

Информатика

<http://www.insycom.ru>

Введение в программирование
на Visual Basic.

Лекция 1. Введение

Содержание лекций во 2-ом семестре

- Введение в программирование в среде Visual Basic
- Реляционные базы данных в СУБД Access
- Программирование на Visual Basic for Application (VBA)

Аттестация

Соответствие баллов (кредитов) видам учебной деятельности

Виды учебной деятельности	Баллы (максимально возможное количество)	Сроки сдачи
Лекции (34)	10 (посещение всех лекций)	в соответствии с расписанием
Лабораторная работа (8 занятий)	5 (40)	в соответствии с расписанием
Расчетно-графическое задание(1)	30	РГЗ – до 17 недели
Зачет	20	Зачетная неделя

Характеристика работы студента	Диапазон баллов рейтинга	Оценка ECTS	Традиционная (4-уровневая) шкала оценки	
«Отлично»	90-100	A+	отлично	зачтено
		A		
		A-		
«Очень хорошо»	80-89	B+	хорошо	
		B		
		B-		
«Хорошо»	70-79	C+	удовлетворительн о	
		C		
		C-		
«Удовлетворительно»	60-69	D+	удовлетворительн о	
		D		
		D-		
«Посредственно»	50-59	E		
«Неудовлетворительно»	25-49	FX	неудовлетво- рительно	не зачтено
«Неудовлетворительно»	0-24	F		

Технологии программирования

- Алгоритмическое программирование
 - Один исполняемый модуль, последовательность действий для решения задачи (Fortran, C, Pascal)
- Логическое программирование
 - Декларативное программирование, решение задачи – доказательство (Prolog, SmallTalk)
- Функциональное программирование
 - Решение задачи – вызов функций (в частности, рекурсивных) (C, Lisp)
- Модульное программирование
 - Набор модулей, разрабатываемых отдельно, задача – набор подзадач (C, Pascal, Modula)
- Структурное программирование
 - Структуризация алгоритмов, алгоритм – набор вложенных стандартных конструкций
- Объектно-ориентированное программирование
 - Структуризация данных и алгоритмов, объекты, иерархия классов, методы классов, наследование свойств
- Визуальное программирование
 - Визуальные компоненты, ориентация на программирование пользовательского интерфейса (Visual Basic, Visual C++, CBuilder, Delphi)
- Событийно-управляемое программирование
 - Программирование реакции на события (Visual Basic, CBuilder, Delphi)

Объектно-ориентированное программирование

- Для описания состояния решаемой задачи помимо переменных используются объекты
- Объект характеризуется именем *name* и классом *class* (аналог типа переменной)
- Объект имеет свойства *property* (аналог переменной) и методы (процедуры или функции). Они описаны в описании его класса.
- Используется иерархия классов с наследованием свойств и методов

Визуальное программирование

- Программирование сводится к конструированию программы с использованием т.н. *визуальных компонент* - объектов, которые можно выбирать из *палитры визуальных компонент* и размещать на форме приложения
- Чаще всего используются визуальные компоненты для конструирования пользовательского интерфейса программы
- Программирование (кодирование) программы сводится к написанию методов (процедур), обрабатывающих события с этими компонентами (например, «кликание» по нему мышью, движение маркера мыши в пределах компонента, загрузка объекта, изменение объекта и т.п.)

Событийно-управляемое программирование

- Событийно-управляемое программирование заключается в описании действий, которые должны быть выполнены при возникновении того или иного события для каждого объекта приложения
- Каждый объект характеризуется определенными параметрами, которые можно разбить на три категории:
 - свойства;
 - методы;
 - события.

