

ВВЕДЕНИЕ

Если Вы посмотрите на операционные системы и приложения, выпущенные в 95, 98, 2003, 2007 и 2013 годах, то сможете заметить, как менялся их внешний интерфейс. Вместо серых панелей появлялись прозрачные элементы с изменяемым цветом, вместо выпадающего меню появилась лента, в которой отображается меню. Подобное стало возможно благодаря введению в языки программирования новых объектов, получивших название Ribbon-компоненты (введены вместе с выпуском ОС Win Vista), и Metropolis UI-компоненты (начиная с Win 8).

В языке Delphi как в развивающемся языке, вышеперечисленные компоненты, конечно же, присутствуют.

Лабораторная работа №2

Создание простейшего калькулятора с внешним интерфейсом, основанным на Ribbon компонентах.

1. Создайте новый проект (file-new-vlc form application). Выделите появившуюся форму. Смените свойство Caption с Form1 на фамилии, выполняющих проект
2. В tool palette найдите **Ribbon Controls**, содержит пять компонентов: *TRibbon*, *TRibbonComboBox*, *TRibbonSpinEdit*, *TRibbonScreenTipsManager* и *TRibbonScreenTipsPopup*.

Сразу обращаю Ваше внимание на то, что использовать **Ribbon** Вы сможете только в связке с компонентом **ActionManager** со страницы *Additional* палитры компонентов.

3. Поместите на форму *TRibbon*
4. Поместите на форму *TActionManager* с закладки *Additional*;

TImageList со страницы Win32 — здесь мы будем хранить все изображения для панели Ribbon;

5. Выделяем *TRibbon* и в свойстве *ActionManager* указываем *ActionManager1*.
6. По умолчанию Ribbon использует тему оформления Luna. Если Вы хотите изменить внешний вид панели инструментов, то можете сделать это изменяя свойство *Style*. Всего можно использовать 3 стандартные схему: Luna, Obsidian и Silver. В object inspector смените *Style* на Obsidian

7. Теперь изменим свойство *Caption* на калькулятор
8. Переходим к *ActionManager*. В свойстве *FileName* указываем имя файла, в котором будут храниться пользовательские настройки панели. Если файл не указывается, то при работе с программой настройки не сохраняться и при каждом запуске панель инструментов будет выстраиваться по умолчанию.

В свойстве *Images* выбираем наш *ImageList1*. Вы можете также указать, где у вас будут храниться картинки для неактивных кнопок
9. У компонента *ImageList* меняем свойства *Height* и *Width* на значение 32, а свойство *DrawingStyle* устанавливаем в *dsTransparent*.
10. Загружаем в *ImageList* картинки дважды щелкнув по нему. В появившемся окне нажимаем кнопку **ADD** и выбираем картинки (рабочий стол – ико-выделить все картинки- открыть). Нажимаем **OK**.
11. Выделяем *Ribbon*, жмем правую кнопку мыши и в меню выбираем «Add Tab» — таким образом мы создали страницу *RibbonPage1*, теперь переходим на эту страницу, снова жмем правую кнопку и выбираем в меню «Add Group» — добавляем *RibbonGroup1*. Вот в *RibbonGroup*‘ы и добавляются все элементы управления. Как видите иерархия компонента довольно проста. Теперь переходим к *RibbonPage1* и меняем свойство *Caption* на *расчеты*. Выделяем *RibbonGroup1* и меняем свойство *Caption* на *простые математические действия*.
12. Добавьте столько *RibbonPage* и *RibbonGroup* сколько находится в приложении *Project2*, находящемся на рабочем столе в папке *калькулятор/win32/debug*
13. Теперь посмотрите на результат — чего-то не хватает? Конечно не хватает самого главного — кнопки меню. Чтобы добавить кнопку меню выбираем *Ribbon*, жмем правую кнопку мыши и в меню выбираем «Add Application Menu».
14. Теперь настроим вид кнопки меню. Для этого переходим к компоненту *TRibbon* и раскрываем группу свойств «Application Menu», находим там пункт «Icon» и загружаем любую картинку в формате *.ico. По желанию, можете изменить свойство «IconSize» на *isLarge*, *isMedium* или *isSmall*.
15. Сменим иконку проекта: нажмите **CTRL+SHIFT+F11** в появившемся окне откройте вкладку *Application*, *Load Icon* – выберите иконку проекта

16. Делаем двойной клик на *ActionManager* и в открывшемся окне переходим на вкладку «Actions» (Действия).

