

Автономная некоммерческая организация высшего образования

«СЕВЕРО-ЗАПАДНЫЙ ОТКРЫТЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Дисциплина «Информационные технологии»

**Методические указания
к контрольной работе**

Составитель:

к.т.н., доцент Рахманова Ирина Олеговна

**Санкт-Петербург
2016**

СОДЕРЖАНИЕ

1	МЕСТО И РОЛЬ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ КРУПНЫМ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ	3
2	ЗАДАНИЕ, ОФОРМЛЕНИЕ И ЗАЩИТА контрольной РАБОТЫ	10
3	МЕТОДИКА ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ	12
4	ЭТАПЫ ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ	15
4.1	Подготовка данных в среде MS ACCESS	15
4.2	Разработка экранных форм приложения	18
4.3	Разработка сайта предприятия	19
	ПРИЛОЖЕНИЕ 1	26
	Структура базы данных Access. Работа с таблицами	26
	Элементы базы данных ACCESS	26
	Свойства таблиц Access	27
	Отношения между таблицами	32
	Обеспечение ссылочной целостности	33
	Запросы к базе данных Access	34
	Понятие запросов, виды и назначение запросов Access	34
	Основы конструирования запроса	35
	Формы и отчеты в Access	40
	Принципы создания экранных форм	40
	Типы элементов управления	41
	ПРИЛОЖЕНИЕ 2	44
	Язык разметки текста HTML	44
	Характеристика HTML как языка программирования для Интернет	44
	Базовая структура страницы WEB	46
	Стили форматирования	46
	Списки	47
	Гипертекстовые связи	48
	Графика	48
	Дополнительные возможности форматирования	49
	Создание таблиц	50
	Бланк	51
	Справочник языка HTML	52
	ПРИЛОЖЕНИЕ 3	58
	ПРИЛОЖЕНИЕ 4	59

1 МЕСТО И РОЛЬ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ КРУПНЫМ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ

В условиях жесткой конкуренции и динамичного рынка автоматизация задач планирования и управления деятельностью предприятия на основе новых информационных технологий становится решающим фактором его выживания. Основой каждой производственной системы являются воплощенные в ней рекомендации по управлению производством. Многоуровневый характер системы управления предприятием нашел свое отражение в соответствующих стандартах интегрированных информационных систем предприятия (ИИСП), которые представляют собой описание наиболее общих правил планирования и управления различными стадиями производственного процесса: от управления технологическим оборудованием до гарантийного и послегарантийного обслуживания клиента (рис. 1).

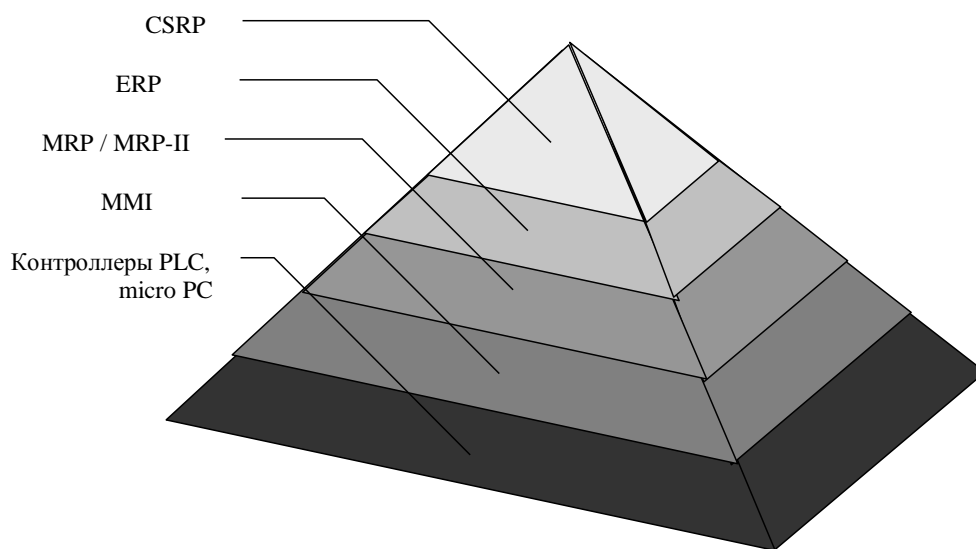


Рис. 1 Иерархия стандартов реализации многоуровневой системы управления предприятием

Стандарты ИИСП ориентируются на традиционную классификацию задач, решаемых на любом промышленном предприятии:

- технологические производственные задачи,
- административно-хозяйственные задачи.

Технологическую группу задач в иерархии управления производством поддерживает стандарт MMI (Man-Machine Interface) – «человеко-машинный интерфейс» в смысле обеспечения двусторонней связи «оператор – технологическое оборудование».

Исходным стандартом, поддерживающим административно-хозяйственную группу задач, исторически был стандарт MRP (Material Requirement Planning), включавший планирование материалов для производства. В дальнейшем этот стандарт был расширен до MRP-II (Manufacturing Resource Planning), который позволял планировать все производственные ресурсы предприятия (сырье, материалы, оборудование и т.д.). Развитием стандартов класса MRP стал стандарт ERP (Enterprise Resource Planning), объединивший все ресурсы предприятия, в том числе управление заказами, финансами и т.д. Последним по времени стандарт CSRП (Customer Synchronized Resource Planning) охватывает также и взаимодействие с клиентами: оформление наряд-заказа, техническое задание, поддержка заказчика на местах.

Каждый из перечисленных стандартов базируется на соответствующих компьютерных системах, осуществляющих задачу автоматизации управления соответствующим уровнем иерархии производственной системы.

Сегодня практически все производственные системы отвечают рекомендациям стандарта ERP. Системы ERP предназначены для управления финансовой и хозяйственной деятельностью предприятий. Это верхний уровень в иерархии систем управления предприятием, затрагивающий ключевые аспекты его производственной и коммерческой деятельности, такие как производство, планирование, финансы и бухгалтерия, материально-техническое снабжение и управление кадрами, сбыт, управление запасами, ведение заказов на изготовление (поставку) продукции и предоставление услуг.

Такие системы создаются для предоставления руководству информации для принятия управленческих решений, а также для создания инфраструктуры электронного обмена данными предприятия с филиалами, поставщиками и потребителями.

Развитие новых информационных технологий заставляет по-новому взглянуть на стратегию и тактику автоматизации современного промышленного предприятия. Современная ИИСП класса ERP обязательно имеет выход в Интернет, который, объединив локальные сети в глобальную структуру, открыл новые возможности для ведения бизнеса в едином информационном пространстве.

Как правило, современная ИИСП имеет коммуникационный модуль, поддерживающий обмен данными средствами Интернет. В основе работы данного приложения лежит трехуровневая схема, включающая следующие компоненты:

1. Корпоративный сервер управления данными и соответствующими методами их обработки;
2. Корпоративный сервер приложений (Web-сервер), отвечающий за взаимодействие с сервером управления данными и обеспечивающий подготовку требуемой информации для визуализации;
3. Универсальная программа просмотра содержимого Web-узлов - браузер.