Событийно-управляемое программирование (2)

- **Свойства** -это параметры объекта, которые определяют его внешний вид и поведение в процессе выполнения приложения. (Привести пример, объект-человек, объект окно)
- **Методы**- это команды объекта, с помощью которых задаются возможные действия над ним
- **События** – это такие параметры объекта, которые характеризуют возможные действия над ними. (например, событие click при нажатии кнопки)

Событийно-управляемое программирование (3)

- С помощью свойств управляют внешним видом и поведение объектов,
- С помощью методов задают действия, которые объект может выполнять,
- При помощи событий объект сообщает о том, что с ним происходит
- Одним из достоинств визуального событийно-управляемого программирования является легкость и простота создания достаточно сложных работоспособных приложений. Он позволяет освободить программиста от трудоемкой рутинной работы по формированию интерфейса приложения и сосредоточить основное внимание на сути задачи
- Системы программирования, реализующие эту технологию, получили название систем быстрой разработки приложений (RAD)

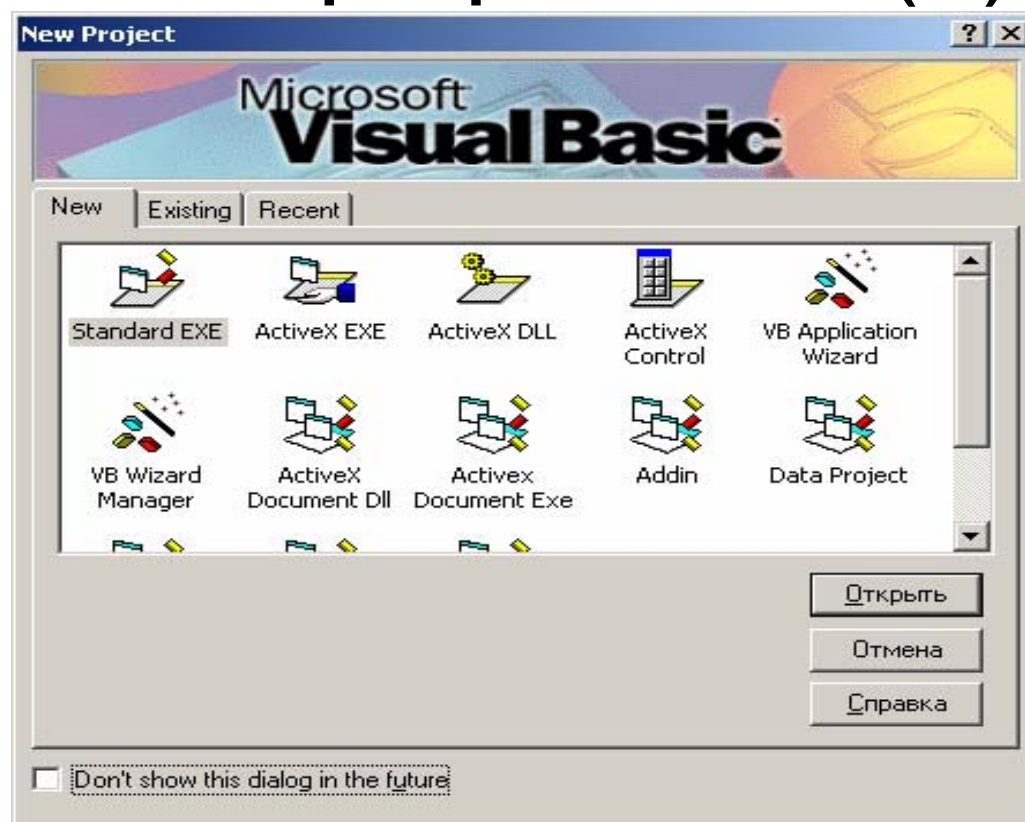
Среда разработки Visual Basic

- Состоит из инструментальных средств:
 - Транслятор,
 - компоновщик (линковщик, сборщик))
 - Отладчик,
 - Шаблоны проектов,
 - Шаблоны форм,
 - Шаблоны управляющих элементов (визуальных компонентов),
 - Настройки среды

Мастер проектов (Project Wizard)