17. Нажмите на кнопку «New Action». Появилось новое действие Action1, выделяем его

18. Изменим свойства:

- *Caption* – на сложение.
- *ImageIndex* - указать картинку из *ImageList*
- *Hint* - если хотите изменить подсказку

Теперь, чтобы добавить это действие на панель инструментов, просто схватите его мышкой и перетяните на *RibbonGroup*. Теперь выберите в *RibbonGroup* это действие и найдите в Object Inspector'e свойство «Command Properties — Button Size» и измените его на *bsLarge*. Вот теперь наша кнопка выглядит действительно как кнопка.

19. Добавьте столько Action, сколько находится в приложении калькулятор

Проблемы при работе с Ribbon Controls

Как говорится, и в этой бочке мёда нашлась таки ложка дёгтя. Несмотря на свою красоту и функциональность Ribbon Controls в Delphi пока ещё сыроват. Например, присутствует ряд проблем при работе с компонентами в Design-Time — иногда, размещенные на закладке компоненты ни в какую не хотят выделяться мышкой. Вроде руки не кривые, жмешь куда надо, а в Инспекторе появляются свойства совсем не того компонента или Action'a. Но это весьма несущественные проблемки, которые видимо будут исправлены в ближайшем будущем.

Основная проблема заключается в том, что при размещении на панелях большого количества стандартных компонентов Ribbon вдруг перестает напрочь реагировать на любые действия — выдает ошибку доступа к ячейкам памяти и умирает, что в итоге приводит к перезагрузке Delphi. Причём перезагрузка восстанавливает работу только оболочки, а проект так и не оживает.

20. Теперь приступим к созданию калькулятора. Поместите на форму столько кнопок, сколько находится в калькуляторе. Измените их ширину и высоту, а так же Caption

21. Поместите на форму текстовое поле edit (buttons and edit are located in standard tabs)

22. Приступаем к программированию кнопок и событий меню

Часть 2. Программирование

0. Опишем переменные, используемые в программе D: char; x:real; помещаем в раздел var после Form2: TForm2; перед implementation

1. Дважды щелкаем на кнопку с 1 и вписываем edit1.Text:=edit1.Text+'1' это означает, что при нажатии кнопки 1 в текстовом поле появится 1.

2. Для кнопок 2-9 и 0 выполните аналогичные действия

3. Дважды щелкаем на Action сложить и вписываем

D:='+';

x:=StrToFloat(Edit1.Text);

Edit1.Text:="";

Это означает, что символьной переменной присваивают значение +

Переменной вещественного типа x присваивают значение перевода текстовой строки в число из текстового поля. И в текстовом поле убирают текущую строку

4. Для Action Вычесть, Умножить и Разделить выполните аналогичные действия

5. Выберите Action Очистить и дважды щелкните по нему
edit1.Clear;

6. Выберите Action Равно

if D='+' then edit1.Text:=floattostr(x+strtofloat(edit1.Text));

if D='-' then edit1.Text:=floattostr(x-strtoffloat(edit1.Text));

Это означает, что если выбран символ сложить, то к числу, хранящемуся в переменной x прибавится число, находящееся в данный момент в текстовом поле и в текстовое поле выведется текст, показывающий результат операции.

Для умножения и деления запишите условия самостоятельно

7. Добавьте новый Action, назвав его Выход, поставив картинку, перетащите его под кнопку меню. В код программы допишите действие закрыть (Close;)