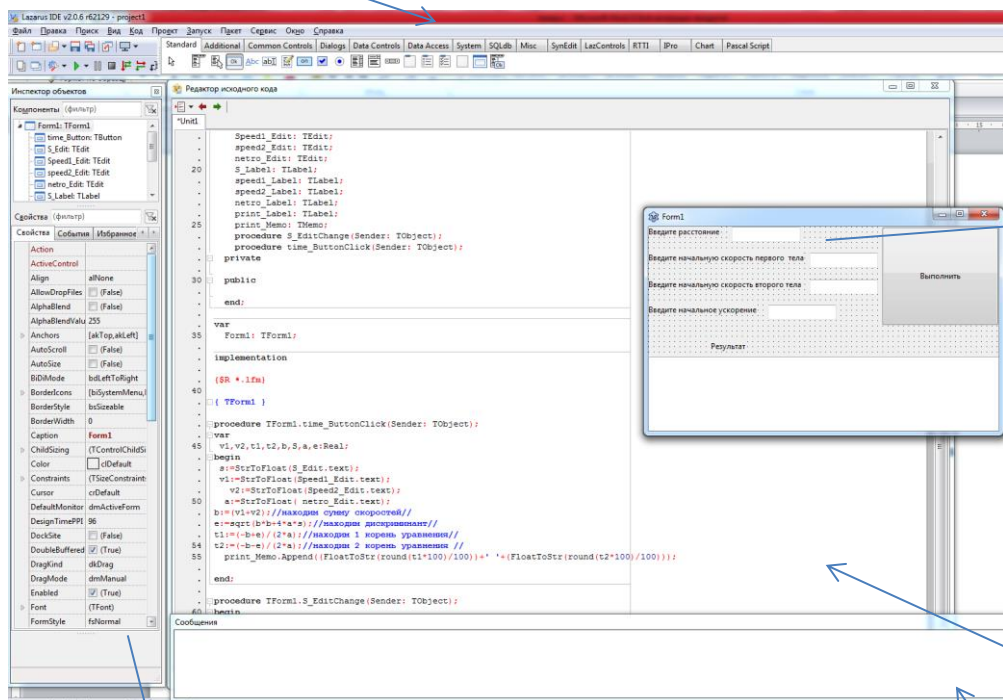
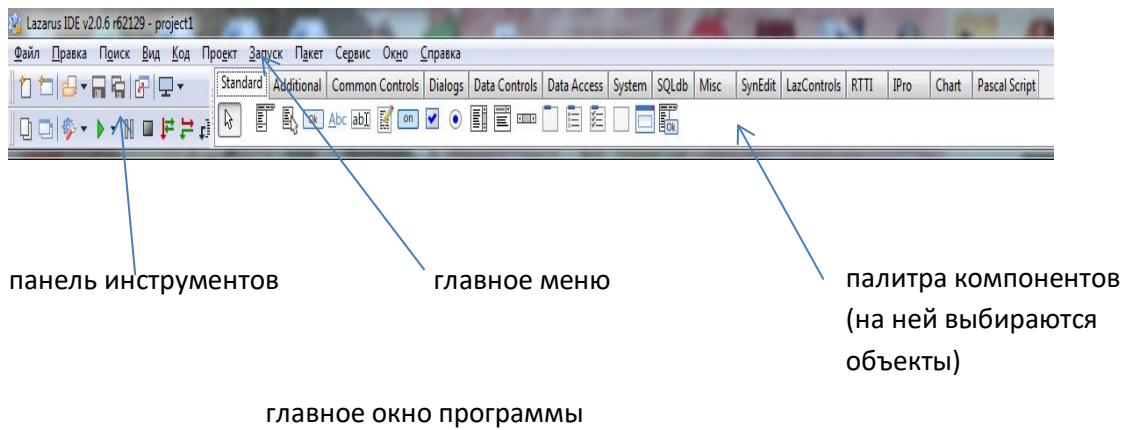
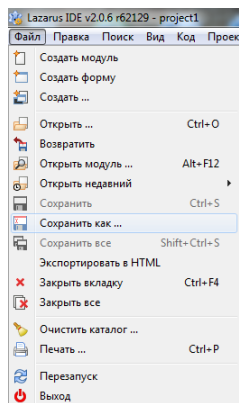