На сервере управления данными хранится корпоративная база данных и бизнес-логика. На сервере приложений хранятся ASP-сценарии (Active Server Pages), позволяющие отображать в формате HTML данные из корпоративной базы данных и заносить в базу информацию, введенную пользователем через Web-интерфейс. Функциональная структура ИИСП данного класса представлена на рис. 2.

1. Сервер управления данными. Средства управления данными предназначены для обеспечения взаимодействия приложений и непосредственно пользователей с данными предметной области, традиционно

аккумулятивными в базах данных (БД). Одним из наиболее важных вопросов создания ИИСП является выбор модели данных и соответствующей ей реализации системы управления базой данных (СУБД). Модель данных представляет собой формализм описания совокупности взаимосвязанных структур данных и операций над ними.

В общем случае модель данных поддерживает спецификацию объектов предметной области, их классификацию и взаимосвязи между ними для адекватного отображения предметной области. Современные базы данных, как правило, строятся на основе реляционной модели данных.

СУБД является универсальным программным средством, предназначенным для создания, поддержания БД, доступа к данным и их обработки. СУБД обеспечивают многоцелевой характер использования БД, защиту и восстановление данных. Наличие развитых диалоговых средств и языка запросов высокого уровня делает СУБД удобным средством для конечного пользователя. Многопользовательские СУБД, работающие в сети, обеспечивают коллективный доступ пользователей к общей БД с защитой данных от одновременной их корректировки несколькими пользователями.

2. *Web-сервер*. Web-технология базируется на специально разработанном протоколе HTTP (HyperText Transfer Protocol — протокол передачи гипертекста). В настоящее время термин «гипертекст» ассоциируется непосредственно с WWW (World Wide Web).

Гипертекст — это текст, отдельные слова или выражения которого содержат ссылки на другие текстовые документы. При чтении такого текста можно быстро и полностью охватить тему без подробного изучения всех документов, посвященных ей. Применительно к WWW понятие гипертекста включает в себя все множество электронных документов (в том числе звуковые, анимационные файлы, протоколы, соединения и службы). В последнее время для этого используется термин «гипермедиа».

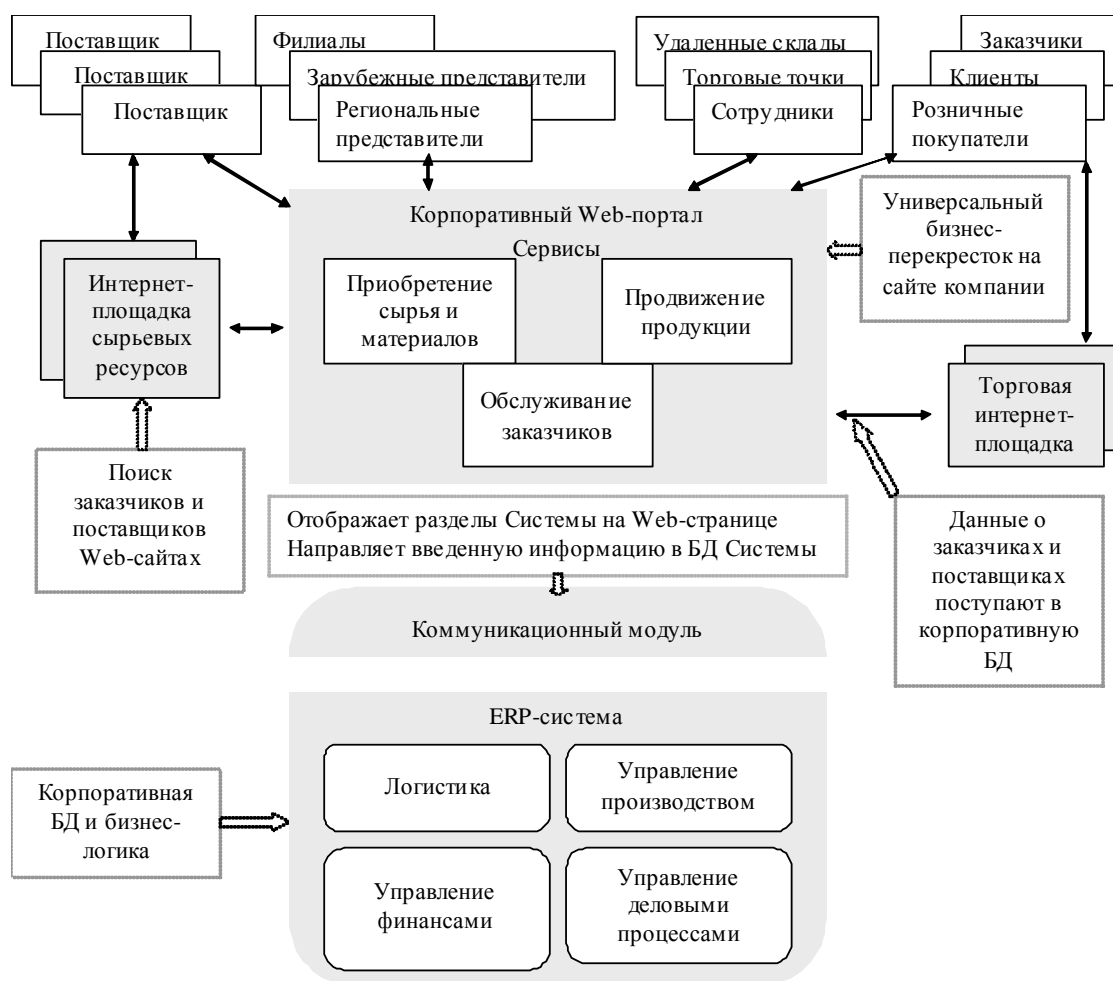


Рис. 2 Функциональная структура современной интегрированной информационной системы предприятия

Протокол HTTP обеспечивает возможность непосредственно из любого документа переходить к чтению другого текстового документа, прослушиванию музыки, установлению соединения по протоколу FTP (протокол передачи файлов), и т. д. Все эти документы и соединения могут находиться на разных компьютерах сети. По ссылке из документа на компьютере, находящемся в Европе, открывается документ на компьютере, расположенном в Австралии, Америке или другом месте.

Технология WWW позволила реально воплотить в жизнь буквальное понимание термина «сеть» на уровне пользователя, в большинстве случаев оградив его от необходимости интересоваться адресом компьютера или

ресурса, позволяя переходить от документа к документу, от соединения к соединению щелчком мыши на ссылке в документе. Процесс открытия документов на различных компьютерах в сети получил название серфинг (surfing) — плавание по волнам Web.

