



Туты меню

Меню (англ. *menu*, фр. *menu*) — элемент интерфейса пользователя, позволяющий выбрать одну из нескольких перечисленных опций программы. В современных операционных системах меню является важнейшим элементом графического интерфейса пользователя.

Элементы меню

Пункты меню могут выбираться пользователем любым из указательных устройств ввода, предоставляемых электронным устройством.

пункт меню — отдельные опции приложения.

пиктограмма, иллюстрирующая действие

разделитель (визуально разделяет группы однородных пунктов меню)

«упор» ограничивает ход сдвига курсора

Элементы меню обычно сгруппированы в:

- Строка меню
- Всплывающее меню

Строка меню (англ. *menu bar*) — основная часть меню, которая постоянно находится в окне приложения (реже, скрывается и появляется при определённых действиях пользователя). Данная строка является так называемым главным меню окна (англ. *main menu*) или меню верхнего уровня (англ. *top-level menu*), которая может содержать

Всплывающее меню (англ. *popup menu*) или подменю (англ. *submenu*). Выбор элемента главного меню обычно приводит к вызову появляющегося под главным подменю, которое в свою очередь может содержать подменю.

Таким образом меню образует иерархическую структуру функциональных возможностей приложения.

Различают следующие типы меню:

по исполнению:

- текстовое
- графическое;

по функциям:

- главное меню приложения;
- всплывающее меню;
- контекстное меню
- системное меню.

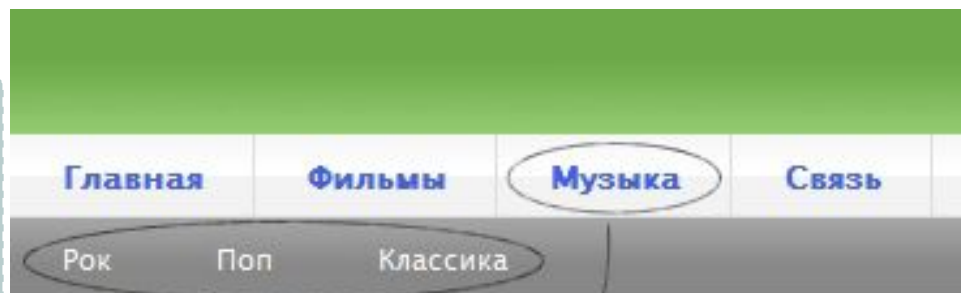


Контекстное меню - преимущество контекстных (всплывающих) меню заключается в том, что они полностью встраиваются в контекст действий пользователей: не нужно переводить взгляд и курсор в другую область экрана, практически не нужно прерывать текущее действие для выбора команды.

Выделяют также:

- **Статические меню**, т.е. меню, постоянно присутствующие на экране. Характерным примером такого типа меню является панель инструментов.
- **Динамические меню**, в которых пользователь должен вызвать меню, чтобы выбрать какой-либо элемент. Примером является обычное контекстное меню.
- **Меню, разворачивающиеся в пространстве** (например, обычное выпадающее меню). Всякий раз, когда пользователь выбирает элемент нижнего уровня, верхние элементы остаются видимыми.
- **Меню, разворачивающееся во времени**. При использовании таких меню элементы верхнего уровня (или, понимая шире, уже пройденные элементы) по тем или иным причинам исчезают с экрана.

Статические меню как правило, обеспечивают меньшую скорость работы, лучше обучают пользователей, но зато занимают место на экране. С динамическими меню ситуация обратная.



При наведении на ссылку появляется соответствующее подменю

Динамическое меню

Default CSS Rollovers - No JavaScript



Blend Default: `$(selector).blend();`



Blend+Pulse: `$(selector).blend({pulse:true});`



Blend+Reverse: `$(selector).blend({reverse:true});`



Статическое меню

Меню, разворачивающиеся в пространстве обеспечивает большую поддержку контекста действий пользователей, но эта поддержка обходится в потерю экранного пространства.

