new SaveFileDialog();  
sfd.Filter = "Файлы (dat)|*.dat|Все файлы|*.*";  
var result = sfd.ShowDialog();  
if (result == true)  
    { . . . }
```

4.2 Использование механизма команд

Присоедините нужную команду к пункту меню:

```
<MenuItem Header="Save" Command="Save"></MenuItem>
```

Зарегистрируйте команду в конструкторе окна:

```
CommandBinding binding = new CommandBinding(ApplicationCommands.Save);  
binding.Executed += Save_Executed;  
binding.CanExecute += Save_CanExecute;  
this.CommandBindings.Add(binding);
```

В обработчике события `Save_CanExecute` укажите условие возможности выполнения команды. Например, можно использовать свойство `Children.Count` контейнера, в котором производится рисование:

```
private void Save_CanExecute(object sender, CanExecuteRoutedEventArgs e)  
{  
    e.CanExecute = drawingArea.Children.Count != 0;  
}
```