Содержание

Глава 1.Основные отличия различных версий DELPHI

1.1 Первая версия языка Delphi

1.1.2 Вторая версия языка Delphi

1.1.3 Третья версия языка Delphi

1.1.5 Пятая версия языка Delphi

1.1.6 Шестая версия языка Delphi

Глава 2. Организация работы в среде визуального программирования Delphi

2.1 Главное окно

2.2 Окно формы Настройка окон

2.3 Модификация формы её свойств и событий

**Введение**

Delphi – это греческий горд, где жил дельфийский оракул. И этим именем был назван новый программный продукт с феноменальными характеристиками.

Delphi – это комбинация нескольких важнейших технологий:

* Высокопроизводительный компилятор в машинный код
* Объектно – ориентированная модель компонент
* Визуальное (а, следовательно, и скоростное) построение приложений из программных прототипов
* Масштабируемые средства для построения баз данных

Компилятор, встроенный в Delphi, обеспечивает высокую производительность, необходимую для для построения приложений в архитектуре «клиент - сервер». Этот компилятор является самым быстрым в на стоящее время в мире, его скорость компиляции составляет свыше ста двадцати тысяч строк в минуту на компьютере 486 DX33. Он предлагает легкость разработки и быстрое время проверки готового программного блока, характерного для языков четвертого поколения (4GL) и в тоже время обеспечивает качество кода, характерного для компилятора 3GL. Кроме того, Delphi обеспечивает быструю разработку без необходимости писать вставки на Си или ручного написания кода (хотя это возможно).

В процессе построения приложения разработчик выбирает их палитры компонент, делающие крупные мазки кистью. Еще до компиляции он видит результаты свой работы – после подключения к источнику данных их можно видеть отображенными на форме, можно перемещаться по данным, представлять их в том или ином виде. В этом смысле проектирование в Delphi мало чем отличается от проектирования в проектирующей среде, однако после выполнения компиляции мы получим код , который исполняется в десять – двадцать раз быстрее, чем то же самое сделанное при помощи интерпретатора. В Delphi компиляция производится непосредственно в родной машинный код, в то время как существуют компиляторы, превращающие программу в так называемый р – код, который затем интерпретируется виртуальной р – машиной. Это не может не сказаться на фактическом быстродействии готового приложения.

# Глава 1. Основные отличия различных версий DELPHI

# 

# 1.1 Первая версия языка Delphi

Первая версия появилась в мае 1995 года, когда еще не существовала Windows 95. Эта единственная версия, работающая от управления 16-разрядной Windows 3.1. В ней впервые была опробована новая модель объектов, заимствованная из различных объектно-ориентированных языков, и главным образом из языка С++. Эта модель оказалась настолько революционной, что существовавшая в то время в поздних Turbo Pascal объекты стали не нужные (их возможности полностью проглотила новая модель), а сама новая модель получила название класса.

Класса активно используют динамическую память в связи, с чем несколько изменилась нотация языка, а сам язык был назван Object Pascal. По сравнению с Turbo Pascal в него были внесены, существенны дополнения и изменения, в том числе здесь и далее перечисляются лишь наиболее важные, с моей точки зрения, изменения:

1. Введены открытые массивы и конструкторы для передачи массивов переменной длины;
2. Введена внутренняя для функций переменная RESOLT и разрешено игнорировать разрешения функции результат;
3. Сняты ограничения на тип возвращаемого функции результатов (этот тип может быть любым, за исключение объектов старого типа);
4. Введен механизм обработки исключительности ситуации.

Библиотека компонентов первой версии самого начала показала основную направленность вновь разработанной системы: программирование баз данных. С этой целью первая из все последующую Delphi снабжаются специальным инструментом доступа к данным BDE (Borland Database Engine – машина баз данных корпорации Borland), а также сервером баз данных Inter Base (этот сервер производится филиалом Borland – компании Inter Base Soft were Corporation) и соответствующими средствами конфигурации сервера, его контроля и связи с ним.

Палитра компонента состоит из 9 страниц и содержит 79 компонентов. В качестве дополнительных утилит поставлялись система генерации отчетов ReportSmiteh v.2.5. и входящая в её состав автономно запускаемая утилита ReportSmiteh Run Timer Viewer, утилита доступа к данным Database Desktop v.5.1. и утилита конфигурации BDE – Database Engine Configuration v.2.5.

Кроме того, утилита WinSighte и Win Specter предназначались для упрощения отладки разрабатываемых программ среди 16-разрядных OC Windows3.x.

