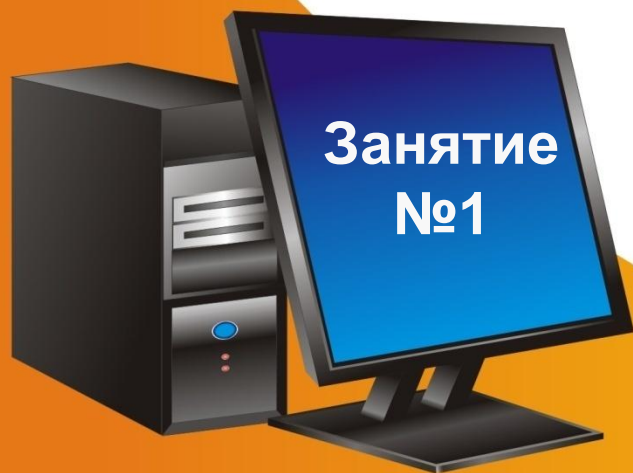


Дистанционное обучение основам программирования в среде Delphi



Как мы будем работать?

Как мы будем работать?

Интерактивные уроки

Изучение нового материала

Обзор разработанных приложений

Образовательный портал

Выполнение практических работ

Самостоятельная работа

Защита проектов

Демонстрация своего проекта

Рекомендуемая литература:

САМОУЧИТЕЛЬ «Delphi 7: Программирование на Object Pascal», Н. Культин, «БХВ – Санкт-Петербург», 2003

Учебный курс «Delphi 6: Программирование», В. Фаронов, «Питер», 2002

Программное обеспечение:

Для создания интеллект-карт: **Visual Mind**
Free Mind

Для решения задач: **Borland Delphi 7**

Взять в ГУ "РЦ-ШДПО"

Системы программирования для персонального компьютера



Системы программирования для персонального компьютера

Системой программирования называется - комплекс программ, предназначенный для автоматизации программирования задач на компьютере.

Системы программирования

```
graph TD; A[Системы программирования] --- B[Машинно-ориентированные]; A --- C[Машинно-независимые];
```

Машинно-ориентированные

Машинно-независимые

Системы программирования для персонального компьютера

Машинно-ориентированные системы

программирования имеют входной язык, наборы операторов и изобразительные средства которых зависят от особенностей компьютера.

Машинно-независимые системы программирования – это системы описания и реализации алгоритмов решения задач и информации, подлежащей обработке, которые не требуют знания особенностей организации и функционирования компьютера.

Системы программирования для персонального компьютера

Включают в себя:

Компилятор – транслирует весь текст программы, написанной на языке высокого уровня, в ходе непрерывного процесса.

Интерпретатор – последовательно анализирует по одному оператору программы, превращая при этом каждую синтаксическую конструкцию, записанную на языке высокого уровня, в машинные коды и выполняя их одна за другой.

Компиляция

Чтобы процессор смог выполнить работу в соответствии с инструкциями исходной программы, исходная программа должна быть переведена на **машинный язык** – язык команд процессора. Эту задачу выполняет специальная программа – **компилятор**.

Компилятор выполняет последовательно две задачи:

1. Проверяет текст программы на наличие синтаксических ошибок.
2. В случае отсутствия синтаксических ошибок создает (генерирует) выполняемую программу в машинных кодах

Убедиться в правильности работы программы можно только во время ее тестирования – пробных запусков программы и анализа полученных результатов.

Информационные технологии в развитии операционных систем (ОС) и сервисных программ



Развитие ОС и программного обеспечения

В развитии ОС и всего программного обеспечения различают три подхода, которые в дальнейшем определяют все недостатки и достоинства разрабатываемых программ:

- 1) Процедурный подход
- 2) Объектно - ориентированный подход
- 3) Модульный подход

Процедурный подход

Цикл работы программы представляет собой последовательность заранее установленных процедур

Недостаток

Данный подход приводит к разобщенности в программе данных и методов их разработки, что накладывает ограничения на возможности создания больших систем и решения больших задач

Примеры:

К процедурным относятся следующие языки программирования: BASIC, Pascal, C и Assembler

Объектно-ориентированный подход (ООП)

Под **объектом** понимается совокупность свойств (структур данных), методов их обработки (подпрограмм изменения свойств) и событий, на которые данный объект может реагировать.