Типы меню в Access

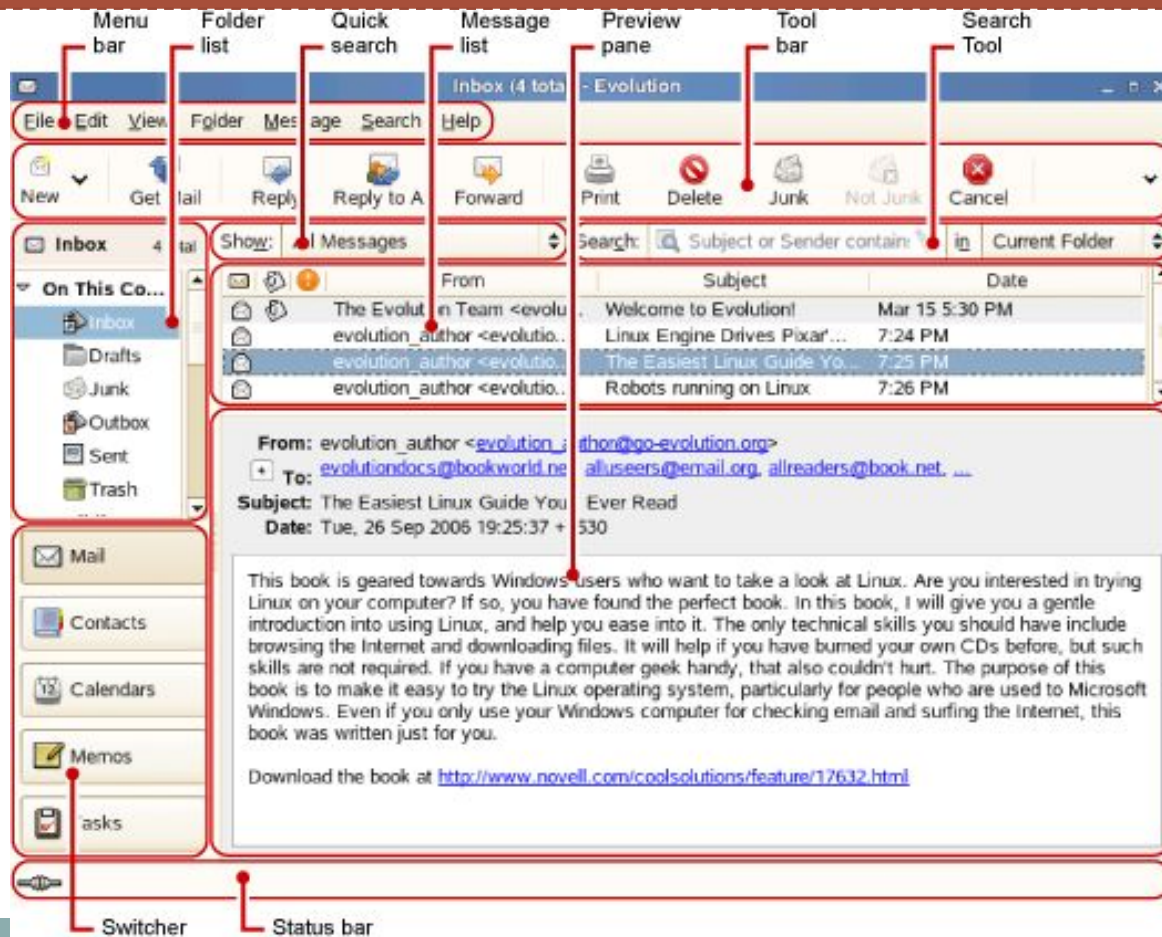
С каждым объектом в Access из них нужно работать в интерактивном режиме, открывая в окне базы данных вкладку требуемого объекта.

Можно связать объекты в единое приложение. При этом не требуется умение создавать сложные программы. Благодаря использованию кнопочных форм, диалоговых окон и меню работу с приложением можно автоматизировать и упростить, скрывая от пользователя ненужные в текущий момент объекты Access.



В Access можно создать три типа меню:

1. **Menu Bar.** Обычное меню, которое может располагаться вверху формы и иметь выпадающие подменю.
2. **Tool Bars.** Группы пиктограмм, обычно располагающиеся под меню.
3. **Shortcut Bar.** Меню, всплывающие после щелчка правой кнопки мыши.



Кнопочная форма представляет собой созданную пользователем форму с кнопками. С их помощью можно, например, запускать макросы открытия форм и печати отчетов. С помощью кнопочной формы выполнить несколько действий можно, щелкнув на одной кнопке. Используя кнопочную форму и другие объекты можно объединять созданные ранее объекты базы данных в одном приложении (при этом приложение может иметь вид, отличный от внешнего вида Access).