# 1.2 Вторая версия языка Delphi

Вторая и все последующие версии Delphi предназначены для работы под управлением 32-разрядных ОС Windows 95/98/200/NT (Windows 32). В неё были внесены многочисленные изменения, связанные с переходом на качественно новую ОС, в том числе:

* введена поддержка 16-битных символов и составленных из них строк;
* введен новый формат строк «обычных» 8-битных символов произвольной длины;
* введены новые типы данных –Variant и Currency;
* введен механизм фильтрации в табличные наборы данных TTable.

Начиная с версии 2, Delphi поставляется в нескольких комплектациях, отличающихся набором инструментальных средств и компонентов. Это соответствует более гибкой ценовой политике: минимальная по возможностям комплектация состоит 200-300 долларов, в то время как цена наиболее мощной комплектации может достигать нескольких тысяч. Во 2-й версии таких комплектов было 3: Desktop, Professional и Client/Server Suite.

Среда обработки обогатилась Репозиторием Объектов и множеством полезных программ-экспертов. К утилитам добавлена весьма эффективная программа SQL Explorer v.1/02, существенно облегчающая операции с базами данных, и программа переноса данных Data Migration Expert, упрощающая переход к архитектуре клиент-сервер. Утилиты ReportSmit и Data-base Desktop представлены более совершенными версиями 3.0. и 7.1. соответственно. Утилита Database Engine Configuration предыдущей версии переименована в BDE Configuration v.3.0. Введена утилита версии переименована в BDE Configuration v. 3.0. Введена утилита Image Editor v. 2.0,служащая для создания различного рода изображений (произвольных рисунков, пиктограмм и курсоров), которые можно включать в файл ресурсов и использовать в работающей программе, и утилита SQL Monitor v. 1.0,позволяющая просматривать SQL- запросы в процессе выполнения программы.

На двенадцати страницах галереи компонентов расположены 114 стандартных компонентов. В том числе на странице Report размещены удобные компоненты для создания отчетов по хранящимся в базах данным. Эти компоненты оказались значительно эффективнее специальной утилиты Report Smith, в связи с чем эта утилита не входит в поставку версии 3 и выше.

# 1.3 Третья версия языка Delphi

Основные новшества этой версии:

* использование механизма пакетов для облегчения распространения и повторного использования комплектов;
* улучшенные свойства редактора кода: построение фрагментов кода по образцу; оперативная подсказка о типе и количестве формальных параметров при обращениях к подпрограмме; оперативный просмотр в режиме откладки содержимого полей, свойств и переменных с помощью указателя мыши;
* поддержка шаблонов компонентов;

поддержка технологий COM, ActiveX, OLEntprice и (частично) COBRA.

Третья версия поставлялась в 4 комплектациях: Standart, Professional, Clint/Server Suit и Enterprise.

Состав утилит комплектации Client/Server Suite не изменялся по сравнению с предыдущей версией, но изменились названия некоторых из них: BDE Configuration в этой и последующих версиях называется BDE Administrator, a Data Migration Expert – Data Migration Wizard.

На 13 страницах галереи компонентов размещены 148 стандартных компонента. В модификации 3.5 введены дополнительные компоненты для реализации многозвенных баз данных.

# 1.4 Пятая версия языка Delphi

В июле 1999 г. вышла пятая по счету версия Delphi, основная особенность которой – попытка заменить громоздкий и не всегда быстрый механизм доступа к данным BDE, который традиционно использовался во всех предыдущих версиях, альтернативными механизмами.

Для этого, во – первых, в Delphi 5 включена поддержка технологии ADO (ActiveX Data Object – объекты данных, построены как объекты ActiveX ), которая усилено развивается корпорацией Microsoft.

Во – вторых, сотрудники Borland и его подразделения InterBase Software Corporation разработали серию компонентов облегченного доступа к данным, хранящимся в таблицах сервера InterBas v.5.5. и выше (страница InterBase палитры компонентов). Эти компоненты также не требуют BDE и, таким образом, создают «облегченное» клиентское место.