Для использования возможностей протокола HTTP документы должны быть подготовлены в соответствии с требованиями языка программирования HTML (HyperText Markup Language), специально разработанного для создания гипертекстовых документов. Для просмотра таких документов применяются специальные программы, называемые браузеры (browser) или программы-обозреватели.

Для определения документов WWW используется термин *страница*. Как правило, файлы, создающие одну страницу, хранятся на сервере в отдельном каталоге. Начальная страница, выводимая первой при обращении к Web-серверу и имеющая ссылки на другие страницы и документы, называется *основной* или *домашней страницей* (home page). Для определения наборов страниц, посвященных одному вопросу, или поддерживаемых одной службой, или принадлежащих одной организации и использующих отдельную домашнюю страницу для входа со своим адресом, используется термин *сайт* (site — место, участок, расположение). При этом физически на одном компьютере может находиться множество сайтов.

Протокол HTTP позволяет программе-обозревателю обращаться к ресурсам не только WWW, но и к другим, для части из которых она имеет встроенную поддержку, например, для работы по протоколу FTP. К другой части ресурсов можно обращаться через специальные шлюзы, предоставляющие возможность связи своего протокола.

Подводя итог вышеизложенному, следует подчеркнуть актуальность решения проблемы создания автоматизированных систем управления сложными производственными системами на базе новых информационных

технологий. К числу современных инструментов разработки ИИСП следует отнести:

- технологию управления информационными массивами с помощью систем управления базами данных;
- технологию визуального программирования, позволяющую разрабатывать компьютерные приложения, работающие с данными и обеспечивающие пользовательский интерфейс к ним;
- технологию программирования для Интернет, обеспечивающую информационную коммуникацию предприятия с внешней средой.

2 ЗАДАНИЕ, ОФОРМЛЕНИЕ И ЗАЩИТА КОНТРОЛЬНОЙ РАБОТЫ

Цель контрольной работы: освоение современных информационных технологий на примере разработки компьютерного приложения, реализующего функции учета готовой продукции предприятия машиностроения (или предприятия иной отрасли) и поддержку электронной формы ее реализации средствами Интернет.

Выполнение контрольной работы предполагает наличие знаний по дисциплине «Информатика» и навыков работы с приложениями Microsoft Office.

Разделение контрольной работы на варианты не предусмотрено.

Используемые информационные технологии. При выполнении контрольной работы используются следующие информационные технологии:

1. Технология управления базами данных в среде Microsoft Access.
2. Технология визуального программирования — создания экранных форм в среде Microsoft Access средствами встроенного кодогенератора (мастер форм).
3. Технология создания Web-документов (Web-дизайна) для Интернет средствами языка разметки HTML.

Техническая база реализации контрольной работы. Контрольная работа выполняется на базе компьютера PC с операционной системой Windows XP и последующие версии, с использованием приложений Microsoft Office (Microsoft Word и Microsoft Access) и просмотрной программы (браузера), например, Internet Explorer 6.0 и выше.

Исходные данные для выполнения контрольной работы:

Контрольная работа выполняется на основе частично заполненной базы данных, отражающей сбытовую (торговую) хозяйственную деятельность предприятия. Вид товара в данном контексте не имеет значения. В исходном файле magazine.mdb содержатся данные по строительным товарам, однако это

могут быть товары любого назначения. Так в методических указаниях применение информационных технологий иллюстрируется на примере продукции машиностроительного предприятия.

Таким образом, при выполнении контрольной работы можно использовать как частично заполненный файл базы данных `magazine.mdb` с товарами строительного направления, так и создавать базу данных с нуля, используя текстовые данные из методических указаний, относящиеся к машиностроительной продукции. Вид продукции является лишь контентом и не влияет на используемые в работе информационные технологии.

Итак, будем считать, что исходный файл базы данных в формате `.mdb` (MS Access) содержит:

- таблицу-справочник параметров двигателей (электро-механических модулей) различного типа, подлежащих реализации в качестве готовой продукции предприятия машиностроения – «Справочник продукции» (пример справочника по электро-механическим модулям представлен в Приложении 3);
- таблицу-справочник реквизитов заказчиков готовой продукции предприятия машиностроения – «Справочник заказчиков» (пример справочника по электро-механическим модулям представлен в Приложении 3);
- таблицу оперативных данных с реквизитами заказов – «Заказы».

Файл в формате `.doc` (MS Word), содержащий подробную информацию о свойствах каждого типа готовой продукции, области применения, назначении и т.д.

Графические файлы в формате `.jpg`, представляющие собой иллюстративный материал по готовой продукции.

Форма представления результатов – работающие компьютерные программы:

- в среде Access – приложение «Система оперативного учета заказов».
- в среде браузера Internet Explorer – сайт предприятия.

3 МЕТОДИКА ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ

Контрольная работа предполагает в качестве конечного результата реализацию следующей схемы взаимодействия программных приложений (рис. 3):

1. В среде MS Access разрабатывается база данных, содержащая следующие основные элементы (рис. 5):

а) Таблицы:

- исходная таблица с параметрами продукции «Справочник продукции»,
- исходная таблица с реквизитами заказчиков «Справочник заказчиков»,
- исходная таблица с оперативными данными по каждому заказу «Заказы»;

б) Запросы:

- запрос «Отгрузка» на выборку данных из исходных таблиц для расчета производных параметров заказа, включая его количественные и стоимостные показатели;
- перекрестные запросы «Продукция-Количество» и «Заказчик-Стоимость» для анализа сбытовой деятельности предприятия по объемным (стоимость заказов) и номенклатурным показателям;
- запросы по категориям продукции с ценами для их последующего отображения на страницах сайта (например, гиперссылка «Прайс_двигатель», рис. 8).

в) Отчеты:

- отчет «Накладная», отражающий форму сопроводительного документа по каждому заказу;
- сводные отчеты «Продукция-Количество» и «Заказчик-Стоимость» для анализа сбытовой деятельности предприятия за плановый период времени.

г) Формы (рис. 7):

- исходная форма приложения «Система оперативного учета заказов»
 - производная форма «Ввод заказа» для ведения основных данных по сбытовой деятельности путем заполнения исходных таблиц.
 - производная форма «Результаты продаж» для поддержки аналитической деятельности и принятию решений в сфере сбыта готовой продукции
 - производные формы, содержащие диаграммы по данным перекрестных запросов «Продукция-Количество» и «Заказчик-Стоимость»
2. Средствами языка HTML подготавливается сайт, структура которого представлена на рис 8. Редактором разработки сайта служит программа «Блокнот», поставляемая в комплекте с ОС Windows. Сайт предназначен для автоматизации сбытовой деятельности предприятия с помощью Интернет-технологий. Его цель – информирование потенциальных заказчиков о новых видах готовой продукции предприятия, их ценах и в конечном варианте – прием и оформление заказа на поставку продукции собственного производства.
3. В среде MS Word подготавливаются данные по каждому виду готовой продукции в соответствии с предложенным шаблоном. Эти описания подлежат опубликованию на сайте предприятия и предназначены для более широкого информирования потенциальных заказчиков о качестве и свойствах продукции предприятия.