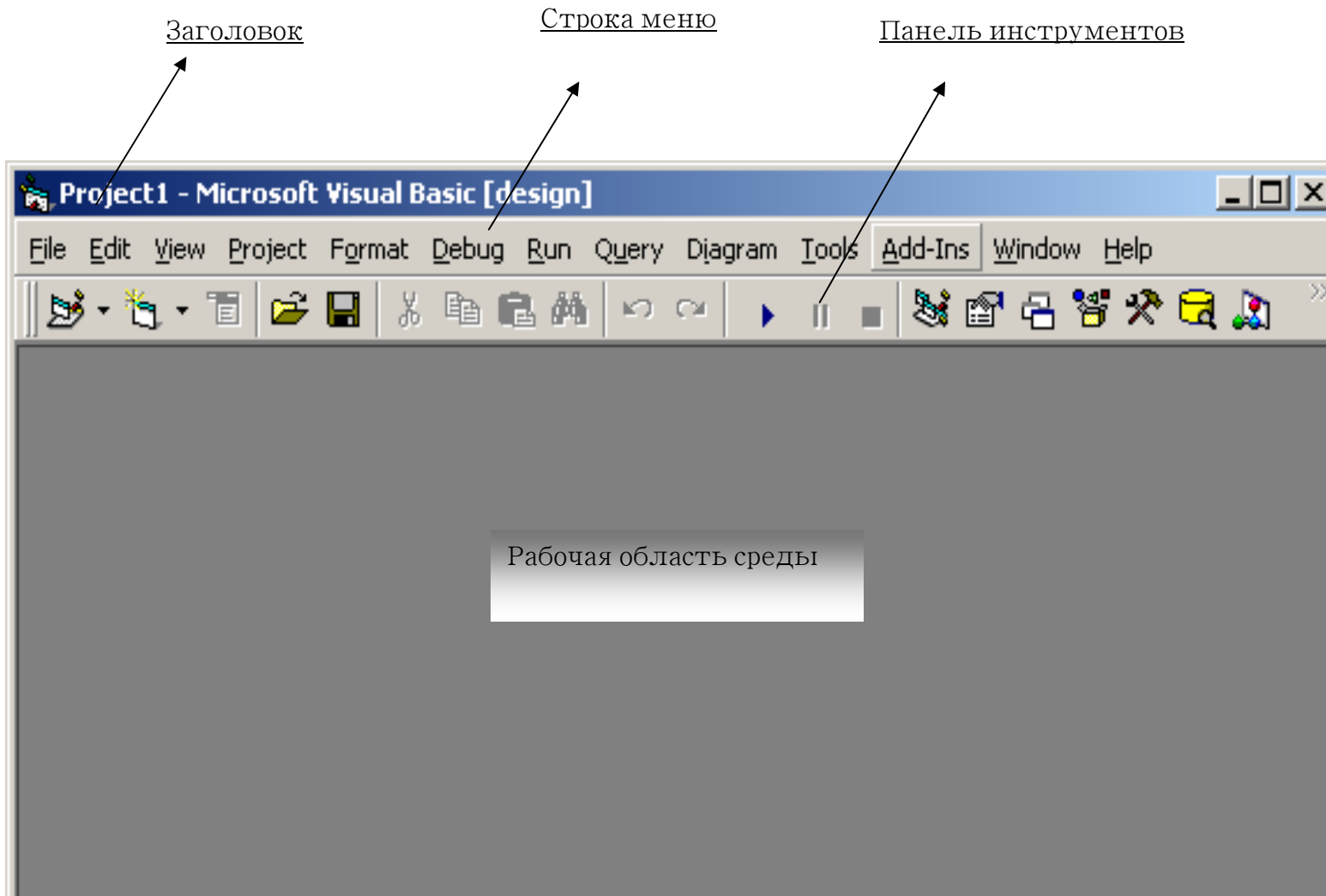
- Все приложения в Visual Basic строятся по модульному принципу, поэтому программа состоит из нескольких файлов. Каждый файл, связан с определенными компонентами (формами, программными модулями и т.д.)
- Проект - это набор файлов, которые создаются на диске в процессе разработки приложения.
- Файл с расширением .vbr содержит описания различных сочетаний компонент, используемых в приложении
- Если несколько проектов объединяются в группу, их имена сохраняются в файле с расширением .vbg

Мастер проектов (2)



Мы будем работать с шаблоном StandartEXE, в котором предусматривается установка стандартной конфигурации среды программирования для создания стандартного Windows-приложения