Менее значительные изменения, внесенные в версию 5:

1. включен эксперт создания и настройки произвольных модулей с расширенными возможностями представления взаимосвязи данных;
2. улучшена технология MIDAS: для более гибкой работы с Microsoft Transaction Server введен повторно – входимый (stateless), брокер данных; упрощен процесс разработки интернет – приложений за счет компонентов новой страницы Internet Express;
3. значительные изменения внесены в интегрированную среду разработчика ИСР в том числе:

* для улучшения координации коллективной работы над одним проектом введен новый инструмент – список To-Do;
* программист теперь может при желании использовать несколько вариантов настройки основных окон Delphi – например, для режима кодирования на экране могут не присутствовать отладочные окна, которые, наоборот, могут понадобиться в отладочном режиме; нужный вариант настойки легко выбирает новыми интерфейсными элементами в главном окне Delphi;
* введены фильтрующие свойства в окне Инспектора Объектов, упрощающий выбор нужного свойства;
* опции Инспектора Объектов теперь могут снабжаться небольшими пиктограммами, облегчающими правильный выбор нужной опции (например, радом с названием каждого цвета показывается небольшой прямоугольник, заполненный этим цветом, рядом с названием курсора – его вид);
* существенно усилены возможности встроенного отладчика: точки отладочного останова можно группировать и сделать доступной или недоступной сразу группу точек; с каждой точкой останов можно связать одно или несколько действий, которые будут выполняться при достижении этой точки; с помощью команды Run Attach to Process можно отлаживать процесс, запущенный в другом экземпляре ИСР (эта возможность существенно упрощает отладку многозвенных приложений); с помощью выбора Run 1 Run Until Return в главном меню можно продолжить пошаговую отладку после завершения текущей подпрограммы.
* введены дополнительные возможности менеджер проекта, упрощающий координацию многих разработчиков в рамках единого проекта;
* создан механизм менеджера трансляций, облегчающий разработку многоязычных программ в рамках единого проекта;
* внесены изменения в кодовый редактор, позволяющий настраивать используемые в нем «горячие» клавиши;
* несколько переработанная встроенная справочная служба;

4. внесены многочисленные изменения и дополнения в галерею компонентов (в новой версии наиболее мощной комплектации Enterprise 20 страница расположены 218 стандартных компонентов).

Версия поставляется в 3 комплектах: Standard, Professional и Enterprise.

Состав утилит, входящих в комплектацию Enterprise, полностью соответствуют утилитам комплектации Client/Server Suite предыдущей версии.

# 1.5 Шестая версия языка Delphi

Версия 6 вышла в мае 2001 г. почти через 2 года после выхода 5 версии. Эту задержку можно объяснить тем, что параллельно с работой над новой версией Delphi шла разработка варианта Delphi для OC Linux – это система программирования вышла в феврале 2001г. и получила название Kylix.

В результате 6 версия Delphi имеет уникальную способность: она способна создавать так называемые межплатформенные приложения, программы, которые одинаково успешно могут работать как под управлением Windows 32, так и под Linux.

В 1991 г. финский программист Линус Торвальдс решил создать полностью открытую UNIX – подобную OC Linux. Полная открытость системы означает, что её исходный код всегда доступен любому желающему.

для реализации своего замысла ученый привлек широкий круг программистов, опубликовал предварительный исходный код Linux в Интернете и пригласив всех желающих дополнять его своими разработками, но с обязательным условием: публикации исходного кода этих разработок. Таким образом, эта ОС изначально разрабатывалась (и продолжает разрабатываться!) как полностью открытая и бесплатная система, которую может получить любой желающий. Сегодня по этой системой пользуются около 8 миллионов человек решились отказаться от услуг Microsoft, вечной политикой которой была полня закрытость исходных текстов своих ОС, в пользу Linux, обеспечивающей, по утверждениям специалистов, не меньшую надежность, чем Windows NT,при приблизительно одинаковых возможностях.

Таким образом, основной идеей Delphi 6 является обеспечение перехода от дорогих патентованных решений корпорации Microsoftк бесплатным (или почти бесплатным) решениям на базе Linux.

На 27 страницах палитры компонентов размещены 387 компонентов (в режиме CLX – 211 компонентов на 21 странице).

**Глава 2. Организация работы в среде визуального программирования Delphi**

# 2.1 Знакомство со средой Delphi

Среда Delphi – это сложный механизм, обеспечивающий высокоэффективную работу программиста. Визуально она реализуется несколькими одновременно открытыми окнами на экране. Окна могут перемещаться по экрану, частично или полностью перекрывая друг друга, что обычно вызывает у пользователя, привыкшего к относительной «строгости» среды текстового процессора Word или табличного процессора Excel, ощущение пройдет, и вы научитесь быстро отыскивать нужное окно, чтобы изменить те или иные функциональные свойства создаваемой вами программы, ибо каждое окно несет в себе некоторую функциональность предназначенную для решения задач. При запуске программы вы увидите шесть окон: главное окно, окно дерево объектов, окно инспекторов объекта, окно браузера, окно формы, окно кода программы.