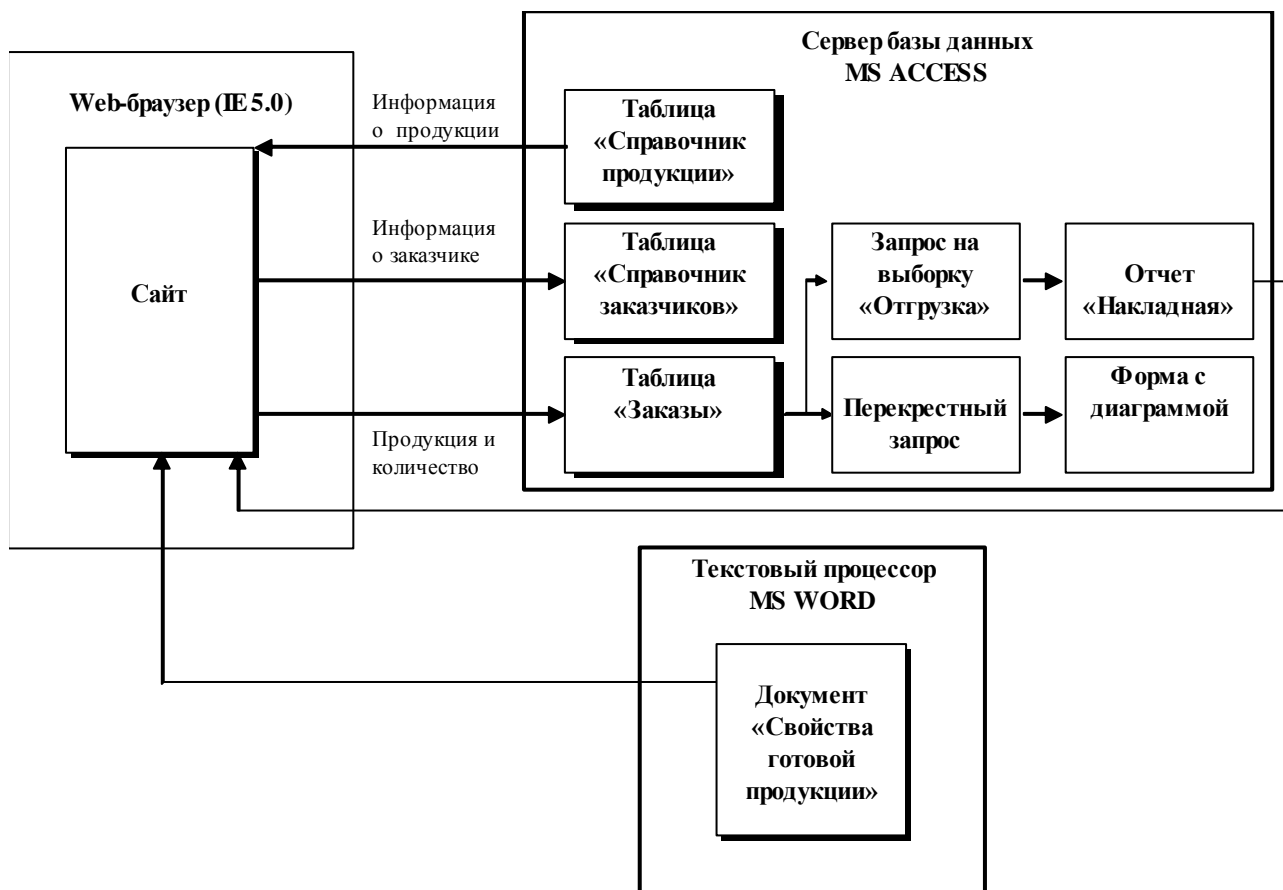


Рис. 3 Схема взаимодействия стандартных приложений MS Office и Интернет в рамках контрольной работы

4 ЭТАПЫ ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ

4.1 Подготовка данных в среде MS ACCESS

Структуры таблиц, запросов и отчетов представлены на рис. 5.

1. Проверить таблицу «Справочник продукции» (закладка «Таблица»). Поле *Код продукции* должно быть назначено ключевым. Поле *Свойство продукции* необходимо связать через механизм OLE с соответствующими фрагментами документа «Свойства готовой продукции» (исходный Word-документ).
2. Проверить таблицу «Справочник заказчиков». Поле *Код заказчика* должно быть назначено ключевым.
3. Проверить таблицу «Заказы» для отображения оперативной информации по принятым к исполнению заказам. Поля *Код продукции* и *Код заказчика* должны подставляться из соответствующих справочников с помощью столбцов подстановки. Если этого нет, следует воспользоваться мастером подстановки для формирования соответствующих столбцов. Столбец должен содержать два поля: код и наименование. Мастер подстановки вызывается из конструктора таблицы путем выбора из раскрывающегося списка в поле «Тип данных».
4. Разработать запрос «Отгрузка» для расчета стоимости каждого заказа (закладка «Запрос»). Запрос создается в режиме конструктора на основе трех исходных таблиц. Поля из таблиц-источников, образующих схему данных запроса и размещающихся в верхней панели окна конструктора запроса, помещаются в бланк запроса путем перетаскивания мышкой. Расчетные величины вычисляются на основе полей таблиц с помощью арифметических операторов. Например, в таблицах содержатся поля с именами *Цена* и *Количество*. Тогда переменная *Стоимость* вычисляется с помощью следующего оператора:

*Стоимость: Цена*Количество*

Данный оператор записывается в первой строке следующего свободного столбца конструктора запроса.

После того, как структура запроса «Отгрузка» будет сформирована, проверить действие механизма отбора данных по условию. Для этого в конструкторе запроса ввести конкретные данные в ячейку, стоящую на пересечении столбца, по которому происходит отбор данных, и строки *Условие отбора*. Например, для выбора всей информации по заказу №5 следует ввести значение «5» в ячейку, стоящую на пересечении столбца *Номер заказа* и строки *Условие отбора*.

Для реализации нечеткого отбора следует воспользоваться маской поиска, содержащей символ *. Например, для выбора всех заказчиков - юридических лиц, имеющих организационную форму «ООО» (общество с ограниченной ответственностью), следует ввести в ячейку, стоящую на пересечении столбца *Наименование заказчика* и строки *Условие отбора*, следующее выражение:

ООО*

5. Оформить документ на отгрузку заказа (отчет «Накладная») (рис. 6). Разработка документа для печати выполняется с помощью мастера отчетов (закладка «Отчет»). Мастер отчетов создает структуру документа, основываясь на источнике данных. В качестве источника данных для отчета «Накладная» необходимо выбрать запрос «Отгрузка», все поля которого должны быть отражены в конструируемом документе для печати. Для получения в отчете итоговой строки «Итого» с суммарными данными по столбцам необходимо применить группировку данных по номеру заказа (шаг мастера «Добавить уровни группировки»). Доступ к формированию групповых (суммарных) данных осуществляется по кнопке «Итого».

