**Панфилов Евгений**

**Вариант 02**

**В каждой лабораторной нужно ответить на контрольные вопросы**

**ЛАБОРАТОРНАЯ РАБОТА 1**

**Тема: Первое приложение на DELPHI**

**Цель работы**: Ознакомиться с основными приемами работы в среде DELPHI

**Задание 1. Выполните последовательно следующие действия**

1. Запустите *Delphi*. (*Пуск → Программы → Borland Delphi 5*)
2. Выберите в [*Палитре компонентов*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link01.htm) закладку [*Standard*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link35.htm).
3. Перенесите с Палитры компонентов на [*форму*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link03.htm) компоненты [*TLabel*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r12) и [*TButton*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r1). Для этого поочередно подводите курсор мыши к компонентам, читая подсказки, до тех пор, пока не появится *Label*. Выберете его, нажав левую кнопку мыши, а затем переместите указатель на форму и нажмите еще раз на кнопку мыши. Помещенный на форму компонент будет иметь имя *Label1*.
4. Проделайте то же самое, только с компонентом *TButton*.
5. Напишите обработчик события *Button1Click*. Для этого выберите на форме *Button1* и щелкните на него дважды. Вы окажетесь в [*окне редактирования*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link32.htm). Приведите обработчик события нажатия кнопки к следующему виду:

**procedure** TForm1.Button1Click(Sender: TObject);  
**begin**  
\_\_\_Label1.Caption:= 'Здравствуй, мир!';  
**end**;

При нажатии на кнопку вызывается обработчик события *Form1.Button1Click(Sender: TObject)*, в котором мы меняем свойство ***Caption*** у объекта *Label1*.

1. Создайте новый каталог с именем MyProject. Для этого выполните команду главного меню [*File → Save Project As*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r1) и с помощью пиктограммы “Новый каталог” создайте новый каталог. Сохраните проект в созданном каталоге, при этом предлагается сохранить два файла (с расширением pas и prj). Можно набирать произвольные имена.
2. Запустите на выполнение созданное Приложение. Для этого выберете в главном меню пункт [*Run → Run*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r2), или нажмите клавишу ***F9***.
3. Нажмите на кнопку в запущенном Приложении и посмотрите, что из этого получилось. Закройте Приложение.   
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы (шаг 1 - 8)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film26.exe)
4. Поместите на форму компонент [*TEdit*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r5) (см. п.3) и [*TButton*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r1).
5. Приведите обработчик события нажатия кнопки *Button2* к следующему виду (вызов обработчика см. в п.5): (Переключение между [*формой*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link30.htm), [*инспектором объектов*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link31.htm) и [*окном редактирования*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link32.htm) - ***F12***).

**procedure** TForm1.Button2Click(Sender: TObject);  
**begin**  
\_\_\_Label1.Caption:= Edit1.Text;  
**end**;

1. Измените свойство *Edit1.Text* на “Привет, студент!!!”. Для этого выберете на форме объект *Edit1*. Перейдите в [*Object Inspector*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link31.htm), в закладке [*Properties*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link36.htm) найдите свойство *Text* и замените его.
2. Запустите на выполнение ваше Приложение. Попробуйте нажимать кнопку, вводя разное значение в строку редактирования. Закройте Приложение.   
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы (шаг 9 - 12)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film27.exe)
3. Добавьте компоненты [*TLabel*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r12) и [*TButton*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r1). Установите свойство объекта *Label2.Caption* равным 1 (см. п.3).
4. Приведите обработчик события нажатия кнопки *Button3* к следующему виду (см. п.5):

**procedure** TForm1.Button3Click(Sender: TObject);  
**Var** i: integer;  
**Begin**  
\_\_\_i:= StrToInt(Label2.Caption);  
\_\_\_Label2.Caption:= IntToStr(i+1);  
**end**;

{ *IntToStr, StrToInt* – *функции, преобразующие целый тип в строковый и строковый в целый соответственно* }

1. Разместите на форме все кнопки вертикально, друг под другом. Выделите все кнопки и выровняйте их размеры и положение с помощью команд [*Edit → Size*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r4) и [*Edit → Align.*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r4)
2. Сохраните и запустите приложение из среды. Проверьте работоспособность всех кнопок формы.