Чтобы упорядочить окна вам придется в ручную изменить их положение и размеры, обычно окно кода программы перекрыто окном формы. Чтобы найти окно кода нужно нажать клавишу F12, повторное нажатие F12вновь активизирует окно формы – эта клавиша работает как двоичный переключатель, попеременно показывая окно формы или окно кода программы. При запуске Delphi поверх всех окон появится окно, с помощью этого окна вы сможете получить доступ к Web – страницам корпорации Inprise для просмотра самой свежей информации о корпорации и ее программных продуктах.

Главное окно осуществляет основные функции управления проектом создаваемой программы. Это окно всегда присутствует на экране и занимает его самую верхнюю часть, даже в максимальном состоянии его размеры и положения не отличаются от обычных. Оно содержит все необходимые средства для управления проектом, все опции главного меню представляют собой заголовки, открывающие доступ к выпадающим меню второго уровня.

Все компоненты главного окна располагаются на специальных панельках, в левой части которых имеются кнопки управления, позволяющие с помощь мыши перетаскивать панельки с помещенными на них элементами. Любую панельку кроме главного меню можно убрать из окна сделать ее невидимой или «пустить плавать» по экрану в отдельном окне. Для этого нужно лишь «стащить» панельку с помощью мыши за кнопку из пределов главного окна. Для изменения состава показываемых на панели кнопок нужно предварительно щелкнуть по ней правой кнопкой мыши. В появившемся после этого окне вспомогательного меню перечислены названия всех панелей и указан их статус (отмеченные флажками панельки видны в главном окне; ели отметку убрать панелька исчезнет). После выбора *Customize* (настройка) появится окно настройки. Теперь можно «стаскиваться» с панелек ненужные кнопки, выбирать из списка в окне *Commands* (закладка) нужные кнопки и перетаскивать их на экран.

*Пиктографические кнопки.*

Пиктографические кнопки открывают быстрый доступ к наиболее важным опциям главного меню. По функциональному признаку они разделены на семь групп. Каждая группа занимает отдельную панельку. В следующей таблице описаны команды, реализуемые стандартным набором пиктографических кнопок.

Начиная с версии 5, главное окно обогатилось тремя новыми интерфейсными элементами, указанными в группе Desktops предыдущей таблицы. С помощью этих инструментов программист может подготовить несколько вариантов расположения остальных окон Delphi и сохранить их в настроенном файле.

Обычно выбираются две или три основные конфигурации окон: для режима разработки форм, для кодирования и для отладки. При разработке формы на экране должна быть видна сама форма, Дерево объектов и Инспектор объектов. Настроив соответствующие размеры и положение этих окон, такую настройку можно сохранить под именем, например Design Desk. Для режима кодирования обычно нужно только максимально распахнутое на экран окно кода с браузером Code Explorer и пристыкованное к нему окно Дерева объектов. Настроив соответствующие размеры и положения этих окон, такую настройку можно сохранить под именем, например, DesignDesck. Для режима кодирования обычно нужно только максимально распахнутое на экране окно кода с браузером Code Explorer и пристыкованное к нему окно Дерева объектов. Такую конфигурацию можно сохранить под именем Code – Decks. Наконец, в отладочном режиме е окну кода желательно «прицепить» некоторые отладочные окна, как Watches (окно наблюдения за переменными / выражениями) и Breakpoints (окно создания / настройки точек контрольного останова). Такую конфигурацию можно сохранить под именем DebugDesk, например. Теперь при переходе к тому или иному режиму достаточно раскрыть список и выбрать из него нужную настойку. Если предварительно указать с помощью кнопки настойку для отладочного режима, эта настройка появиться автоматически, как только среда перейдет в режим отладки.

Палитра компонентов – это главное богатство программы. Она занимает правую часть главного окна и имеет закладки, обеспечивающее быстрый поиск нужного компонента. Под компонентом понимается некий функциональный элемент, содержащий определённые свойства и размещаемый программистом в окне формы. С помощью компонентов создаётся каркас программы, во всяком случае - её видимые на экране внешние проявления: окна, кнопки, списки выбора.

Как и панель компонентов может настаиваться.

Для этого используется специальный редактор, окно которого появляется на экране после щелчка правой кнопки мыши на любой пиктограмме в палитре компонентов и выбора опции Properties (свойства).

Нет смысла удалять какой – то из компонентов, а создавать новые пока еще не умеем. В стандартном наборе страниц палитры компонентов есть страница Samples, содержащая довольно часто используемые компоненты. В то же время ее закладка в палитре закрыта двумя небольшими кнопками «прокрутки» закладок палитры. Имеет смысл поменять ее местами с менее используемой страницей, например, со страницей ADO. Можно «перетащить» строку Samples станет доступна в любой момент. Ели к тому же использовать в своих проектах технологию ADO для доступа к базам данных, перетащите строку ADO в самый конец списка.