В основном кнопочная форма используется в качестве главного меню приложения.

Поскольку кнопочная форма выполняет только роль меню приложения, в ней следует использовать минимальное число элементов управления. Поэтому обычно в кнопочную форму включают только кнопки, надписи, рамки объектов, линии и прямоугольники.

«Особенности использования и программирования каждого типа меню»

Создавая приложения важно уделять внимание интерфейсу пользователя: от того, насколько удобно будет пользоваться программой во многом зависит успех разрабатываемого приложения. В современных приложениях используется множество элементов управления, не последнюю роль играют и меню.



Устройство меню

На эффективность меню наибольшее влияние оказывают устройство отдельных элементов и их группировка. Несколько менее важны другие факторы, такие как выделение элементов и стандартность меню.

Программирование каждого типа меню

Средства макропрограммирования в MS Access

Access, как и любая другая развитая программная система, обладает средствами разработки программных приложений, ориентированных на конечных пользователей. Эти средства базируются на инструментах двух типов: *макросах и модулях*.

Понятие макроса подразумевает наличие набора некоторых стандартных команд системы, или макрокоманд (допустим, таких, как открытие формы, выполнение запроса, вывод отчета), из которых и конструируется сам макрос.



Макрос может быть как собственно макросом, состоящим из последовательности макрокоманд, так и группой макросов. *Группой макросов называют их набор, сохраняемый под общим именем.* В некоторых случаях для решения, должна ли в запущенном макросе выполняться определенная макрокоманда, может применяться условное выражение.

Модули, в отличие от макросов, являются более тонким и мощным средством создания программных расширений в среде Access, максимально приближающимся по своим функциональным возможностям к таким профессиональным инструментам, как Delphi, Visual Basic или Power Builder.

The image shows the logo for Microsoft Visual Basic 6.0 SP6. The text "Microsoft" is in a smaller font above "Visual Basic 6.0", which is in a large, bold, black font. "SP6" is in a smaller font to the right of "6.0". The background features a blue, 3D-style graphic of a computer keyboard.

Для программирования в Access используется процедурный язык *Visual Basic для приложений* с добавлением объектных расширений и элементов SQL. Сам процесс создания программных расширений в среде Access предполагает активное использование технологии *объектно-ориентированного программирования* (ООП). В основе ООП лежит идея "упакованной функциональности", в соответствии с которой программа строится из фундаментальных сущностей, называемых объектами.

Устройство отдельных элементов

Пиктограммы в меню

Пиктограммы в меню, если они повторяют пиктограммы в панели инструментов, обладают замечательной способностью обучать пользователей возможностям панели. Помимо этого они ускоряют поиск известного элемента и точность его выбора, равно как и общую разборчивость меню.

Пиктограммы лучше работают, когда ими снабжены не все элементы. Когда все элементы имеют пиктограммы, разборчивость каждого отдельного элемента падает: в конце концов, пиктограммы всех ненужных в данное время элементов являются визуальным шумом.



Не следует снабжать пиктограммами все элементы меню, а только самые важные.

Предсказуемость действия

Пользователей нужно снабжать чувством контроля над системой.

Применительно к меню это значит, что по виду элемента пользователи должны догадываться, что произойдет после выбора. Сделать это невероятно трудно, поскольку на экране нет места под такие подсказки. Можно сделать только одно, но сделать это нужно обязательно: нужно показать пользователям, какой элемент запускает действие или меняет параметр, а какой открывает окно с продолжением диалога.



Почти во всех ОС стандартным индикатором продолжения диалога является многоточие после текста элемента, так что пользоваться этим признаком стоит везде, включая интернет. Также необходимо показывать, какой элемент срабатывает сразу, а какой открывает элементы меню нижнего уровня.

Группировка элементов

В большинстве меню группировка оказывает не меньшее значение при поиске нужного элемента, нежели само название элемента, просто потому, что даже идеальное название не сработает, если элемент просто нельзя найти.

Чтобы уметь эффективно группировать элементы в меню, нужно знать ответы на три вопроса: зачем элементы в меню нужно группировать, как группировать элементы и как разделять группы между собой.