Полученный с помощью мастера отчет следует отредактировать в режиме конструктора путем перетаскивания объектов отчета из области данных в область заголовка. Дата заказа, номер, заказчик, адрес доставки должны

попасть в область заголовка, в области таблицы остается только информация об отгруженной продукции.

6. Разработать структуры перекрестных запросов для представления сводной информации по выполненным заказам за отчетный период («Заказчик - Стоимость», «Продукция - Количество»). Перекрестные запросы создаются с помощью мастера (выбор опции Перекрестный запрос меню Создать), результаты перекрестных запросов представлены на рис. 4.
7. Создать запросы с ценами по категориям продукции (например, «Прайс_двигатели») в режиме конструктора. Источником для данных запросов служит таблица «Справочник продукции». Для отбора данных по категориям продукции используется поле *Код продукции*. В строке *Условие отбора* конструктора запроса следует установить нечеткое условие отбора по полю *Код продукции*, используя символ *. Например, для выбора всех двигателей типа MFD из общей номенклатуры готовой продукции необходимо ввести условие отбора:

MFD*

Каждый из запросов «Прайс» следует преобразовать в формат html. Для этого необходимо выбрать из меню Файл опцию Экспорт и сохранить файл под уникальным именем в каталоге, предусмотренном для хранения файлов сайта, установив тип файла «HTML Документ».

Структура перекрестного запроса «Заказчик-Стоимость»

	Итоговое значение	Дата 1	Дата 2	...
<i>Заказчик 1</i>	Сумма поля <i>Итого</i>	<i>Итого</i>	<i>Итого</i>	...
<i>Заказчик 2</i>	Сумма поля <i>Итого</i>		<i>Итого</i>	...
<i>Заказчик 3</i>	Сумма поля <i>Итого</i>	<i>Итого</i>		...
...

Структура перекрестного запроса «Продукция-Количество»

	Итоговое значение	Дата 1	Дата 2	...
<i>Продукция 1</i>	Сумма поля <i>Кол-во</i>	<i>Кол-во</i>	<i>Кол-во</i>	...
<i>Продукция 2</i>	Сумма поля <i>Кол-во</i>		<i>Кол-во</i>	...
<i>Продукция 3</i>	Сумма поля <i>Кол-во</i>	Кол-во		...
...

Рис. 4 Результаты перекрестных запросов «Заказчик-Стоимость» и «Продукция-Количество»

4.2 Разработка экранных форм приложения

Разработать прикладное приложение, состоящее из пяти форм (рис. 7):

1. Исходная форма создается в режиме конструктора, содержит логотип предприятия, название приложения «Система оперативного учета заказов» и три кнопки выбора режима входа в систему: «Оператор», «Аналитик», «Выход». По кнопке «Оператор» вызывается экранная форма «Ввод заказа». По кнопке «Аналитик» вызывается экранная форма «Результаты продаж». По кнопке «Выход» приложение закрывается. Установка кнопок на форму производится после того, как вызываемые по ним формы будут созданы.
2. Форма «Ввод заказа» создается с помощью мастера. Источником для нее является таблица «Заказы». Созданная мастером форма редактируется в режиме конструктора. При этом с помощью панели элементов на форму помещаются два элемента «Подчиненная форма», источниками данных для которых служат таблицы «Справочник продукции» и «Справочник заказчиков» соответственно. С помощью панели элементов на форму помещаются две кнопки: кнопка «Накладная», по которой можно

просмотреть на экране отчет «Накладная», и кнопка «Выход» для закрытия данной формы и возврата в исходную форму.

3. Форма «Результаты продаж» создается с помощью мастера. Источником для нее является запрос «Отгрузка». Созданная мастером форма редактируется в режиме конструктора. С помощью панели элементов на форму помещаются 5 кнопок:

- кнопка вызова экранной формы с диаграммой «Заказчик – Стоимость»
- кнопка вызова экранной формы с диаграммой «Продукция – Количество»
- кнопка для просмотра отчета «Заказчик – Стоимость»
- кнопка для просмотра отчета «Продукция – Количество»
- кнопка «Выход» для закрытия данной формы возврата в исходную форму

4. Форма с диаграммой «Заказчик – Стоимость» создается с помощью мастера (выбор опции **Диаграмма** меню **Создать**). Источником для диаграммы является перекрестный запрос «Заказчик – Стоимость». Диаграмма строится по двум полям перекрестного запроса – *Наименование заказчика* и *Итоговое значение Итого*. Созданная мастером форма редактируется в режиме конструктора. При этом с помощью панели элементов на форму помещается кнопка «Выход» для закрытия данной формы возврата в форму «Результаты продаж». Форма с диаграммой «Продукция – Количество» создается аналогично.

4.3 Разработка сайта предприятия

Исходными данными для разработки сайта являются графические и текстовые файлы, содержащие:

- логотип предприятия;
- графический иллюстративный материал по видам готовой продукции;
- текстовая информация о предприятии и его продукции;

- графический материал, предназначенный для оформления страниц сайта (элементы меню).

Логическая структура сайта представлена на рис. 8. Сайт состоит из страниц, связанных друг с другом гиперссылками. На каждой странице располагается логотип предприятия, текст, элементы меню с гиперссылками на другие страницы. Страницы имеют общий фоновый цвет, отличный от белого, и шрифт минимум двух цветов.

Средой для разработки сайта является стандартное приложение ОС Windows текстовый редактор Блокнот. Каждая страница сайта представляет собой файл в формате .html.

1. Исходная (домашняя) страница сайта представлена файлом «Исходная.html» и содержит помимо логотипа основное меню сайта – опции: «Предприятие», «Продукция», «Заказы».

Каждая опция представлена рисунком (графическим файлом в формате .jpg), с которым связана гиперссылка. Аналогом опций меню на сайте являются кнопки на экранной форме, позволяющие осуществлять переход с одной экранной формы на другую.

Гиперссылка, связанная с опцией меню «Предприятие», обеспечивает переход на страницу «Предприятие» (файл «Предприятие.html»).

Гиперссылка, связанная с опцией меню «Продукция», обеспечивает переход на страницу «Продукция» (файл «Продукция.html»).

Гиперссылка, связанная с опцией меню «Заказы», обеспечивает переход на страницу «Заказы» (файл «Заказы.html»).

2. Страница «Предприятие» (файл «Предприятие.html») содержит, помимо логотипа, общую информацию о специфике предприятия в виде отформатированного средствами языка HTML текста. Текст должен состоять минимум из трех абзацев под общим заголовком, содержать нумерованные и маркированные списки, разные гарнитуры шрифта.