# 2.2 Структура и свойства окна Delphi

Окно формы представляет собой проект Windows – окна будущей программы. Вначале это окно пусто. Точнее, оно содержит стандартные для Windows интерфейсные элементы – кнопки вызова системного меню, максимизации, минимизации и закрытия окна, полосу заголовка и очерчивающую рамку. Вся рабочая область окна обычно заполнена точками координатной сетки, служащей для упорядочения размещаемых на форме компонентов (вы можете убрать эти точки, вызвав с помощью меню Tools Environment Options соответствующее окно настроек и убрав флажок в переключателе Display Grid на окне связанном с закладкой Preferences).

*Окно Дерева объектов.*

Это окно появилось в версии шесть и предназначено для наглядного отображения связей между отдельными компонентами, размещенными на активной форме или активном модуле данных. Щелчок по любому компоненту в этом окне активизирует соответствующий компонент в окне формы и отображает свойства этого компонента в окне Инспектора объектов. Двойной щелчок приводит к срабатыванию механизма Code Insight, который вставляет в окно кода заготовку для обработчика события On click. Наконец, компонент можно «перетащить» в окне и таким образом понять его владельца (свойство Parent). В предыдущих версиях такую замену можно было сделать только с помощью межпрограммного буфера обмена Clipboard.

*Окно инспектора объектов.*

Любой размещаемый на форме компонент характеризуется некоторым набором параметров: положением, размером, цветом. Часть этих параметров, например, положение и размеры компонента, программист может изменять, манипулируя с компонентом в окне формы. Для изменения других параметров предназначено окно Инспектора объектов. Это окно содержит две страницы – Properties (свойства) и Events (события). Страница Properties служит для установки нужных свойств компонента, страница Events позволяет определить реакцию компонента на то или иное событие. Совокупность свойств отображает видимую сторону компонента: положение относительно левого верхнего угла рабочей области формы, его размеры и цвет, шрифт и текст надписи на нем; совокупность событий – его поведенческую сторону: будит ли компонент реагировать на щелчок мыши или на нажатие клавиш, как он будет вести себя в момент появления на экране или в момент изменения размеров окна,

Каждая страница окна Инспектора объекта представляет собой двухколончатую таблицу, левая колонка которой содержит название свойства или события, а правая – конкретное значение свойства или имя подпрограммы, обрабатывающей соответствующее событие.

Строки таблицы выбираются щелчком мыши и могут отображать простые или сложные свойства. К простым относятся свойства, определяемые единственным значением – числом, строкой символов, значением True (истина) или False (ложь). Сложные свойства определяются совокупностью значений. Слева от имени таких свойств указывается значок «+», а щелчок мышью по этому символу приводит к раскрытию списка составляющих сложного свойства. Чтобы закрыть раскрытый список, нужно щелкнуть по значку «-» сложного свойства.

Щелчок мышью на правой колонке строки таблицы активизирует указанное в ней значение свойства, при этом в правом конце строки может появится одна из кнопок.

В верхней части окна Инспектора объектов располагается раскрывающийся список вех помещенных компонентов. Поскольку форма сама по себе является компонентом, ее имя также присутствует в том списке. В отличие от предыдущих версий раскрывающейся список содержит не только имена компонентов, и их классы.

В локальном меню окна, которое появляется после щелчка по нему правой кнопкой, после выбора Stay on top, окно Инспектора объектов будет «всплывать» над всеми другими окнами независимо от его активности. Такое состояние окна удобно при частом его использовании, например, при конструировании сложной формы, содержащей множество компонентов. Если выбрать в локальном меню опцию Arrange и затем by Category, все строки окна Инспектора объектов буду представлять собой раскрывающееся списки свойств, упорядоченные по категориям.

Любые категории можно сделать невидимыми. Для этого нужно в локальном меню выбрать View и затем в дополнительном меню убрать флажок слева от категории.

В отличие от предыдущих версий Delphi в версии шесть сложные свойства, ссылающиеся на объекты, выделяются в окне Инспектора свойства, ссылающиеся на объекты, выделяются в окне Инспектора объектов цветом и показывается на странице Events. Если такое свойство определено, слева от него появляется кнопка «+», позволяющая увидеть (и при необходимости изменить) свойства и обработчики событий связанного объекта.