Должно получиться подобное приложение

|  |
| --- |
| Image142.gif |
| ***Рисунок 1.1****Работа приложения* |

D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы (шаг 13 - 16)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film28.exe)

**Задание 2.**

1. Создайте новое приложение (команда главного меню [*File → New → Application*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r1)).
2. Добавьте на форму три компонента *TButton*, три компонента *TLabel* и один компонент *TEdit*.
3. Выравняйте размеры и положение компонентов на форме. Измените названия кнопок и текст в строке редактирования. Строка редактирования должна содержать следующую информацию: название группы и последние две цифры шифра студента. Форма может выглядеть примерно таким образом.

|  |
| --- |
| Image143.gif |
| ***Рисунок 1.2****Примерный вид формы* |

1. Запрограммируйте действие каждой из кнопок. При нажатии на кнопку в соответствующей метке должны появляться фамилия, имя и отчество студента.

|  |
| --- |
| Image144.gif |
| ***Рисунок 1.3****Работа приложения* |

**Контрольные вопросы**

1. Назовите основные составляющие среды DELPHI и их назначение.
2. Назовите состав и назначение пунктов главного меню системы.
3. Каково назначение Инспектора объектов?
4. Из каких основных файлов состоит проект приложения?
5. Как производится размещение, перемещение, изменение размеров компонентов на форме?
6. Как запустить приложение на выполнение из среды DELPHI?
7. Поясните назначение и основные свойства и события компонентов формы приложения.
8. Поясните тексты методов приложения.

**ЛАБОРАТОРНАЯ РАБОТА 2**

**Тема: Определение нажатой клавиши и нажатого символа.**

**Цель работы**: Ознакомиться с основными клавиатурными событиями визуальных компонентов.

**Задание 1. Выполните следующие действия**

1. Разместите на форме 6 объектов типа [*Label*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r12) и одну кнопку с растровым изображением (компонент [*BitBtn*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r4) на закладке Additional палитры компонентов). Установите у кнопки свойство *Kind* равным *bkClose*. Тогда при нажатии на эту кнопку приложение будет закрываться.
2. При создании одинаковых по размеру объектов пользуйтесь буфером обмена или командой [*Edit → Size*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r4). Для выравнивания положения объектов на форме используйте команду [*Edit → Align*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r4).
3. Для всех объектов типа *Label* установить свойство *AutoSize* равным *False*, свойство *Aligment* – *taCenter*. Для меток слева установите нужные свойства *Caption, Font*. Для меток справа удалите свойство *Caption*, измените свойство *Font* по своему усмотрению.
4. Для всех объектов типа *ТLabel* оформите границы, используя такие декоративные элементы, как фаски (компонент [*ТBevel*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link38.htm) на закладке палитры компонентов Additional). У объектов типа *ТBevel* установите значения свойств *Shape* и *Style*.

В результате окно будет выглядеть примерно так

|  |
| --- |
| Image145.gif |
| ***Рисунок 2.1****Примерный вид окна* |

1. Установите для **формы** значение свойства *KeyPreview* равным *True* для того, чтобы форма могла обрабатывать события для клавиатуры.
2. В процедуре обработки события *OnKeyDown* для **формы** необходимо выполнить следующие действия. Показать значение переменной Key (код виртуальной клавиши) в качестве нажатой клавиши. Для преобразования кода в строковую переменную используйте функцию [*IntToStr*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lab_raboti.htm#r1). Проверить были ли нажаты клавиши Shift, Alt, Ctrl (принадлежат ли элементы ssShift, ssAlt, ssCtrl множеству Shift, которое передаётся процедуре в качестве параметра). Создать строку из комбинации нажатых клавиш (например, Shift+Alt+Ctrl, если были нажаты все три клавиши) и поместить её на форму в соответствующее место.  
   Например, для проверки нажатия клавиши Alt можно использовать следующий код

**procedure** TForm1.FormKeyDown(Sender: TObject; var Key: Word; Shift: TShiftState);  
**begin**  
**if** ssAlt **in** Shift **then** Label6.Caption:= 'Alt';  
**end**;

1. Процедура обработки события *OnKeyPress* получает значение параметра Key, равное символу ASCII. Для отображения символа в объекте Label2 используйте Label2.Caption:= Key.
2. В процедуру обработки события формы *OnKeyUp* добавьте операторы очистки заголовков объектов типа *Label* из среднего столбца. Например, Label2.Caption:= ''.