Системы программирования, относящиеся к объектно - ориентированным: **Visual Basic, Delphi, C++ , Visual C++**.

В контексте развития ОС Windows ООП реализован в разработке пользовательского графического интерфейса, многозадачности, задач оптимального управления памятью.

Модульный подход

Модуль – это самостоятельная часть программы, имеющая определенное назначение и обеспечивающая заданные функции обработки автономно от других программных модулей.

Программный продукт представляет собой совокупность взаимосвязанных частей – программных модулей.

Borland Delphi



Назначение и возможности Delphi

Delphi - система программирования для разработки приложений, работающих в среде Windows, которая относится к классу инструментальных средств ускоренной разработки за счет визуального конструирования форм и широкого использования библиотеки визуальных компонентов.

При разработке интерфейса программы программист использует специальное окно, которое называется окном **формы** и наполняет его **компонентами**, реализующими нужные интерфейсные свойства (различные списки, кнопки, полосы прокрутки и т. п.).

Понятие проекта

Проект – это набор файлов, используя которые компилятор создает исполняемый файл программы (**exe-файл**).

В простейшем случае проект будет состоять из:

- файла описания проекта (**dpr-файл**),
- файла главного модуля (**pas-файл**),
- файла ресурсов (**res-файл**),
- файла описания формы (**dfm-файл**),
- файла описания функций (**cpp-файл**).

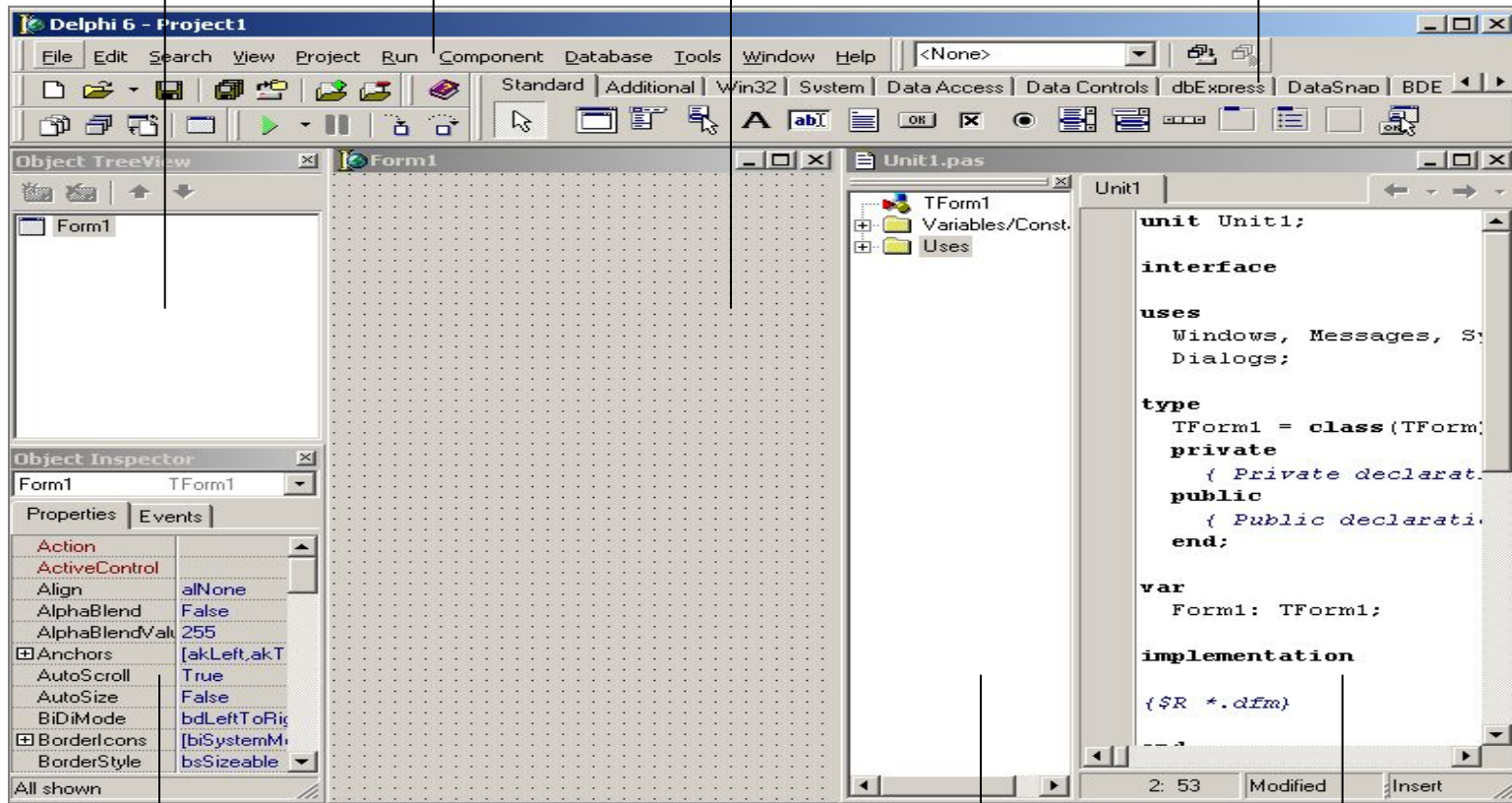
Для корректной работы будущего приложения важно, чтобы все эти файлы находились в одной папке. Поэтому прежде чем приступить к разработке программы, необходимо создать папку с названием будущего приложения.