*Окно кода программы.*

Окно кода предназначено для создания и редактирования текста программы. Этот текст составляется по специальным правилам и описывает алгоритм работы программы. Совокупность правил записи текста называется языком программирования. В системе Delphi используется язык программирования Object Pascal,который представляет собой расширенную и усовершенствованную версию широко распространенного языка Паскаль, впервые предложенного швейцарским ученым Н. Виртом еще в 1970 г. и усовершенствованного сотрудниками корпорации Borland (созданные ими языки назывались Turbo Pascal, Borland Pascal и Object Pascal). Несмотря на то, что визуальная среда Delphi берет на себя многие рутинные аспекты программирования, знание языка Object Pascal является не переменным условием для любого программиста, работающего в этой среде.

Первоначально окно кода содержит минимальный исходный текст, обеспечивающий нормальное функционирование пустой формы в качестве полноценного Windows – окна. В ходе работы над проектом программист вносит в него необходимые дополнения, чтобы придать программе нужную функциональность. Поскольку для создания даже простых программ вам понадобится создавать и изменять (редактировать) код программы, ниже описываются основные приемы работы с окном кода.

Сразу после открытия нового проекта в нем будут такие строки:

unit unitl;

interfece

uses

Windows, Messages, Sysutils, Classes, Graphics, Controls, Forms, Dialogs;

type

TForm 1 = class (TForm)

private

{ Private declarations }

public

{ public declarations }

end;

var

Form 1: TFoem1;

implementation

{SR \*. dfm}

end.

Эти строки Delphi автоматически вставляет в окно кода для ново формы. Как уже говорилось, окно кода определяет поведенческую сторону окна программы (т.е. окна, появляющегося после начала работы программы), а окно формы – его внешнее проявления. Оба окна тесно связаны между собой, причем Delphi «хозяйничает» в его верхней части, вставляя необходимые строки между unit unitl; и implementation. Пока не стоит изменять эту часть текста. В дальнейшем мы будем вставлять в окно текст программы между строками {SR \*. DFM}

И end. в нижней части окна.

Чтобы вставить в окно новую строку (строки), нужно сначала, с помощью клавиш курсора или щелкнув по окну мышью, поставить текстовый указатель (мигающую вертикальную черту) на нужное место, а затем с помощью клавиатуры ввести текст. Обычно текст кода программы располагается в нескольких строках. Для перехода на новую строку используйте клавишу Enter.

Если в процессе ввода вы ошиблись и тут же заметили свою ошибку, удалите ошибочный символ клавишей Backspace ( продолговатая серая клавиша в правом верхнем углу зона алфавитно - цифровых клавиш или расположенная там же клавиша с левой стороны). Эта клавиша удаляет символ слева от мигающего указателя, а клавиша Delete – справа от него. Если понадобится удалить сразу всю строку текста, поставить в любое место строки мигающий указатель, нажмите клавишу Ctrl и, не отпуская ее клавишу с латинской буквой Y. Такое совместное нажатие клавиш в дальнейшем будем обозначать символом «+»: Ctrl + Y. Чтобы отменить последнее изменение текста, нажмите Ctrl + Z или выберите пункт меню Edit Undo.

Текстовый редактор версии Delphi 4, 5, 6, имеет расширенные «интеллектуальные» возможности Code Insight, связанные с контролем текста по мере его ввода программистом. К сожалению, этот контроль отнимает у компьютера определенные ресурсы. Чтобы избавится от раздражающих пауз в момент набора кода, особенно заметных на ПК с памятью менее 32 Мб, оснащенных процессорами с тактовой частотой 100 и менее МГц, раскройте список опции Tools главного меню, щелкните по опции Editor Options (для версии 4 на опции Environvent Options) и на странице, связанной с закладкой Code Insight, уберите флажки в переключателях Code Completion, Code Parameters и Tooltip Symbol Insign.

Вместе с окном кода обычно активизируется также и окно браузера Code Explorer, облегчающее поиск нужных элементов в случае когда в окне набрано много строк кода.

Двойной щелчок мыши браузера приводит к позиционированию текстового указателя в окне кода на описание. При разработке несложных учебных программ можно закрыть можно окно браузера маленькой кнопкой в его верхнем правом углу. Восстановить закрытое окно можно после щелчка правой кнопкой мыши по окну кода и выбора опции View Explorer. Окно браузера располагается на вспомогательной панели и имеет управляющую полсу в верхней части рамки. «Схватив» ее мышью, вы можете перетащить окно на любое свободное место экрана или «причалить» его к окну Инспектора объектов.