Работа элементов меню

Все элементы меню или часть элементов меню могут работать как настоящее меню, т.е. как группа независимых счётных кнопок (счётных триггеров, Т-триггеров) из которой, как в настоящем меню, можно выбрать один, два или более пунктов, но чаще встречаются группы зависимых кнопок, которые работают как многофазный триггер, т.е. из группы пунктов можно выбрать только один пункт, после выбора которого прежде выбранный пункт сбрасывается, но это уже не меню, а переключатель.

Глубина меню

Наличие многих уровней вложенности в меню приводит к так называемым **«каскадным ошибкам»**: выбор неправильного элемента верхнего уровня неизбежно приводит к тому, что все следующие элементы также выбираются неправильно. При этом широкие меню больше нравятся пользователям. Поэтому большинство разработчиков интерфейсов стараются создавать широкие, а не глубокие меню.



Недостаток широких меню: занимают много места. Это значит, что, начиная с определенного количества элементов, меню физически не сможет оставаться широким, оно начнет расти в глубину. Возникает проблема, которую надо решать.

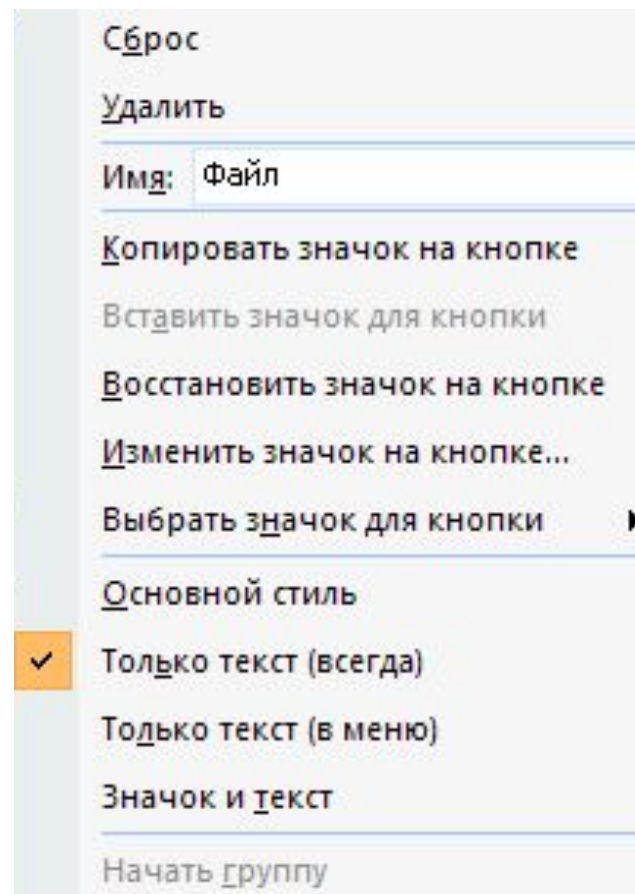
Обобщение материала

Вопросы:

1. Устройство отдельных элементов меню
2. Средства разработки программных приложений Access
3. Процедурный язык для программирования в Access
4. Специальный интерфейс Data Access Object
5. Достоинство объектной модели

Тема: «Команды управления меню»

Для манипуляции
видом и свойствами
отдельного пункта
меню
или кнопки панели
инструментов
предназначено
диалоговое окно,
своим видом и
поведением
напоминающее
контекстное меню



Создание главного меню в Microsoft Access

Для создания главного меню запускаем VBA нажатием клавиш Alt+F11, выбираем пункт **Сервис – Настройка...** (или кликаем правой кнопкой по панели инструментов и выбираем пункт **Настройка** в контекстном меню). Появляется диалоговое окно Настройка.