Основные окна Delphi

Окно дерева объектов
и меню

Окно формы

Палитра компонентов



Окно инспектора объектов

Окно браузера

Окно кода программы

Основные понятия Delphi

Метод в Delphi - действие, выполняемое над объектом.

Событие в Delphi - действие, распознаваемое объектом, для которого можно запрограммировать отклик.

Свойство в Delphi - атрибут объекта, определяющий характеристики объекта.

Панель элементов Delphi - содержит компоненты, из которых можно построить интерфейс приложения.

Форма в Delphi - будущее окно приложения, в котором размещаются различные элементы управления.

Компоненты вкладки Standard



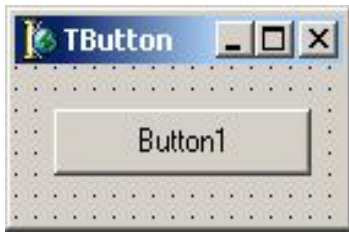
TLabel – метка



Компоненты класса **TLabel** предназначены для размещения на форме различных текстовых надписей. Для этого служит центральное свойство компонента – **Caption**. С помощью свойства **Font** можно разнообразить вид надписи.

Свойство	Описание
AutoSize:Boolean	Указывает, будет ли метка изменять свои размеры в зависимости от помещенного в нее текста
WordWrap:Boolean	Разрешает/запрещает разрыв строки на границе слова
Transparent:Boolean	Определяет прозрачность метки.

TButton – кнопка



Компоненты **TButton** широко используются для управления программами. Связанный с кнопкой алгоритм управления реализуется в обработчике события **OnClick**. В отличие от большинства других видимых компонентов кнопка является элементом операционной системы Windows и поэтому свойства этого компонента зависят от текущих настроек самой системы.

Свойство	Описание
Cancel: Boolean	Если значение True, событие OnClick кнопки возникает при нажатии клавиши Esc
Default: Boolean	Если значение True, событие OnClick кнопки возникает при нажатии клавиши Enter