В версии шесть на нижней кроме окна кода располагаются две закладки – Code и Diagram. щелчок по последней активизирует страницу диаграмм. Первоначально это страница пуста. Чтобы наполнить ее, нужно «перетащить» мышью два и более компонента из окна Инспекторов компонентных связей. Это окно пристыковано к верхней части окна браузера. Чтобы перетащить компоненты в поле диаграмм, нужно нажать и удерживать клавишу Shift, после чего щелкнуть по компонентам и, продолжая удерживать эту клавишу, переместить указатель мыши в поле диаграмм и щелкнуть по нему левой кнопкой.

Для сложных форм множеством компонентов можно подготовить несколько диаграмм. Дав каждой диаграмме уникальное имя можно в любой момент с помощью выпадающего списка выбрать нужную. Страницу диаграмм можно напечатать.

# 2.3 Модификация формы её свойств и событий

Программирование в Delphi строится на тесном взоимодействии двух процессов: процесса конструирования визуального проявления программы Windows – окна и процесса написания кода, придающего элементам этого окна и программе в целом необходимую функциональность. для написания кода используется окно кода, для конструирования программы – остальные окна Delphi, и прежде всего – окна формы.

Между содержимым окон формы и кода существует неразрывная связь, которая строго отслеживается Delphi. Это означает, что размещение на форме компонента приводит к автоматическому изменению кода программы и наоборот – удаление тех ли иных автоматически вставленных фрагментов. Помня об этом, программисты вначале конструируют форму, размещая на ней очередной компонент, а уже после этого переходят, если это необходимо, к написанию фрагмента кода, обеспечивающего требуемого поведение компонента в работающей программе.

*Пустая форма и ее модификация.*

Как уже говорилось, окно формы содержит проект Windows – окна программы. Важно помнить, что с самого начала работы над новой программой Delphi создает минимально необходимый код, обеспечивающий ее нормальное функционирование в Windows. Таким образом, простейшая программа готова сразу после выбора опции File New Application, и нам остается просто запустить программу. Однако до этого выполнить две важные вещи: создать собственный рабочий каталог (папку) и нужным образом настроить Delphi.

В процессе работы над проектами программ, описываемых в этой книге, вам понадобится создать множество форм и модулей. Полезно сохранять эти данные в виде дисковых файлов в отдельной папке. Более того, для каждой программы в этой папке имеет смысл создать свою вложенную папку. Тогда, чтобы освободить место на диске для серьезной программы, вам будет достаточно уничтожить основную папку, а чтобы передать ту или иную учебную программу своему коллеге – переписать на дискету содержимое соответствующей вложенной папки.

Создайте папку с именем, а в нем вложенную папку для вашей первой программы. Чтобы не утруждать себя придумыванием подходящих имен для «близких» учебных программ, советую использовать для папки имя PGM\_1 или что – нибудь подобное.

Вторе, что предстоит сделать, внести изменения в стандартную настройку среды Delphi. Это необходимо для того, чтобы среде автоматически сохраняла на диске последнюю версию создаваемой вами программы.

Выберите опцию меню Tools Environment Options и убедитесь, что в появившимся диалоговом окне активна страница Preferences. В левом верхнем углу этой страницы в группе Autosave Options есть переключатели Editor Files и Desktop (в других версия эти переключатели располагаются в правом верхнем углу). Активизация переключателей приведет к автоматическому сохранению текста окна кода программы и общего расположения

окон Delphi перед началом очередного прогона создаваемой программы, что избавит вас от возможных потерь в случае зависания программы. Советую активизировать эти переключатели, щелкнув по каждому мышью. Чтобы следить за ходом компиляции, активизируйте также переключатель Show Compiler Progress в группе Compiling and Running. Далее будит полезным сохранять в ходе программы комментарии – текстовые фрагменты, которые не влияют на программы, но делают ее текст более понятным. Если вы предпочитаете комментарии на русском языке, следует выбрать соответствующий шрифт для отображения кода программы. По умолчанию редактор Delphi использует шрифт Courier New, в котором может не быть символов кириллицы. В этом случае нужно выбрать опцию Tools Editor Options и на странице Display в строке Editor Font установите Courier New Cyr (для предыдущих версий страница Display доступна в окне Tools Environment Options).

Теперь все готово прогона вашей первой программы. Щелкните мышью по кнопке в главном окне или, что проще, нажать клавишу F9: именно таким способом подготовленная Delphi – программа последовательно подходит три главных этапа своего жизненного цикла – этапы компиляции, компоновки и исполнения. На этапе компиляции осуществляется преобразования подготовленного в окне кода программы на языке Object Pascal в последовательность машинных инструкций, на этапе компоновки к ней подключаются необходимые вспомогательные программы, а на этапе исполнения готовая программа загружается в оперативную память и ей передается исполнение.