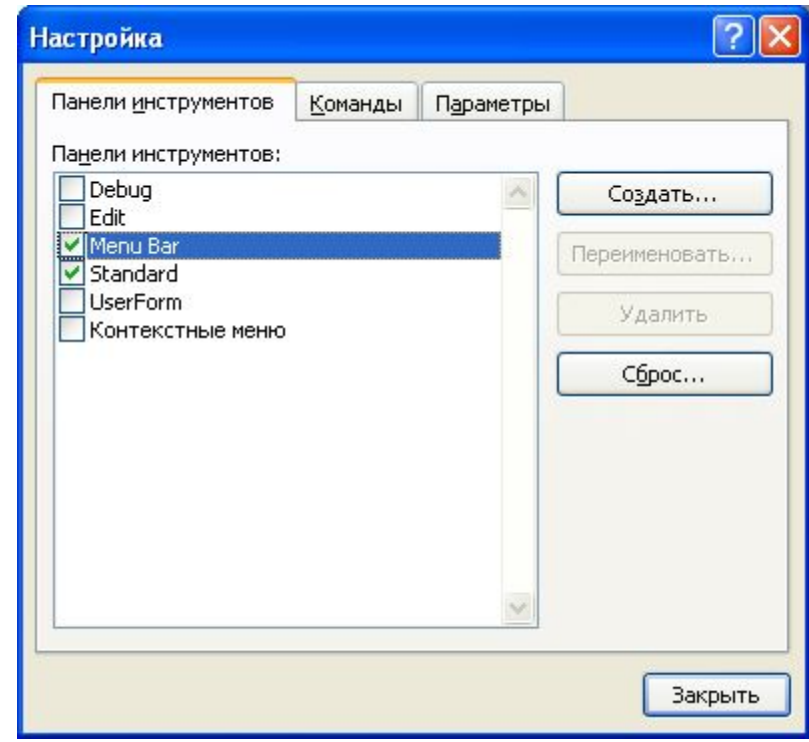
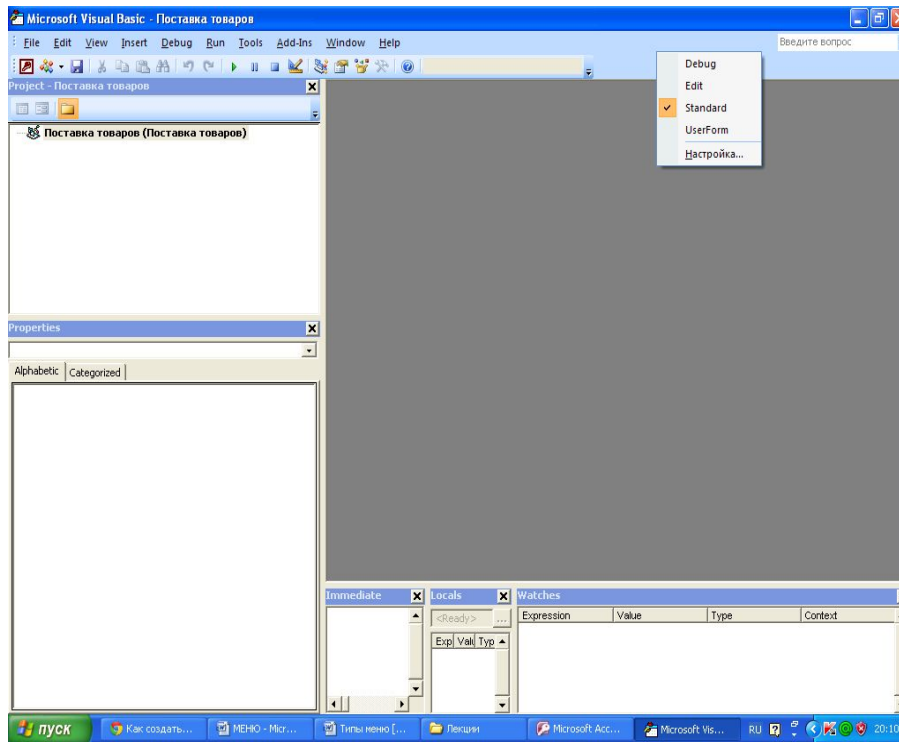


Рис. 1. Диалоговое окно Настройка

В диалоговом окне Настройка следует нажать кнопку Создать для создания новой панели инструментов для главного меню, изменить имя создаваемой панели (например на MainMenu). После проделанных операций должна появиться панель инструментов, не содержащая ни одного элемента (рис. 2).