1

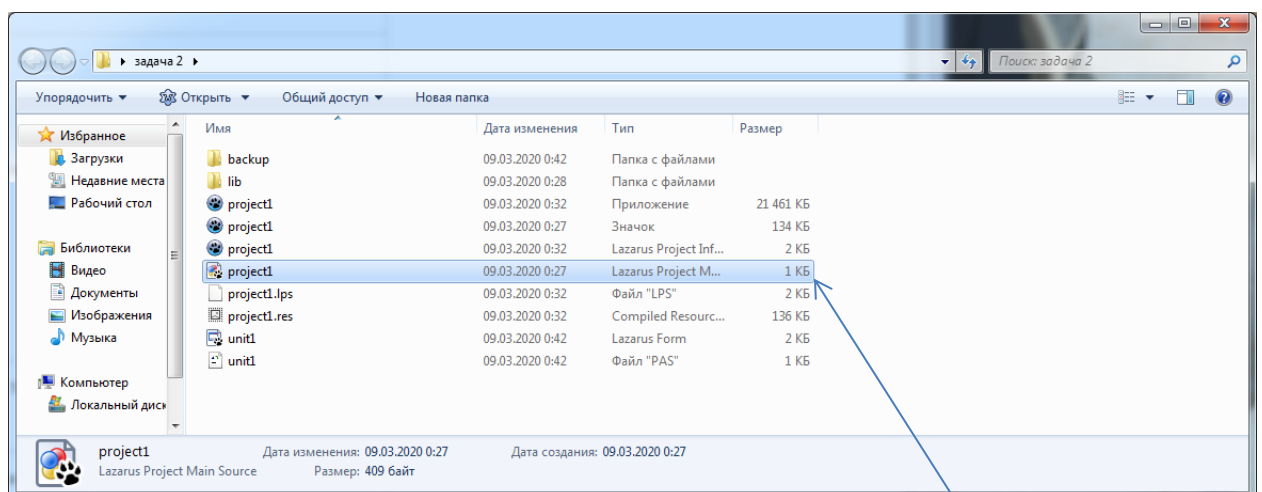
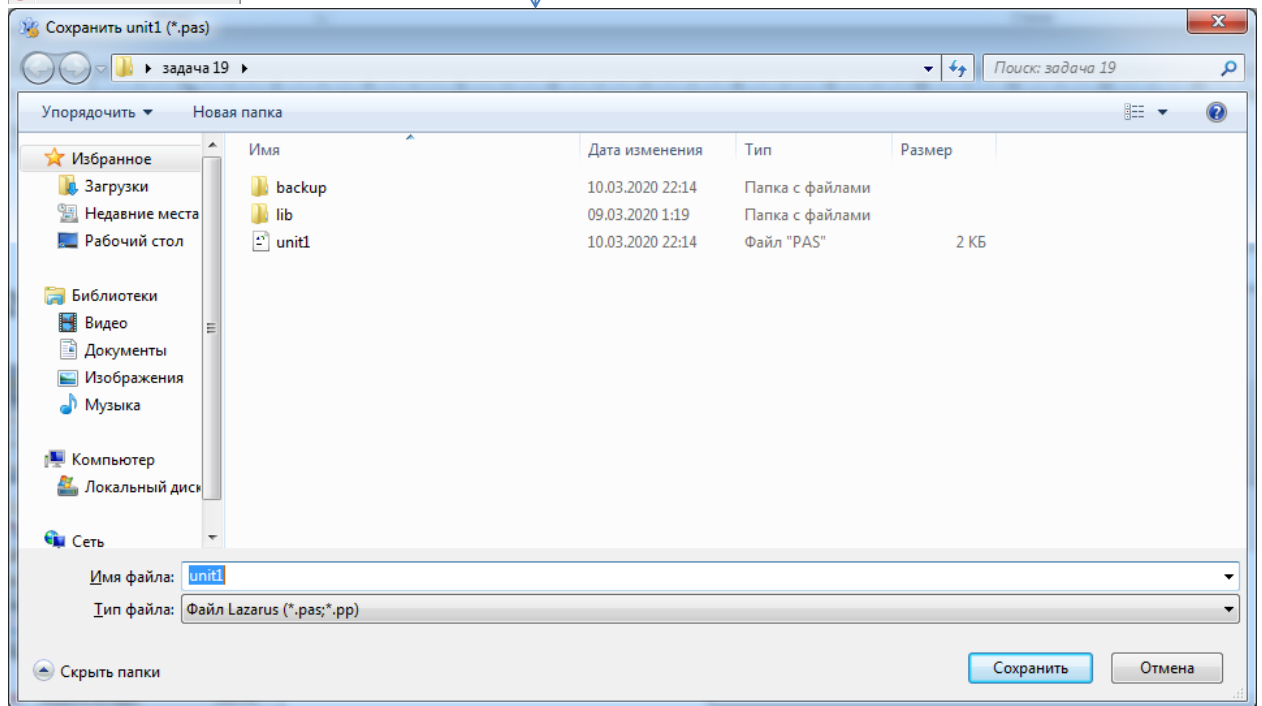


2

сообщения компилятора



чтобы сохранить проект , нужно
заранее создать папку, в которую
будем сохранять



данный ярлык откроет все окна лазаруса

Структура программ Lazarus

Любая программа в Lazarus состоит из файла проекта (файл с расширением `dpr`) и одного или нескольких модулей (файлы с расширениями `pas`). Каждый из таких файлов описывает программную единицу Lazarus.