Гиперссылка, связанная с рисунком «Назад» (файл «Back.jpg»), позволяет вернуться на страницу «Исходная».

3. Страница «Продукция» (файл «Продукция.html») содержит опции меню, позволяющие выбрать категории продукции. Опции представляют собой рисунки (файлы в формате .jpg) со связанными с ними гиперссылками:
 - при выборе опции «Двигатели» происходит переход на страницу «Двигатели» (файл «Двигатели.html»);
 - при выборе опции «Муфты» происходит переход на страницу «Муфты» (файл «Муфты.html»);
 - при выборе опции «Коробки передач» происходит переход на страницу «Коробки передач» (файл «Коробки.html»).

Гиперссылка, связанная с рисунком «Назад» (файл «Back.jpg»), позволяет вернуться на страницу «Исходная».

4. Страница «Двигатели» (файл «Двигатели.html») содержит, помимо логотипа, таблицу с изображениями двигателей различных марок и текстом, представляющим основные характеристики каждого образца. Текст следует брать из файла «Свойства готовой продукции.doc». Гиперссылка, связанная с надписью «Прайс», обеспечивает переход на страницу «Цены на двигатели» (файл «Прайс_Двигатели.html»). Гиперссылка, связанная с рисунком «Назад» (файл «Back.jpg»), позволяет вернуться на страницу «Продукция».

5. Аналогичную структуру имеют страницы «Муфты» и «Коробки передач». Каждой категории продукции соответствует своя страница с таблицей цен.

6. Страница «Заказы» (файл «Заказы.html») предназначена для регистрации заказчика (указания его реквизитов) и составления заявки на продукцию. Основными элементами страницы являются два бланка как элемента языка HTML. Первый бланк предназначен для регистрации, второй – для ввода заказа. Гиперссылка, связанная с рисунком «Назад» (файл «Back.jpg»), позволяет вернуться на страницу «Исходная».

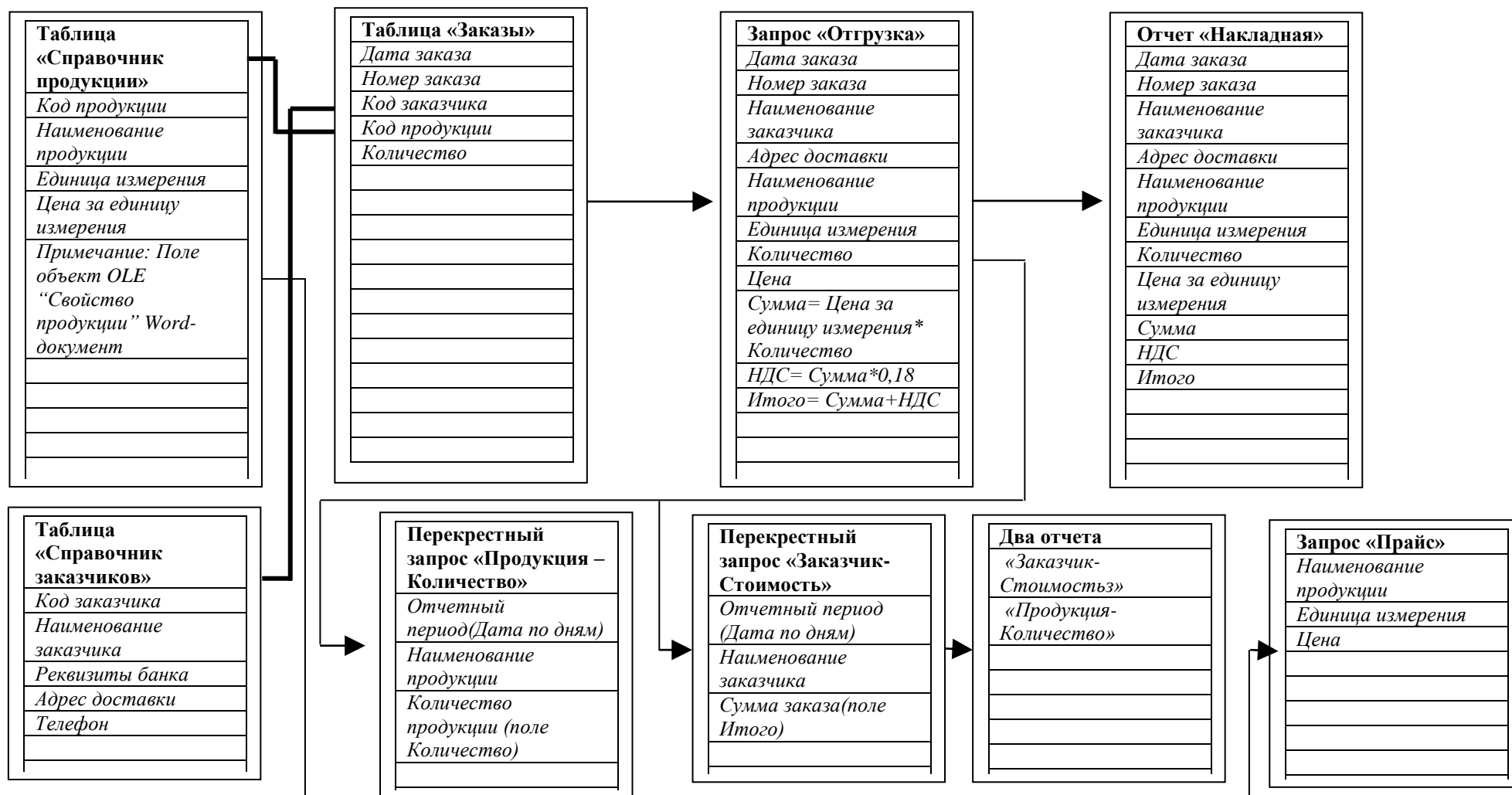


Рис. 5 Структура элементов данных по оперативному учету заказов, принятых к исполнению

Поставщик: ОАО «ЛЭМЗ»
 198096, г. Санкт-Петербург, ул. Корабельная, д. 6
 ИНН 7806162972, ОКПО 52185612
 ЗАО «Балтийский Банк»
 р/с 40702810400005920632,
 к/с 30101810100000000875,
 БИК 044030875

НАКЛАДНАЯ № _____

от «___» _____ 200___ г.