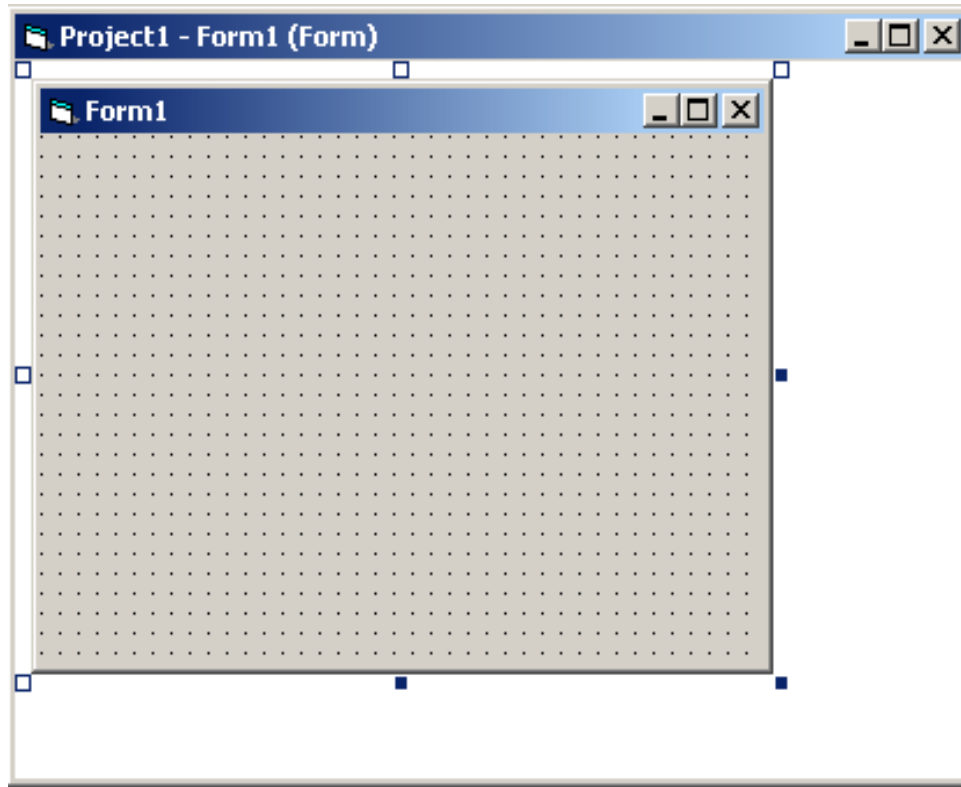
Экран среды Visual Basic



Экран среды Visual Basic (2)

- Строка меню содержит команды работы в среде.
- Панели инструментов (toolbar) упрощают доступ к наиболее часто используемым командам среды. Как правило, это панель “Стандартная” (standard).
- В рабочей области экрана располагаются окна, в которых и происходит работа программиста. Рассмотрим некоторые из них.

Конструктор форм



В начале здесь пустая форма.
В ней размещаются в процессе проектирования компоненты (элементы) – окна, кнопки и т.д.

Панель компонентов (toolbox)

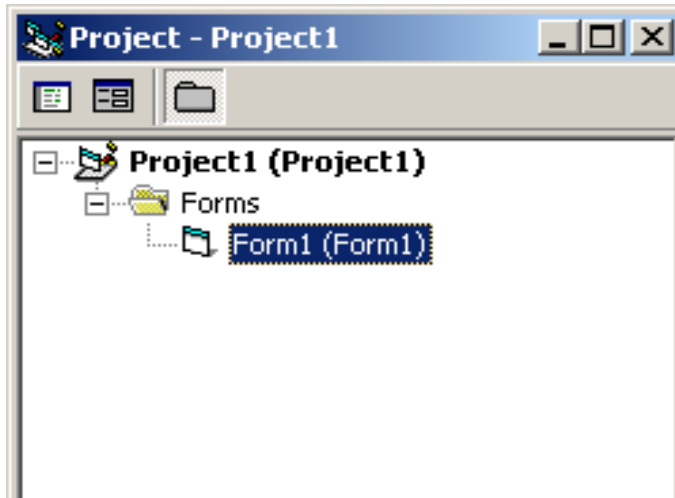


Панель элементов интерфейса содержит набор элементов, из которых формируется графический интерфейс приложения.

