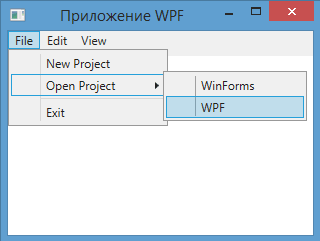
### Menu

Данный элемент служит для создания стандартных меню:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | <Menu Height="25" VerticalAlignment="Top">      <MenuItem Header="File">          <MenuItem Header="New Project" ></MenuItem>          <MenuItem Header="Open Project" >              <MenuItem Header="WinForms"></MenuItem>              <MenuItem Header="WPF" ></MenuItem>          </MenuItem>          <Separator />          <MenuItem Header="Exit" ></MenuItem>      </MenuItem>      <MenuItem Header="Edit" ></MenuItem>      <MenuItem Header="View" ></MenuItem>  </Menu> |



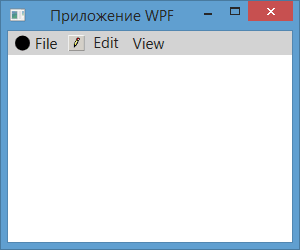
Элемент Menu включает набор элементов MenuItem, которые опять же являются элементами управления содержимым и могут включать другие элементы MenuItem и не только. Также мы можем вложить в меню и другие элементы, которые неявно будут преобразованы в MenuItem. Например:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <Menu Height="25" VerticalAlignment="Top">      <MenuItem Header="File">          <Button Content="Exit" />      </MenuItem>      <MenuItem Header="Edit" ></MenuItem>      <MenuItem Header="View" ></MenuItem>      <Button Content="Кнопка в меню" />  </Menu> |

Также для разделения отдельных пунктов меню можно включать элемент Separator, как в примере выше.

Мы также можем настроить внешний вид отображения, задав свойство **MenuItem.Header** или использовав свойство Icons:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | <Menu Height="25" VerticalAlignment="Top" Background="LightGray">      <MenuItem>          <MenuItem.Header>              <StackPanel Orientation="Horizontal">                  <Ellipse Height="10" Width="10" Fill="Black" Margin="0 0 5 0" />                  <TextBlock>File</TextBlock>              </StackPanel>          </MenuItem.Header>      </MenuItem>      <MenuItem Header="Edit">          <MenuItem.Icon>              <Image Source="C:\Users\Eugene\Documents\pen.png"></Image>          </MenuItem.Icon>      </MenuItem>      <MenuItem Header="View"></MenuItem>  </Menu> |



Чтобы обработать нажатие пункта меню и произвести определенное действие, можно использовать событие Click, однако в будущем мы познакомимся с еще одним инструментом под названием команды, который также широко применяется для реакции на нажатие кнопок меню. А пока свяжем обработчик c событием:

|  |  |
| --- | --- |
| 1 | <MenuItem Header="View"  Click="MenuItem\_Click"></MenuItem> |

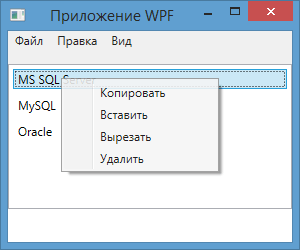
И определим сам обработчик в коде C#:

|  |  |
| --- | --- |
| 1  2  3  4  5 | private void MenuItem\_Click(object sender, RoutedEventArgs e)  {      MenuItem menuItem = (MenuItem)sender;      MessageBox.Show(menuItem.Header.ToString());  } |

### ContextMenu

Класс ContextMenu служит для создания контекстных всплывающих меню, отображающихся после нажатия на правую кнопку мыши. Этот элемент также содержит коллекцию элементов MenuItem. Однако сам по себе ContextMenu существовать не может и должен быть прикреплен к другому элементу управления. Для этого у элементов есть свойство ContextMenu:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | <ListBox Name="list" Height="145">      <ListBoxItem Margin="3">MS SQL Server</ListBoxItem>      <ListBoxItem Margin="3">MySQL</ListBoxItem>      <ListBoxItem Margin="3">Oracle</ListBoxItem>      <ListBox.ContextMenu>          <ContextMenu>              <MenuItem Header="Копировать"></MenuItem>              <MenuItem Header="Вставить"></MenuItem>              <MenuItem Header="Вырезать"></MenuItem>              <MenuItem Header="Удалить"></MenuItem>          </ContextMenu>      </ListBox.ContextMenu>  </ListBox> |



И при нажатии правой кнопкой мыши на один из элементов отобразится контекстное меню.