Кому: ООО «Турбина»
 Адрес: г. Санкт-Петербург, Косая линия, д. 5

№ п/п	Наименование	Цена, руб.	Ед. изм.	Кол-во	Стоимость руб.	НДС (18%), руб.	Итого, руб.
1	Вентилятор СК315С	13289.17	Шт.	1	13289.17	2657.83	15947.00
2	Вентилятор СК315В	11533.33	Шт.	1	11533.33	2306.67	13840.00
3	Глушитель LDC315-900	7276.67	Шт.	4	29106.67	5821.33	34928.00
4	Фильтр FFR315	4075.00	Шт.	1	4075.00	815.00	4890.00
5	Фильтрующий элемент EU5	2166.67	Шт.	1	2166.67	433.33	2600.00
6	Нагреватель канальный	14908.33	Шт.	1	14908.33	2981.67	17890.00
7	Регулятор скорости вентилятора	2182.50	Шт.	1	2182.50	436.50	2619.00
8	Регулятор нагрева ТТС	13133.33	Шт.	1	13133.33	2626.67	15760.00
9	Канальный датчик TG-K330	2361.17	Шт.	1	2361.17	472.23	2833.40
10	Обратный клапан RSK315	1452.50	Шт.	2	2905.00	581.00	3486.00
11	Наружная решетка BRYJ-2-315-0	2201.67	Шт.	2	4403.33	880.67	5284.00
12	Хомут МК 315	466.67	Шт.	4	1866.67	373.33	2240.00
13	Приточный клапан VST160	610.00	Шт.	7	4270.00	854.00	5124.00
14	Вытяжной клапан VEF160	610.00	Шт.	7	4270.00	854.00	5124.00
15	Воздуховод d.315	312.50	П.м.	90	28125.00	5625.00	33750.00
16	Отвод 90 град. d.315	469.17	Шт.	2	938.33	187.67	1126.00
17	Ниппель d.315	56.67	Шт.	30	1700.00	340.00	2040.00
18	Врезка воротниковая d.315/d.160	225.00	Шт.	14	3150.00	630.00	3780.00
	ИТОГО				144384.50	28876.90	173261.40

Итого отпущено товара на сумму: 173261-40 (Сто семьдесят три тысячи двести шестьдесят один рубль 40 копеек)

В том числе НДС 18%: 28876-90 (Двадцать восемь тысяч восемьсот семьдесят шесть рублей 90 копеек)

Отпустил _____
 (подпись)

 (расшифровка подписи)

МП

Принял _____
 (подпись)

 (расшифровка подписи)

Рис. 6 Пример формы первичного учетного документа «Накладная»

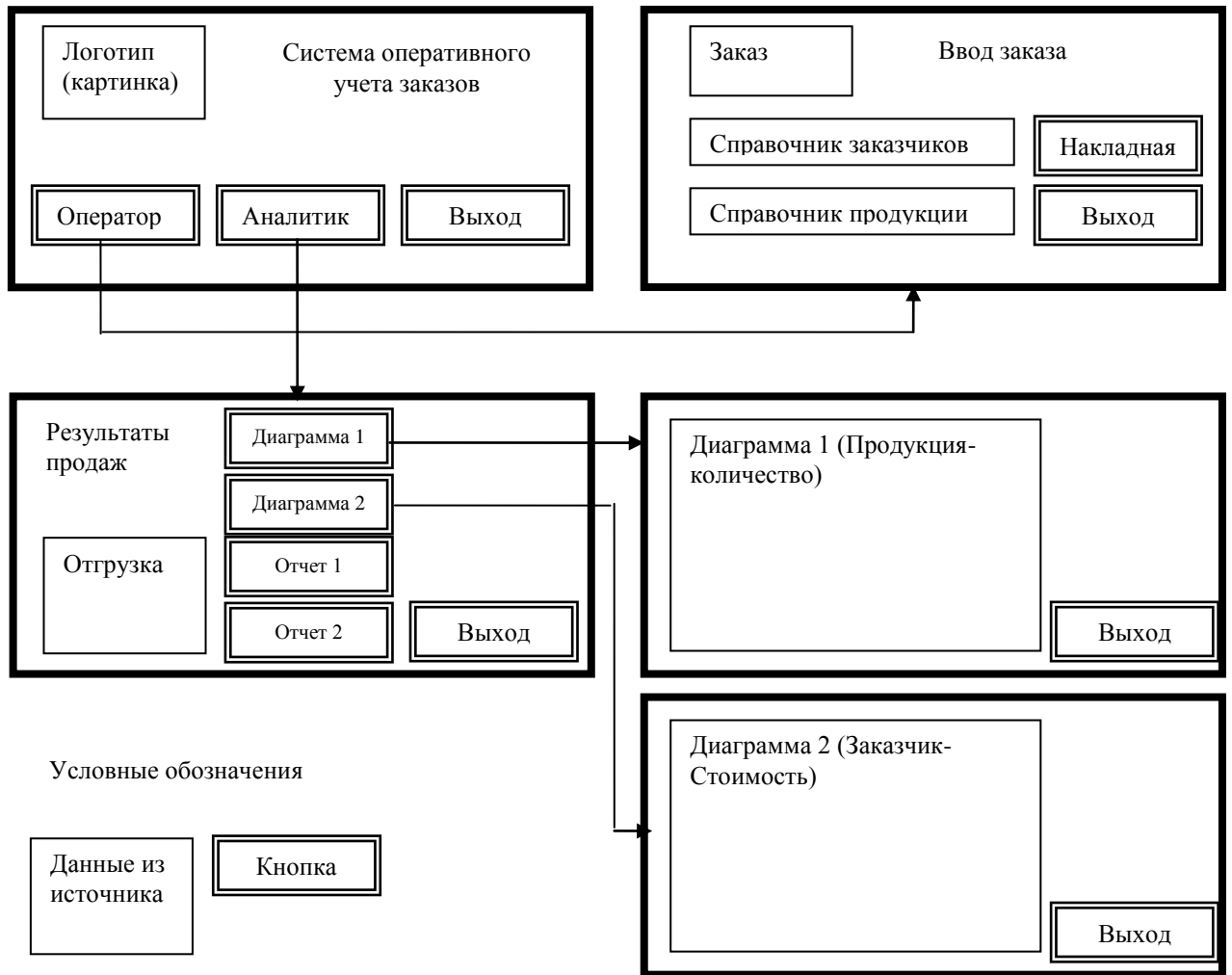


Рис. 7 Экранные формы приложения «Система оперативного учета заказов»