При нажатии F9 появится диалоговое окно Save Unitl As, в котором Delphi попросит указать имя файла для модуля Unitl – pas и папку его размещения.

По умолчанию Delphi предлагает разместить файл модуля и проекта в системной папке BIN. Поскольку эта папка содержит жизненно важные для Delphi файлы, обязательно измените ее на вашу рабочую папку.

*Имена в Delphi.*

Дело в том, что Delphi принципиально не русских, ни французских, ни китайских, ни вообще каких то ни было имен, в которых используя символы, отличные от латинских букв, цифр и знака подчеркивания , причем имя не должно начинаться цифрой, но может начинается знаком подчеркивания. Так как перечне нет пробела, имена не могут так же состоять из нескольких слов.

Далее сокращения до не более восьми символов – это дань ограничениям Delphi версии 1. Эта версия рассчитана на работу под управлением Windows 3.x,которая не признает длинных имен файлов. Если в работе с Delphi 32 (с любой версией от второй до шестой), можно не сокращать имена и написать my \_ first \_ unit или даже moy \_ pervij \_ modul, потому что эти версии рассчитаны на современные тридцатидвух зарядные операционные системы, разрешающие использовать длинные имена файлов.

Первый и самый простой шаг в Delphi – программировании уже сделан: создана нормальная Windows – программа. Разумеется, эта программа не реализует придуманного вами алгоритма, она вообще ничего путнего не умеет делать.

Теперь по работаем с модулем. В первом приближении можем считать модулем самостоятельный раздел программы, в чем – то подобной главе в книге. Модуль создается каждый раз, когда вы создаете новую форму (в программе может быть и, чаще, бывает не одна, а несколько – иногда несколько десятков форм и связанных с ними модулей). При компиляции программы Delphi создает файлы с расширениями pas, dfm, dcu для каждого модуля: pas – файл содержит копию текста из окна кода программы, в файле с расширением dfm хранится описание содержимого окна формы, а в dcu – файле – результат преобразования в машинные инструкции текста из обоих файлов. Файлы dcu создаются компилятором и дают необходимую базу для работы компоновщика, который преобразует их в единый загружаемый файл с расширением EXE.

Попробуем модифицировать программу, например, изменим заголовок ее окна. По умолчанию (без каких либо усилий с нашей стороны) заголовок окна совпадает с заголовком формы: Form1. Чтобы изменить заголовок, нужно обратиться к окну Инспектора объектов. Закройте окно работающей программы Form1 и щелкните мышкой по строке Caption (заголовок) окна Инспектора объектов. Теперь правая колонка этой строки с текстом Form1 выделена цветом и справа от выделенного виден текстовый мигающий курсор. Переключите клавиатуру в режим ввода кириллицы и введите «моя первая программа» после чего еще раз нажмите F9. Новый прогон программы создаст окно с заголовком «моя первая программа», что несет в себе гораздо больше информации, чем просто Form1.

Простым изменением содержимого строки в окне Инспектора объектов мы добились важной перемены: изменили одно из свойств окна программы – его заголовок. Таким же образом можно изменять любое другое свойство формы.

Разместить на форме новый компонент не представляет труда. Для этого сначала нужно его выбрать (щелкнуть мышью) в палитре компонентов, а за тем щелкнуть мышью по точке свободного пространства формы, где должен располагаться левый верхний угол компонента.

Попробуем таким способам включить в окно программы компонент Lembel (метка), предназначенный для размещения различного рода надписей. Убедитесь в том что в палитре компонентов выбрана страница Standart, и щелкните мышью по кнопке (эта кнопка отображает компонент Label в палитре компонентов). Теперь щелкните мышью по форме так, чтобы компонент появился на форме и располагался левее и выше ее центра. Первоначальные размеры и положения компонента на форме легко изменяется мышью, поэтому добиваться полного сходства с рисунком

Новый компонент имеет стандартное имя Label 1, и надпись на нем повторяет это имя. Изменим эту надпись: с помощью строки caption окна Инспектора объекта введите надпись: «я программирую на Delphi». Как только вы начнете вводить новую надпись, компонента начнет меняться динамически отражая все изменения, производимые вами в окне Инспектора объекта.

Выделим надпись цветом и сделаем ее шрифт более крупным. Для этого щелкните мышью по свойству Font окна Инспектора объектов и с помощью кнопки в правой части раскройте диалоговое окно настройки шрифта. В списке Size (размер) этого окна выберите высоту шрифта 24 пункта, а с помощью списка Color (цвет) выберите нужный цвет, после чего закройте окно кнопкой ОК.