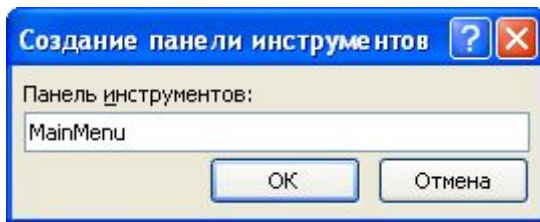
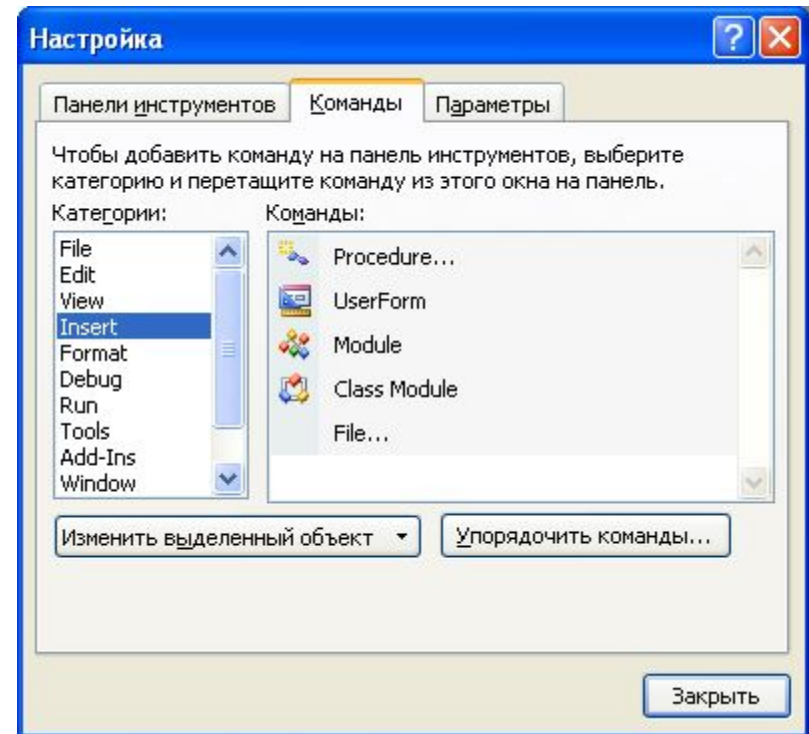


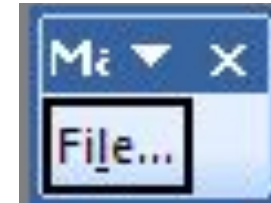
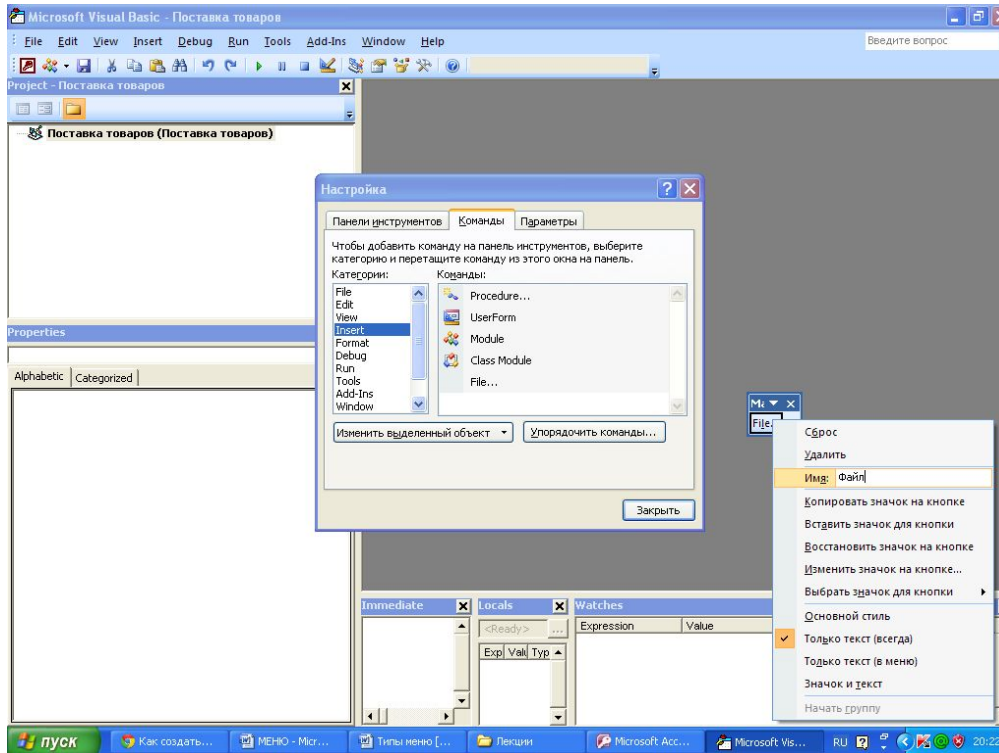
Рис - Пустая панель инструментов