Это единственное окно среды, в заголовке которого название окна не указывается

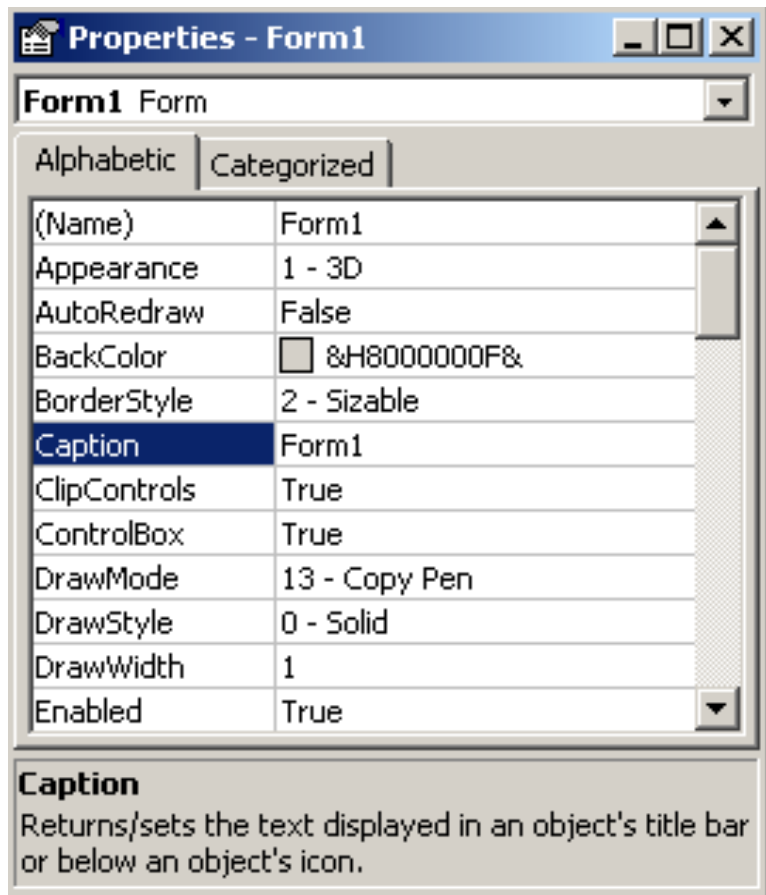
Размещаться может в разных местах экрана

Окно проекта (Project browser)



- Окно проекта (Project-Project1) предназначено для просмотра составляющих проекта, которые отображаются в нем в виде дерева.
- При создании проекта его желательно всегда помещать в отдельную папку на диске. Это позволит хранить в ней все файлы проекта в одном месте.
- Здесь же можно хранить и другие файлы, относящиеся к данному проекту, например описание приложения.

