Лабораторная работа №1

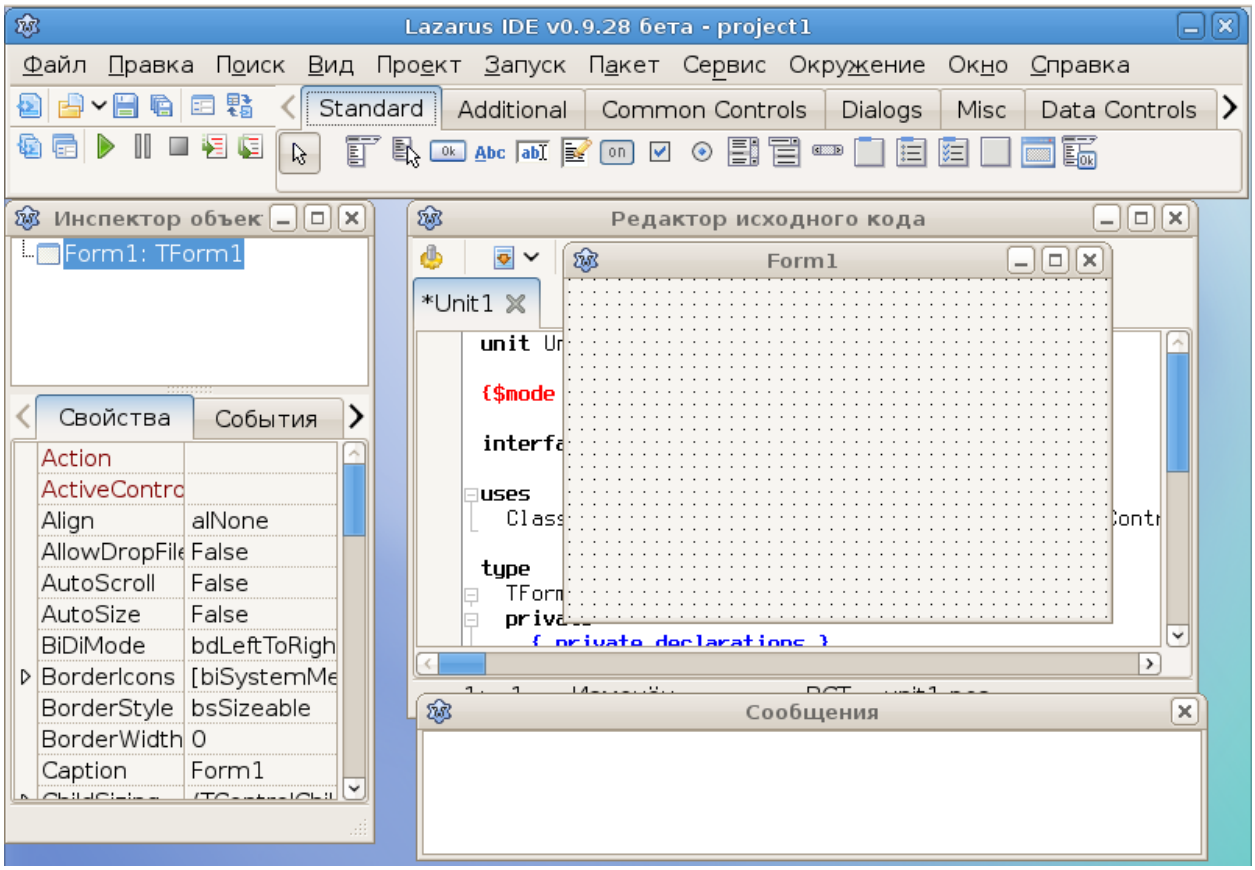
Введение. Знакомство с интегрированной средой Lazarus

Цель: Знакомство с интегрированной средой разработки программного обеспечения Lazarus. Создание нового проекта.

Справочный материал.

1. Пользовательский интерфейс Lazarus

Интегрированная среда разработки Lazarus представляет собой многооконную систему, вид (пользовательский интерфейс) которой может различаться в зависимости от настроек. После загрузки интерфейс Lazarus выглядит следующим образом:

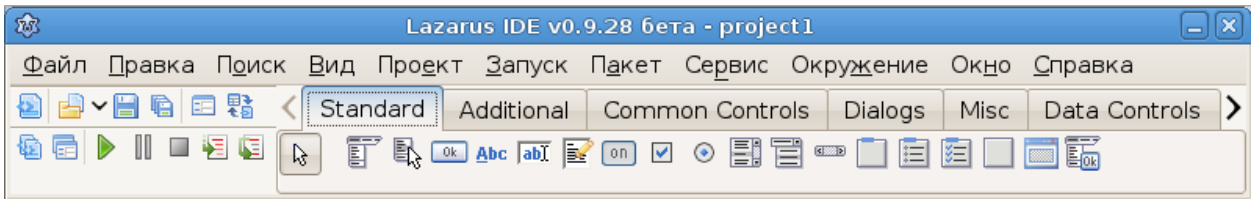


и первоначально включает:

1. Главное окно;
2. окно Инспектора объектов;
3. окно Формы, или Конструктора формы;
4. окно Редактора кода;
5. окно Сообщений.