При нажатии клавиши с буквой A (англ. раскладка) будет отображаться нажатый символ «а» и код нажатой клавиши 65.

|  |
| --- |
| Image147.gif |
| ***Рисунок 2.2****Работа приложения* |

При нажатии комбинации клавиш Shift + A (англ. раскладка) будет отображаться нажатый символ «A», код нажатой клавиши 65 и название клавиши Shift.

|  |
| --- |
| Image148.gif |
| ***Рисунок 2.3****Работа приложения* |

D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film29.exe)

**Задание 2.**

1. Добавьте на форму еще один компонент *ТLabel*. При нажатии определенной комбинации клавиш в метке должны появляться название группы и фамилия студента. Комбинация клавиш выбирается по последней цифре шифра студента.
   1. Ctrl + Shift
   2. Alt + Q
   3. Ctrl + 5
   4. Alt + 5
   5. Shit + 1
   6. Shift + L
   7. Shift + B
   8. Ctrl + 9
   9. Alt + 9
   10. Alt + Shift

**Контрольные вопросы**

1. Поясните назначение и основные свойства и события компонентов формы приложения.
2. Поясните тексты методов приложения.
3. Какие события используются для получения комбинации нажатых или отпущенных клавиш?
4. Какое событие позволяет определить нажатый символ?
5. Для чего используется компонент ТBevel?

**ЛАБОРАТОРНАЯ РАБОТА 3**

**Тема: Создание главного меню.**

**Цель работы**: Научится использовать в приложении главное меню.

**Задание 1. Выполните следующие действия**

1. Поместите на форму компонент [*MainMenu*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r7) (закладка Standart на Палитре компонентов). Вызовите Дизайнер Меню, дважды щелкнув мышкой по компоненту на форме. Затем создайте меню как показано на рисунке 85. Для создания подменю можно воспользоваться командой Create SubMenu в контекстном меню, которое вызывается правой кнопкой мыши. После разработки закройте Дизайнер Меню. В верхней части формы теперь можно видеть разработанное меню.

|  |
| --- |
| Image149.gif |
| ***Рисунок 3.1****Вид меню при конструировании* |

1. При нажатии на отдельный пункт меню должно появляться информационное окно, которое содержит значение соответствующей характеристики формы. Двойной щелчок по пункту меню дает заготовку обработчика. Например, нажатие пункта *Ширина* можно обработать следующим образом

**procedure** TForm1.N2Click(Sender: TObject);  
**var** s: **string**;  
**begin**  
\_\_\_s:= format('Width %d', [width]);  
\_\_\_ShowMessage(s);  
**end;**

1. Аналогичным образом обработайте остальные пункты меню, создавая информационные окна с соответствующей информацией о форме.
2. Для быстрого вызова команд можно использовать так называемые быстрые клавиши. Для этого для каждого пункта меню установите свойство *ShortCut*, выбрав значение из списка. При этом нужно следить, чтобы быстрые клавиши не повторялись во избежание коллизий.
3. Поместите на форму компонент [*ImageList*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link39.htm) (закладка Win32 на Палитре компонентов). Дважды щелкнув на компонент, можно вызвать редактор этого компонента. Используйте кнопку Add, чтобы добавить иконки (файлы с расширением ico). Можно использовать любые готовые иконки либо создать их самостоятельно. Для этого нужно вызвать редактор изображений (пункт главного меню [*Tools → Image Editor*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link40.htm)). Далее создайте новое изображение [*File → New → Icon file ( .ico)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link41.htm). Нарисуйте икону, используя различные цвета и инструменты на панели. Сохраните иконку в свою директорию. Рекомендуется добавить в *ImageList* 5-6 различных иконок.
4. Выделите компонент *MainMenu1* и установите свойство *Images* равным *ImageList1*. После этого для каждого пункта меню установите свойство *ImageIndex*, выбрав из списка нужную иконку. Меню может выглядеть так.

|  |
| --- |
| Image153.gif |
| ***Рисунок 3.2****Вид меню* |

1. Для каждого из пунктов меню *Размеры* установите свойство *Break* равным *mbBarBreak*. Запустите приложение, проверьте его работу.  
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film30.exe)

**Задание 2.**

1. Создайте новое приложение. Добавьте на форму компонент [*MainMenu*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r7) и приведите его к следующему виду.

|  |
| --- |
| D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image154.gifImage155.gif |
| ***Рисунок 3.3****Вид меню* |

1. Для каждого пункта меню укажите иконку и горячую комбинацию клавиш.
2. Каждый пункт меню запрограммируйте так, чтобы при вызове появлялось информационное окно с соответствующей информацией. При вызове пункта меню «Выход» приложение должно завершать свою работу.