Следующим этапом создания главного меню будет **добавление пунктов меню**, на созданную ранее панель инструментов MainMenu. Для этого следует в диалоговом окне Настройка (рис. 1) перейти на закладку **Команды** и из списка команды перетянуть пункт на панель инструментов MainMenu.



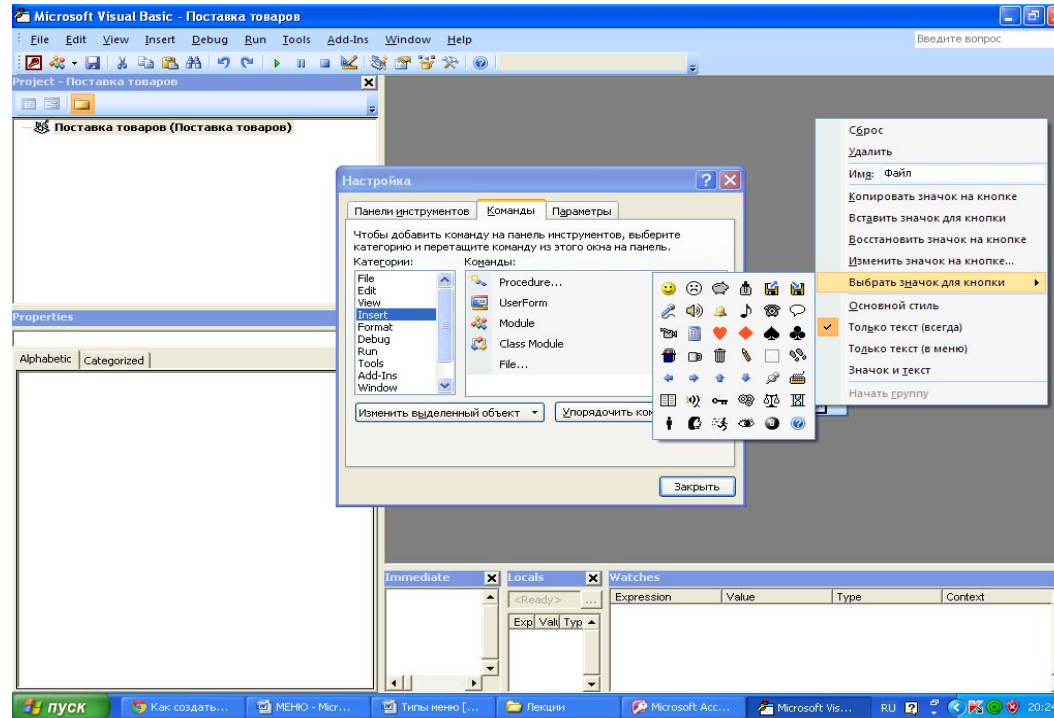
После этого можно вызвать диалоговое окно Свойства добавленного пункта меню и изменить его подпись

Результат добавления нового пункта меню:



После создания меню нужно **добавить в него команды**. Можно выбрать либо существующую команду, с которой уже связано какое-то действие, либо взять за основу команду Специальная (в категории Файл) и настроить ее в соответствии со своими требованиями.

В качестве команд удобно использовать команды из категорий *Все таблицы, Все запросы, Все формы, Все отчеты* – в этом случае элемент меню будет содержать значок таблицы, запроса, формы или отчета соответственно, кроме того с ним будет ассоциировано соответствующее действие.



Для того, чтобы добавить значок (иконку) к элементу меню нужно перейти в режим настройки панели инструментов (*Сервис – Настройка...*), кликнуть правой кнопкой на элементе меню и выбрать значок для кнопки из одноименного пункта меню. Допустимо использование собственного значка в элементе меню.