Главное окно Lazarus содержит:

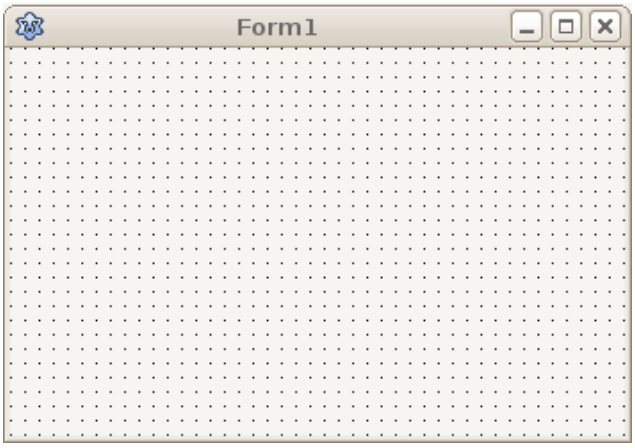
1. главное меню;
2. панели инструментов;
3. палитру компонентов.



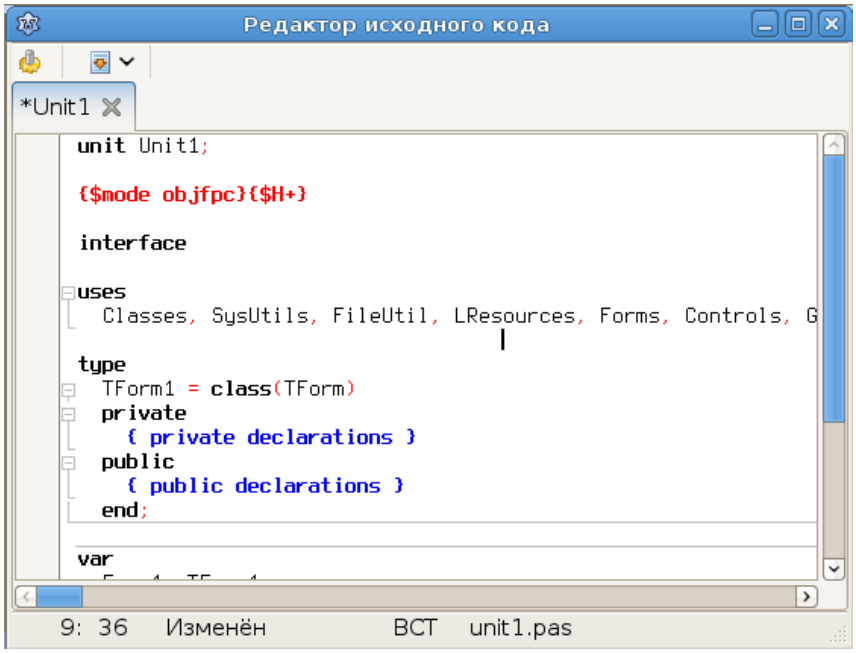
Главное меню содержит обширный набор команд для доступа к функциям Lazarus.

Панели инструментов находятся под главным меню в левой части главного окна и содержат кнопки быстрого доступа к наиболее частым командам главного меню.

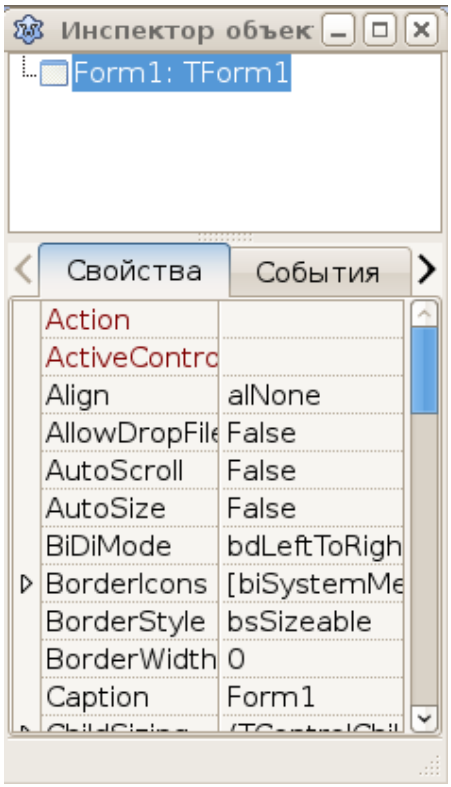
Палитра компонентов находится под главным меню в правой части главного окна и содержит множество компонентов, размещаемых в создаваемых формах. *Компоненты* являются своего рода строительными блоками, из которых конструируются формы приложения.