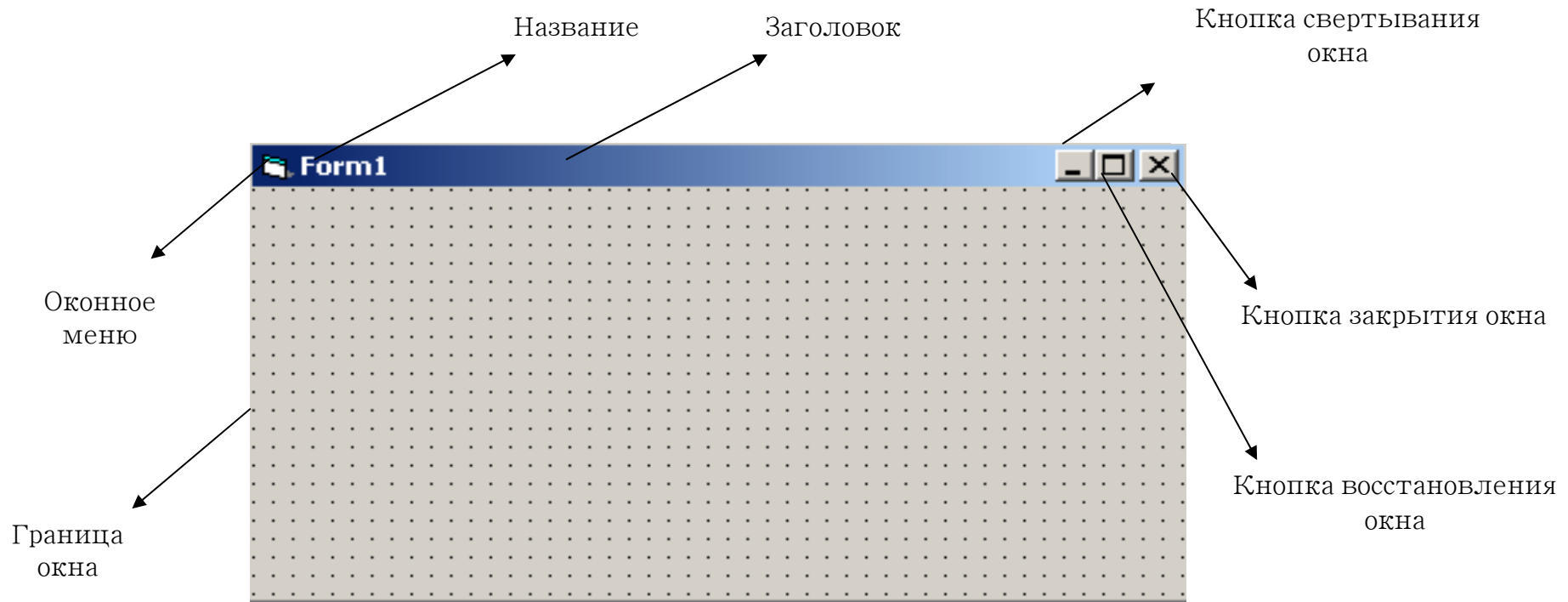
Окно свойств объектов



- В окне свойств объектов (“Properties”) отображаются свойства (атрибуты) объектов, выделенных в окне конструктора.
- Оно отличается от других окон тем, что информация в нем представляется в табличном виде

- Кроме перечисленных окон при работе в среде могут использоваться и другие (например окно кода).
- Месторасположение и визуализация окон рабочей области среды произвольны и определяется пользователем по его усмотрению

Пустая форма, имеющая стандартную структуру Windows-приложения



Панель элементов интерфейса – это инструментальный разработчика или набор объектов, которые могут быть размещены в формах.

Следует отметить, что сама форма также является элементом интерфейса.

Количество и виды объектов, отображаемых на панели элементов, зависят от настройки среды. Большинство из них являются встроенными или стандартными

Окно свойств

- После создания объектов проектируемого приложения необходимо задать значения их свойств.
- Значения свойств объектов могут устанавливаться как при разработке приложения, так и при исполнении программы.
- В первом случае работа в среде происходит в режиме конструирования (**design time**). Во втором - в режиме выполнения (**run time**).
- Режим работы указывается в заголовке окна среды после названия проекта в квадратных скобках.
- Проще всего значения свойств устанавливать в окне свойств в процессе разработки приложения
- **Окно свойств** – окно, в котором располагается список свойств объектов. Окно свойств состоит из двух колонок: в первой приведены наименования свойств, которыми обладает объект, во второй – установленные значения

Свойства объектов

- **По способу установления значения свойств их можно разделить на три группы.**
 - Свойства, произвольно задаваемые пользователем.
 - Свойства, значения которых выбираются из заранее определенного списка.
 - Свойства, для установления значений которых используется специальное окно диалога
- **Список свойств может отображаться в двух видах:**
 - Упорядоченный по алфавиту (закладка Alphabetic).
 - Сгруппированный по категориям (закладка Categorized)