**Контрольные вопросы**

1. Поясните назначение и основные свойства и события компонентов формы приложения.
2. Поясните тексты методов приложения.
3. Как вызвать Дизайнер Меню?
4. Как создать подменю?
5. Каким образом можно поместить иконки для пунктов меню?
6. Как установить быстрые клавиши для пунктов меню?
7. На что влияет свойство Break пункта меню?

**ЛАБОРАТОРНАЯ РАБОТА 4**

**Тема: Создание простейшего текстового редактора.**

**Цель работы**: Освоить использование основных диалоговых компонентов.

**Задание 1. Выполните следующие действия**

1. Поместите на форму компонент [*ТMainMenu*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r7). Структура меню должна иметь вид:

|  |
| --- |
| D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image156.gifImage157.gif |
| ***Рисунок 4.1****Структура меню* |

1. При выборе пункта *Автор* должно появляется информационное окно с данными автора. При выборе пункта *Выход* приложение завершает свою работу.
2. Запустите приложение. Проверьте работу приложения.
3. Поместите на форму компонент всплывающее меню – [*ТPopupMenu*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r13). Всплывающее меню появляется в работающем приложении при щелчке правой кнопкой мыши по форме. Это меню должно содержать пункты *Автор, Сохранить, Выход* с назначенными им клавишными эквивалентами для быстрого выполнения пунктов меню.
4. Для того, чтобы форма могла использовать объект *PopupMenu1*, определите значение свойства *PopupMenu* из выпадающего списка для **формы**.
5. Для создания обработчиков событий, поступающих от всплывающего меню, для пункта *Автор* определите значение события *OnClick*, выбрав из раскрывающегося списка название процедуры, которое использовалось для пункта *Автор* главного меню. Аналогичным образом определите значение события *OnClick* для пункта *Выход*. Таким образом, события для разных объектов будут совместно использовать один и тот же обработчик события.
6. Запустите приложение и проверьте работу всплывающего меню.  
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы шаг(1 - 7)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film31.exe)
7. Поместите на форму компонент [*ТLabel*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r12). Присвойте свойству *Caption* значение - Имя файла.
8. Поместите на форму компонент [*ТEdit*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r5) под компонентом *ТLabel*. Очистите значение свойства *Text* объекта *Edit1*. В дальнейшем здесь будет имя редактируемого файла.
9. Поместите на форму объект [*Memo*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r5) (многострочный редактор), измените его размер и выровняйте визуальные компоненты, расположенные на форме, используя в главном меню Delphi [*Edit → Align*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r4). Уберите у свойства *Lines* объекта *Memo1* его значение. Установите вертикальную и горизонтальную полосы прокрутки, используя свойство *ScrollBar* объекта Мемо1.

Теперь приложение будет выглядеть так

|  |
| --- |
| Image158.gif |
| ***Рисунок 4.2****Вид приложения* |

1. Поместите на форму компоненты [*TOpenDialog*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r14) и [*TSaveDialog*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r14), расположенные на закладке [*Dialogs*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link18.htm) палитры компонентов. Для объектов *OpenDialog1* и *SaveDialog1* определите свойства *DefaultExt, Filter, FilterIndex, Title*, которые имеют следующий смысл

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Свойство** | **Назначение** | | ***FileName*** | Содержит маршрут поиска и выбранный файл при успешном завершении диалога. | | ***Filter*** | Используется для фильтрации (отбора) файлов, показываемых в диалоговом окне. | | ***InitialDir*** | Содержит имя начального каталога. | | ***DefaultExt*** | С помощью свойства *DefaultExt* формируется полное имя файла, если при ручном вводе пользователь не указал расширение. В этом случае к имени файла прибавляется разделительная точка и содержимое этого свойства. | | ***Title*** | Содержит заголовок диалогового окна. | |

1. Для активизации диалогового компонента *OpenDialog* используйте его метод *Execute* в обработчике события *OnClick* для пункта меню *Открыть*. Если пользователь закрывает диалоговое окно, щелкнув на кнопке OK, то функция *Execute* возвращает значение *True* и содержимое файла загружается в многострочный редактор.  
   Например,