Окно формы (или **Конструктора формы)** первоначально находится в центре экрана и имеет заголовок **Form1**. В нем выполняется проектирование формы, путем размещения на форме различных компонентов.



Окно редактора кода после запуска системы программирования находится под (или поверх) окна Формы. Редактор кода представляет собой обычный текстовый редактор, с помощью которого можно редактировать текст программы (**листинг**).



Окно Инспектора объектов расположено под Главным окном в левой части экрана и состоит из двух частей: в верхней части окна Инспектора объектов располагается список всех созданных объектов (форма и все компоненты, которые расположены на форме); в нижней части отображены свойства и события объектов для текущей формы или компонента. Вкладка *Свойства* отражает свойства выбранного компонента или формы, а вкладка *События* – процедуры, которые должны быть выполнены при возникновении указанного события.

2. Создание нового проекта

Для создания нового проекта необходимо воспользоваться пунктом «Создать» меню «Файл» главного меню и выбрать пункт «Project Application».

3. Объекты формы и их свойства

На палитре компонентов располагаются различные объекты пользовательского интерфейса. Наиболее востребованные объекты:

- Кнопка – **TButton**  ;
- Текстовое поле - **TEdit**  ;
- Метка - **TLabel** 

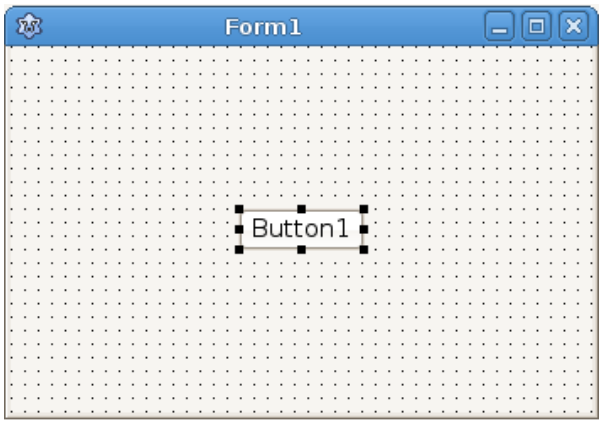
Данные объекты расположены на вкладке *Standart* Палитры компонентов.

Все объекты характеризуются:

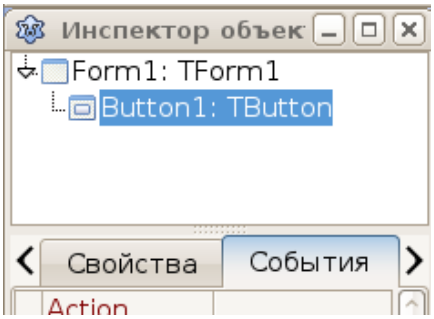
- свойствами (цвет, положение на экране и пр.),
- методами (действия или задачи, которые выполняет объект) и
- событиями (на какое событие должен реагировать объект).

4. События и процедуры обработки событий

Большинство компонентов могут реагировать на определенные действия пользователя - *события*, Наиболее распространенными событиями являются: щелчок кнопкой мыши на кнопке, ввод данных в поле ввода, наведение курсора мыши на компонент и т.д. При этом именно разработчик программы решает, как компонент будет реагировать на возникшее событие. Реакция программы на определенное событие реализуется с помощью *процедур обработки событий*. Рассмотрим пример процедуры обработки события нажатия кнопки TButton левой кнопкой мыши – OnClick. Для этого поместим компонет TButton на форму. Это можно сделать следующим образом: щелкаем на компонент TButton на Палитре компонентов и затем щелкаем в том месте формы где необходимо разместить кнопку. Кнопка появится на форме:



а в списке объектов окна Инспектора объектов появится запись Button1: TButton:



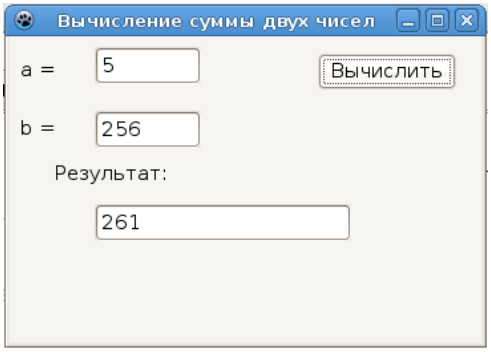
Выполнив двойной щелчок на кнопке Button1, расположенной на форме мы вызовем обработчик события, который создаст «скелет» процедуры обработки события OnClick нажатия кнопки в окне Редактора кода:

```
{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);
begin

end;

initialization
  {$I unit1.lrs}

end.
```

Рассмотрим структуру процедуры обработки события. Первая строка включает в себя служебное слово **procedure**, которое означает начало процедуры, затем идет имя процедуры, которое состоит из двух частей: Тип объекта-родителя (в нашем слусчае это тип TForm1, так как объектом-родителем является форма Form1) и, через точку, имя самого события. В скобках указан объект-инициатор события. Затем идут операторные скобки **begin … end**, внутри этих скобок и будет содержаться код, описывающий действия, которые должны происходить при возникновении события нажатия кнопки.

Практические задания.

Задание №1. *Создать простейший пользовательский интерфейс, для программы сложения двух чисел (a + b = c), содержащий объекты Button, TextBox - для ввода значений переменных a и b и вывода значения c, Label - для поясняющих надписей.*



Выполнение задания

Выполнение задания начнем с размещения на форме необходимых компонентов и задания их основных свойств:

1. Поместите на форму три метки TLabel и измените значение свойства Caption на «a =», «b =» и «Результат:» соответственно.
2. Поместите на форму три компонента TEdit и измените значение свойства Text на пустое значение (т.е. просто удалите текст «Edit1», «Edit2» и «Edit3»).
3. Поместите на форму кнопку TButton и измените значение ее свойства Caption на «Вычислить».
4. Измените значение свойства Caption формы Form1 на «Вычисление суммы двух чисел» (если вводимый текст не помещается в заголовке формы, то необходимо увеличить ширину формы, это можно сделать путем изменения значения свойства формы Width).

Теперь создадим процедуру обработки события (или обработчик события) OnClick нажатия кнопки «Вычислить». Для этого необходимо сделать двойной щелчок левой кнопкой мыши по кнопке «Вычислить» - при этом появится «скелет» процедуры Buuton1Click.

Далее создадим само «тело» процедуры. Прежде всего необходимо задать переменные, которые будут использоваться в процедуре. Для этого необходимо после строки

procedure TForm1.Button1Click(Sender: TObject);

написать следующую строчку:

var a, b, c: integer;

таким образом мы определяем три переменные, в которых будут хранится, соответственно, первое слагаемое, второе слагаемое и результат.

Далее, в операторных скобках необходимо написать следующие строки кода:

a := StrToInt(Edit1.Text);

b := StrToInt(Edit2.Text);

c:= a+b;

Edit3.Text := IntToStr(c);

Рассмотри подробнее введенные строки кода. Первые две строчки кода предназначены для присвоения введенных в поля Edit1 и Edit2 первого и второго слагаемых соответствующим им переменным a и b. При этом используется функция StrToInt (s: String):Integer, которая выполняет перевод введенного значения из строкового типа (свойство Text компонента TEdit имеет строковый тип данных) в числовой (т.к. переменные a и b числового типа).

Далее происходит операция сложения двух чисел (третья строка кода), а затем вывод результата в поле Edit3, причем здесь происходит обратное действие – мы переводим числовой тип данных в строковый.

Таким образом, полный код процедуры Button1Click имеет вид:

```
procedure TForm1.Button1Click(Sender: TObject);
var a, b, c: integer;
begin
    a := StrToInt(Edit1.Text);
    b := StrToInt(Edit2.Text);
    c := a + b;
    Edit3.Text := IntToStr(c);
end;
```

Теперь можно запустить проект на компиляцию и сборку (пункт Главного меню Запуск, подпункт Запуск или нажмите кнопку  на панели инструментов или нажмите кнопку F9, в результате чего мы получим работающую программу. Попробуйте ввести в поля a и b два числа и нажать кнопку «Вычислить», в поле «Результат» должна появиться сумма введенных чисел.

Задание №2. *Изменить название кнопки «Вычислить» на «Сложение».*

Задание №3. *Добавить к существующей программе еще три кнопки TButton, которые выполняют соответственно операции вычитания, умножения и деления (в данном случае необходимо учесть, что вместо типа integer необходимо использовать тип real, так как результатом операции деления не всегда является целое число).*