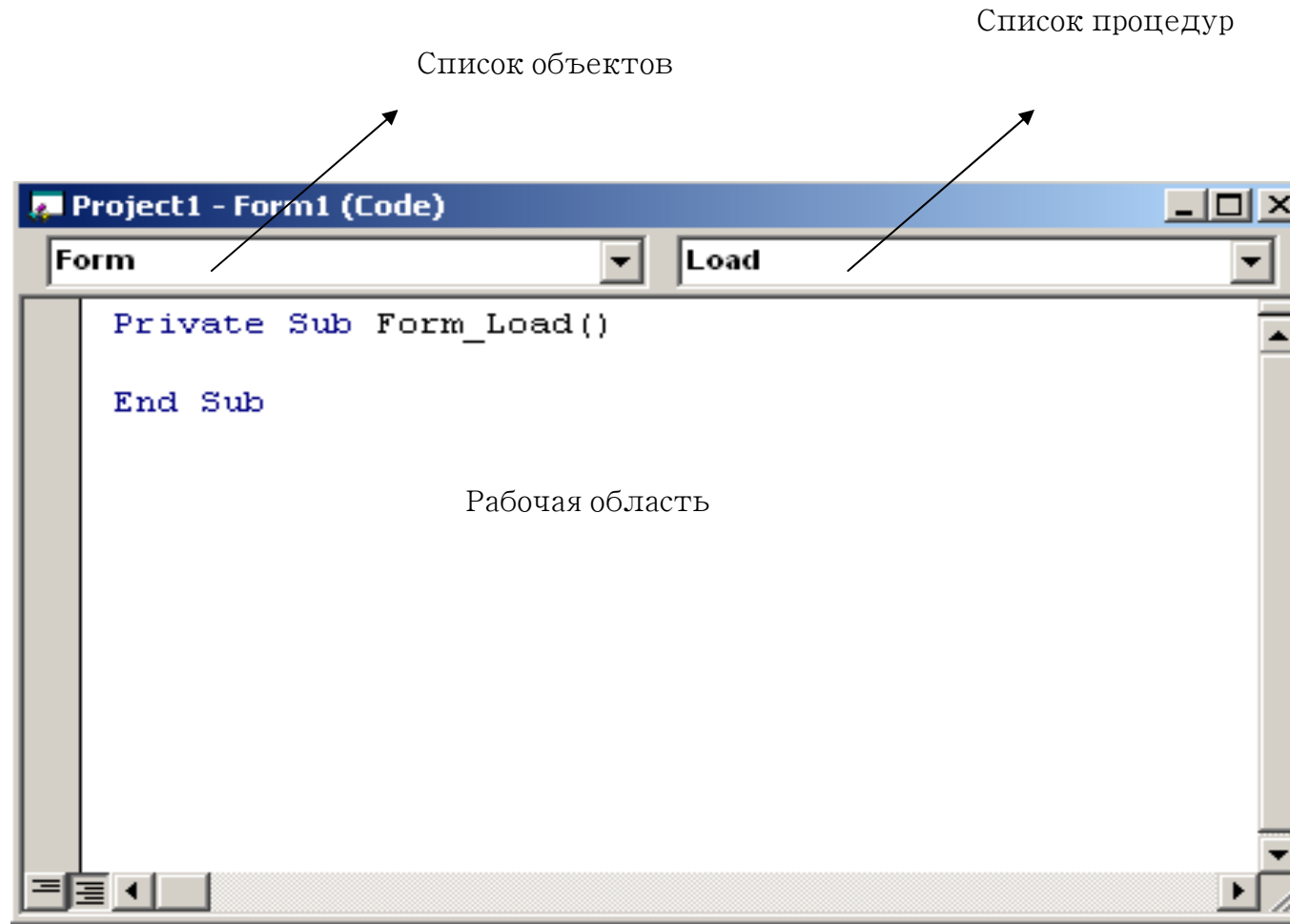
Категории свойств

- **Appearance** – свойства, определяющие внешний вид интерфейса
- **Behavior** – свойства, определяющие поведение элемента интерфейса.
- **Data** – свойства, используемые для установления связи элемента интерфейса с базами данных.
- **Font** – группа из единственного свойства, с помощью которого задается шрифт, используемый элементом интерфейса.
- **Position** – свойства, определяющие расположение и размеры элемента интерфейса

Свойства Name и Caption

- Каждый элемент интерфейса имеет свой набор свойств. Но имеется несколько свойств, которые являются общими для большинства объектов.
- К таким относится свойство Name(имя). Его значение используется в подпрограммах обработки событий для идентификации объекта. Для упрощения обращения в программе к объектам им следует задавать имена по всем правилам задания имен переменных.
- Вместе с этим принято в именах переменных использовать трехбуквенный префикс, который характеризует тип объекта. Оставшаяся часть имени должна описывать назначение объекта. Например, имя frmMain. Здесь префикс frm означает, что данное имя принадлежит объекту типа Form, а остальная часть имени свидетельствует о том, что это главная форма проекта.
- Кроме свойства Name очень часто встречается также свойство Caption – надпись на объекте. Это свойство имеют почти все объекты.