**procedure** TForm1.N2Click(Sender: TObject);  
**begin**  
\_\_\_**If** OpenDialog1.Execute **then**  
\_\_\_\_\_\_Memo1.Lines.LoadfromFile(OpenDialog1.FileName);  
**end**;

Тогда при выборе пункта *Открыть* будет открываться такое окно

|  |
| --- |
| Image159.gif |
| ***Рисунок 4.3****Диалоговое окно открытия файла* |

1. Для проверки существования файла используйте функцию *FileExists*, поскольку открываться может только уже существующий файл.
2. Перед загрузкой файла в компонент *Memo* необходимо очистить его. Это можно сделать, используя метод *Memo1.Clear* для очистки окна *Memo1*
3. Имя загружаемого файла сохраняется в свойстве *FileName* компонента *OpenDialog* и должно отобразиться в строке редактирования под надписью «Имя файла».
4. Для обработки события *OnClick* пункта *Открыть* всплывающего меню подключите ту же процедуру, что используется для пункта *Открыть* главного меню.
5. Аналогичные действия проделайте и с пунктом *Сохранить*, при этом проверку на существование файла делать не нужно, поскольку если такой файл не существует, то приложение автоматически его создаст.
6. Содержимое редактора *Мемо1* загружается в файл методом *SavetoFile*. При сохранении файла в диалоговом окне должно предлагаться имя, заданное в компоненте *TEdit*.
7. Для того, чтобы курсор установился в окно редактора, необходимо после загрузки файла в окно редактора установить значение свойства *ActiveControl* равным *Мемо1*. Поэтому в процедуру обработки события *OnActivate* формы добавьте следующий оператор:

Form1.ActiveControl:= Memo1;

1. Запустите приложение и проверьте его работу.  
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы шаг(8 - 20)*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film32.exe)
2. Создайте фильтры файлов, используя встроенный редактор свойства *Filter* диалоговых компонентов (вариант выбирается по последней цифре шифра студента)
   1. исполняемые файлы (exe), все файлы, текстовые файлы (txt)
   2. doc-файлы, все файлы, pas-файлы
   3. asm- файлы, исполняемые файлы (exe), bat- файлы
   4. все файлы, bmp-файлы, текстовые файлы (txt)
   5. pas-файлы, cpp-файлы, все файлы
   6. исполняемые файлы (exe), все файлы, текстовые файлы (txt)
   7. doc-файлы, все файлы, pas-файлы
   8. asm- файлы, исполняемые файлы (exe), bat- файлы
   9. все файлы, bmp-файлы, текстовые файлы (txt)
   10. pas-файлы, cpp-файлы, все файлы

**Контрольные вопросы**

1. Поясните назначение и основные свойства и события компонентов формы приложения.
2. Поясните тексты методов приложения.
3. Какие компоненты используются для организации диалога при сохранении и открытии файлов?
4. Какой метод активизирует диалоговые компоненты?
5. Что такое контекстное меню?
6. ККак осуществляется совместное использование одного обработчика различными компонентами?

**ЛАБОРАТОРНАЯ РАБОТА 5**

**Тема: Создание заставки текстового редактора. Создание таймера.**

**Цель работы**: Научиться использовать различные виды форм.

**Порядок выполнения работы:**

*Создание заставки*

1. Откройте ранее созданный текстовый редактор. Создайте с помощью команды меню [*File → New → Form*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r1) новую форму *Form2*. Эта форма будет использоваться в качестве заставки перед запуском главного окна приложения. Для новой формы удалите значение свойства *Caption*. Свойству *BorderStyle* присвойте значение *bsNone* (результат увидите только при запуске программы). Для удаления кнопок минимизации, максимизации формы и системного меню установите у свойства *BorderIcons* значения подсвойств *biMinimize, biMaximize, biSystemMenu* равными *False* (результат увидите только при запуске программы).
2. Также для новой формы свойство *Enabled* установите равным *False* для того, чтобы заблокировать обработку событий и лишить возможности пользователя управлять окном с помощью клавиатуры или мыши.
3. Сохраните проект, определив имя второго программного модуля проекта.
4. Выполните команду [*Project → Options*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r6). В предложенном окне выберите закладку страницы Forms. Переместите Form2 из списка AutoCreate в список Available для рационального использования памяти и ресурсов.
5. Преобразуйте исходный файл проекта. Для этого чтобы открыть файл проекта выполните команду [*Project → View Sourсe*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.htm#r6) и измените операторы между **begin** и **end** следующим образом:

Form2:= TForm2.Create(Application);  
*{ Этот оператор создает объект формы заставки и присваивает его адрес переменной Form2, которая определена в программном модуле заставки }*  
  
Form2.Show;  
Form2.Update;  
*{ Отображение объекта из памяти на экране и обновление его содержимого }*  
  
Application.CreateForm(TForm1,Form1);  
Form2.Hide;  
Form2.Free;  
*{ Скрытие формы заставки и освобождение занимаемой ею памяти }*  
  
Application.Run;

1. Для того, чтобы заставка оставалась на экране несколько секунд, для обработчика события *OnCreate* главной формы *Form1* добавьте следующие операторы:

CurrentTime:= GetTickCount **div** 1000;  
*{ Процедура CurrentTime возвращает время работы Windows в миллисекундах}*  
  
**while**((GetTickCount **div** 1000) < (CurrentTime + 4)) **do**;  
*{ Организация паузы в 4 секунды}*

Объявите переменную CurrentTime типа longint в разделе **var** текущего модуля.

1. Поместите на форму заставки компонент [*Image*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link42.htm) из закладки Additional палитры компонентов. Вызовите встроенный редактор свойства *Picture*. Щелкнув по кнопке Load, выберите файл с картинкой для загрузки картинки в окно Picture Editor. (Можно загрузить стандартные картинки C:\Program Files\Common Files\Borland Shared\Images\…). Для того, чтобы растровое изображение внутри объекта *Image1* занимало всю клиентную область, требуется изменить значение свойства *Strech* образа на *True*.
2. Поместите на форму заставки справа и снизу от рисунка два компонента [*Label*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec7.htm#r12) из закладки Standard палитры компонентов. Отобразите на них информацию об авторе приложения и название приложения, используя различный размер, цвет и начертание шрифта.
3. Запустите приложение и проверьте его работу.  
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы ( создание заставки )*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film33.exe)

*Создание таймера*

1. Добавьте к проекту новую форму *Form3*. Переместите *Form3* из списка AutoCreate в список Available для рационального использования памяти и ресурсов.
2. Измените границы окна у *Form3*, сделайте их невидимыми, уберите значки максимизации, минимизации, системного меню.
3. Поместите на форму *Form3* компонент *Label*, свойству *Caption* этого компонента присвойте значение 00:00:00. Выберите для своих часов шрифт, размер и стиль.
4. Преобразуйте исходный файл проекта. Для этого выполните команду [*Project → View Sourсe*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\lec3.hmt#r6) и добавьте перед строкой *Application.Run* операторы

Form3:=TForm3.Create(Application);  
Form3.Show;  
Form3.Update;

1. Поместите на форму *Form3* компонент [*Timer*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\link43.htm) из закладки System. В обработчик события *OnTimer* объекта *Тimer1* добавьте следующую строку:

Label1.Caption:=TimeToStr(Time);  
*{ TimeToStr преобразует текущее время в строку }*

1. Для события *OnCreate* формы *Form3* выберите обработчик *Timer1Timer*, который был создан для компонента *Timer1*. Тогда текущее время будет записано в метку до появления окна.
2. Для формы *Form3* установите свойство *FormStyle* равным *fsStayOnTop* для того, чтобы часы были расположены всё время поверх других окон. Кроме того, установите свойство *Visible* в *True*.
3. Запустите приложение и проверьте его работу.  
     
   D:\СИБГУТИ\6семестр\Языки программирования высокого уровня\course173_big\doc\img\Image777.gif[*Выполнение лабораторной работы ( создание таймера )*](file:///D:\СИБГУТИ\6семестр\Языки%20программирования%20высокого%20уровня\course173_big\doc\video\film34.exe)

**Контрольные вопросы**

1. Поясните назначение и основные свойства и события компонентов формы приложения.
2. Поясните тексты методов приложения.
3. В каком свойстве содержится заголовок формы?
4. Назовите возможные значения свойства BorderStyle. Поясните их смысл.
5. Для чего используется свойство BorderIcons?
6. На что влияет свойство Formstyle?
7. Когда совершается событие OnTimer объекта Тimer?