Структура проекта

Файл проекта представляет собой программу, написанную на языке Lazarus и предназначенную для обработки компилятором. Эта программа автоматически создается Lazarus и содержит лишь несколько строк. Чтобы увидеть их, запустите Lazarus и выберите команду **Проект ► Просмотр исходника** главного меню. Lazarus покажет окно кода с вкладкой `Project1`, содержащее такой текст:

```
program Project1;

{$mode objfpc}{$H+}

uses

{$IFDEF UNIX}{$IFDEF UseCThreads}
cthreads,
{$ENDIF}{$ENDIF}

Interfaces, // this includes the LCL widgetset

Forms

{ add your units here }, Unit1;

begin

Application.Initialize;

Application.CreateForm(TForm1, Form1);

Application.Run;

end.
```

В окне кода полужирным шрифтом выделяются так называемые зарезервированные слова, синим шрифтом комментарии, а красным шрифтом директивы компилятора. Как видим, текст программы начинается зарезервированным словом **program** и заканчивается словом **end** с точкой за ним. Сочетание слова **end** со следующей за ним точкой называется *терминатором программной единицы*: как только в тексте программы встретится такой терминатор, компилятор прекращает анализ программы и игнорирует оставшуюся часть текста.

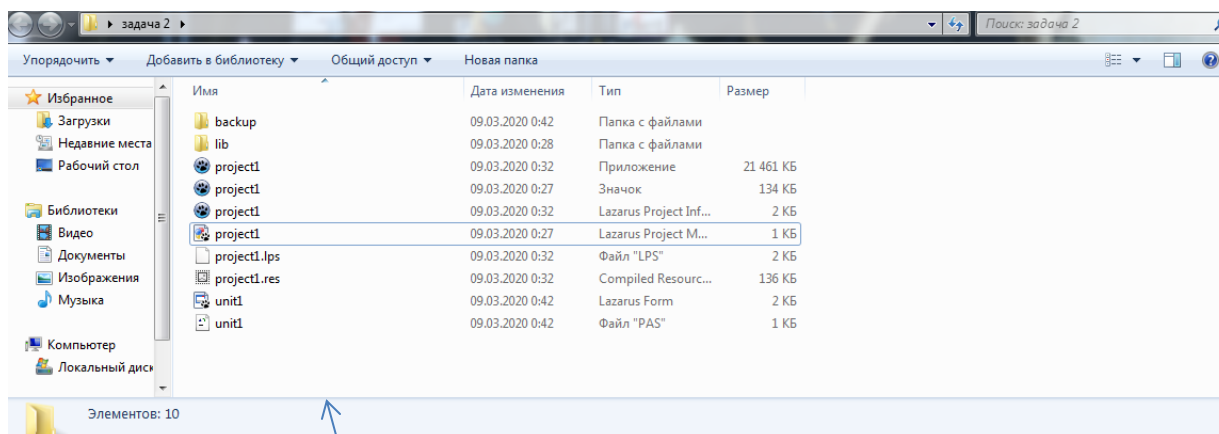
Структура проекта

Любой проект в Lazarus – это совокупность файлов, из которых создаётся единый исполняемый файл.

1. файл описания проекта (.lpi)
2. файл проекта (.lpr);
3. файл ресурсов (.lrs);
4. модуль формы (.lfm);
5. программный модуль (.pas);

КОМПИЛЯЦИЯ

+ Исполняемый файл с расширением .exe



файлы после сохранения

4 Компонент TLabel (Надпись)

Компонент Label (Надпись) используется для вывода на форму текста, который пользователь не может изменить во время выполнения программы

Caption	Отображаемый в поле надписи текст.
Color	Цвет фона области вывода текста.

Font	<i>Параметры шрифта, используемые для отображения текста: Font.Name — вид шрифта; Font.Size – размер шрифта; Font.Color – цвет шрифта.</i>
------	--

Компонент Button (Кнопка) – командная кнопка, с помощью которой пользователь может вызывать выполнение какого-либо действия.

Name	<i>Имя компонента. Используется в программе для доступа к компоненту и его свойствам.</i>
Caption	<i>Текст на кнопке.</i>

Компонент TEdit (Поле ввода) – текстовое поле, которое в отличие от TLabel можно редактировать во время выполнения приложения. Текст, вводимый в поле ввода, хранится в свойстве Text.

Компонент TEdit (Поле ввода) – текстовое поле, которое в отличие от TLabel можно редактировать во время выполнения приложения. Текст, вводимый в поле ввода, хранится в свойстве Text.

Name	<i>Имя компонента. Используется в программе для доступа к компоненту и его свойствам.</i>
------	---

Основное предназначение компонента TМето — работа с большим количеством строк (ввод, отображение и редактирование текстового материала).

Text	Текст, находящийся в поле Мето. Рассматривается как единое целое.
Font	Шрифт, используемый для отображения вводимого текста.

5 На форме нажать двойным кликом на кнопку и автоматически вызовется процедура обработчика