Окно кода



Окно кода (2)

- Оно представляет собой окно специального текстового редактора, предназначенного для ввода и редактирования текста программ. Для открытия окна кода нужно выполнить команду строчного меню View-Code или дважды щелкнуть по объекту, для которого предполагается писать процедуру обработки событий. Для каждой компоненты проекта открывается свое окно кода. Структура окна представлена на рисунке.
- В списке объектов окна находится перечень всех объектов, которые были размещены на форме. В списке процедур содержится перечень всех возможных событий, для выбранного объекта в списке.
- Когда в списке процедур выбирается то или иное событие, то тем самым среде дается команда на создание соответствующей процедуры обработки этого события. При этом в рабочей области окна кода отображается шаблон процедуры, который выглядит:

Private Sub<имя процедуры>

End Sub

Имя процедуры состоит из имени объекта и имени события, разделенных знаком подчеркивания

Создание программы в среде Visual Basic

- Разработка экранной формы, заключающаяся в размещении на ней элементов интерфейса.
- Определение свойств каждого элемента интерфейса.
- Разработка программ по обработки событий
- После этого программа запускается и отлаживается. Запуск производится командой меню Run-Start или Run-Start With Full Compile. В первом случае транслятор работает в режиме интерпретации, во втором - в режиме компиляции

Ошибки

- **Ошибки бывают четырех типов:**
 - Синтаксические
 - В структуре программ
 - Ошибки выполнения программ
 - Логические

Синтаксические ошибки

- **Синтаксические ошибки** распознаются сразу же, как только курсор будет переведен на другую строку. При этом выдается сообщение с пояснением ошибки. Строка с синтаксической ошибкой выделяется красным цветом.
- Кроме того, в Visual Basic встроены средства, которые позволяют избежать ошибок. Одно из них - это механизм контекстной подсказки, в которой отображается полный синтаксис вводимого оператора. Другое - автоматическое отображение списка доступных имен, свойств, методов, полей. Содержимое списка зависит от типа объекта

Ошибки в структуре программы и логические

- **Ошибка в структуре программы** – это синтаксические ошибки в многострочных операторах ветвления или цикла. Они распознаются не при вводе текста программы, а при его компиляции. При этом ошибочный оператор выделяется и выдается сообщение с пояснением ошибки.
- Наиболее трудно обнаруживаемые ошибки – **логические**. Для их поиска и устранения в среде имеются эффективные инструменты, обратиться к которым можно из пункта меню Debug

Средства отладки

- Режим трассировки программы, при котором строки кода программы последовательно выполняются одна за другой с остановкой исполнения в заданных точках
- **Step Into (F8)** – шаг с заходом. В этом случае остановка выполнения программы происходит после выполнения каждого оператора. Если это оператор вызова процедуры или функции, то происходит переход в эту процедуру или функцию.
- **Step Over (Shift+F8)** – шаг с обходом. В отличие от предыдущей команды вызов процедуры выполняется как один шаг, т.е. без захода в процедуру. Эта команда полезна при поиске ошибок в процедурах, содержащих вызовы других процедур.
- **Step Out (Ctrl+Shift+F8)** – команда, позволяющая выполнить оставшуюся часть текущей процедуры и возвратиться в точку вызова.
- **Step to Cursor (Ctrl+F8)** – команда выполнения программы до текущей позиции курсора. В этом случае остановка выполнения программы происходит после исполнения строки кода, в которой находится курсор, после чего среда переводится в режим трассировки

Средства отладки (2)

- Контроль значений переменных и выражений. Он возможен только в режиме отладки программы.
- Для реализации механизма контроля имеется несколько способов. Самый простой – использование окна значения, которое появляется на экране при позиционировании указателя мыши на соответствующем объекте в окне кода программы. Другой способ – использование окна контрольного значения (окна *Watches*), вызываемого командой меню **View-Watch Window**. В нем отображается список контролируемых выражений и их текущие значения.
- Кроме названных инструментов поиска логических ошибок имеются и другие. Например, использование окна отладки (окно *Immediate*)