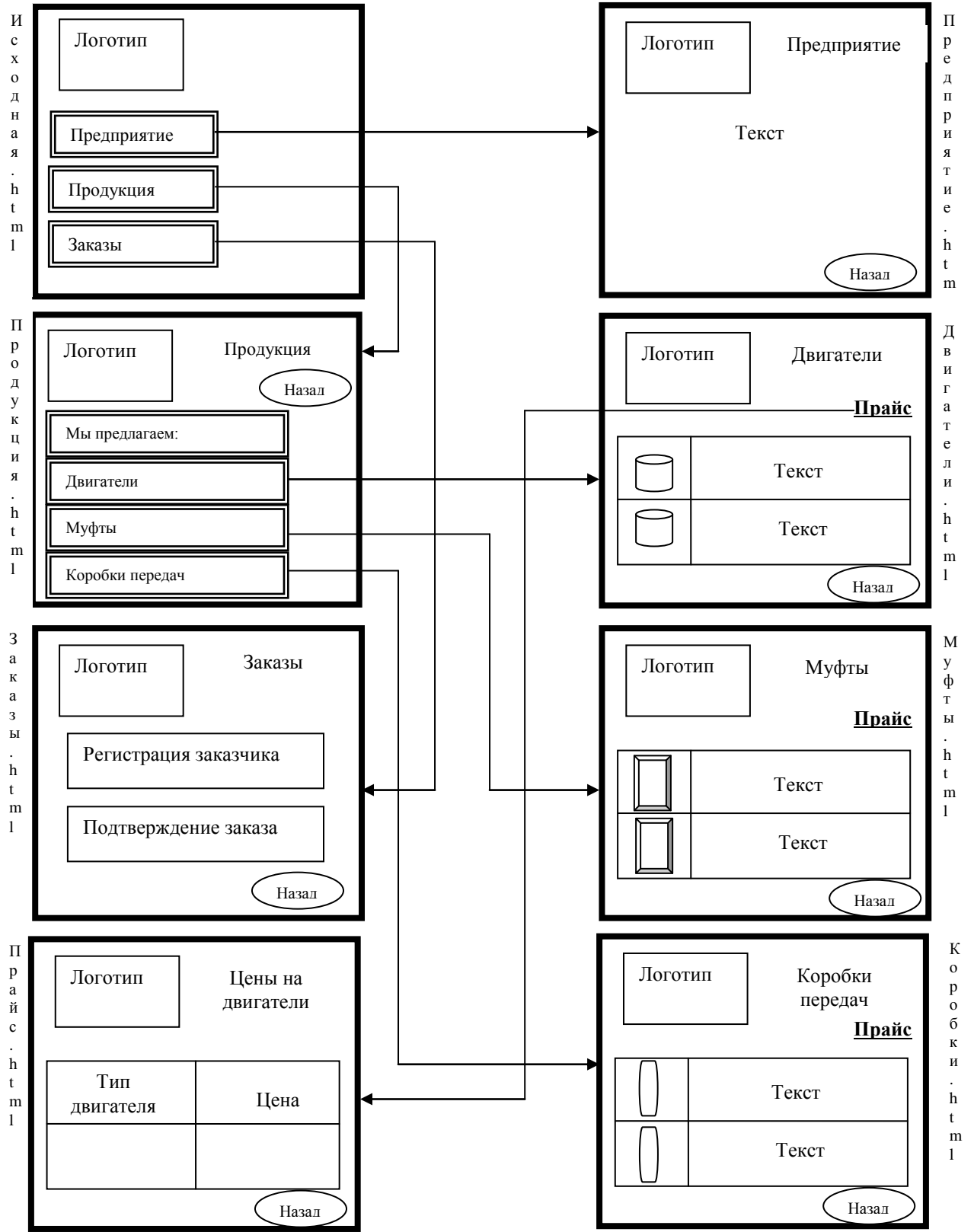


Рис. 8 Сайт «Электронный магазин»

ПРИЛОЖЕНИЕ 1

СТРУКТУРА БАЗЫ ДАННЫХ ACCESS. РАБОТА С ТАБЛИЦАМИ

Access является уникальной системой разработки баз данных для персональных компьютеров. База данных — это совокупность структурированных и взаимосвязанных данных и методов, обеспечивающих добавление, изменение, выборку и отображение данных. Единственный файл в формате базы данных Access .mdb может содержать не только объекты данных (таблицы, индексы и запросы), но и объекты приложения (формы, отчеты, макросы и модули, написанные на языке Access VBA), что делает процесс создания приложения более простым и эффективным. База данных Access может содержать до 32768 объектов (комбинаций таблиц, форм, отчетов и т.д.), причем одновременно могут быть открыты 1024 таблицы (при наличии достаточного количества системных ресурсов).

Элементы базы данных ACCESS

В состав любой базы данных Access входят следующие элементы:

Таблицы. В базе данных информация хранится в виде двумерных таблиц. Помимо создания таблиц вручную можно импортировать таблицы из других систем управления базами данных, баз данных типа клиент/сервер (таких как Microsoft SQL Server) и систем управления электронными таблицами (Microsoft Excel). Базу данных Access можно связать с таблицами баз данных других приложений, файлами электронных таблиц, форматированными текстовыми файлами и таблицами, хранящимися в другой базе данных Access.

Запросы. При помощи запросов можно произвести выборку данных, соответствующих некоторому критерию. Выборка данных может производиться одновременно из 16 таблиц. При создании запроса можно указать таблицы, в которых будет производиться выборка данных, указать те поля таблиц, которые должны быть включены в результат запроса, и указать условие отбора данных. В запрос можно включить до 255 полей (столбцов).

Условие отбора определяет то, какие данные будут выбраны из таблицы в результате запроса.

Формы. Формы позволяют отображать данные, содержащиеся в таблицах или запросах в более удобном для восприятия виде. При помощи форм можно добавлять в таблицы новые данные, изменять и удалять существующие. Форма может содержать рисунки, графики и другие внедренные объекты.

Отчеты. Отчеты предназначены для печати данных, содержащихся в таблицах и запросах, в красиво оформленном виде. Access позволяет использовать различные форматы отчетов и включать в них иллюстрации. Возможности отчетов в Access намного превосходят возможности отчетов других систем управления базами данных. Отчеты, как и формы, позволяют включать процедуры обработки событий, написанные на языке Access VBA.

Модули. Модули содержат Access VBA код, используемый для написания процедур обработки событий, таких как нажатие на кнопку в форме или отчете, для создания функций настройки, используемых в формах, отчетах и запросах, для автоматического выполнения операций над объектами базы данных и программного управления операциями, которыми нельзя управлять с помощью макросов. Другими словами, добавление VBA кода дает возможность создавать полные базы данных с настраиваемыми меню, панелями инструментов и многими другими возможностями.

Свойства таблиц Access

Информация в базе данных представляется в виде двумерной таблицы, строки которой называются *записями*, а столбцы *полями*. Термины *запись* и *поле* используются при обсуждении таблиц, а термины *строка* и *столбец* — при обсуждении результатов выполнения запросов.

Свойства таблицы Access определяют параметры таблицы в целом. Заполнение свойств не является обязательным. Чтобы открыть окно свойств таблицы в режиме конструктора, следует выбрать кнопку «Свойства» (Properties) на панели инструментов. Каждая таблица имеет 5 свойств:

Описание (Description). Определяет необязательный комментарий, содержащий описание таблицы в окне базы данных. Чтобы увидеть описание таблицы, активизируйте ее и выберите команду «Вид, Таблица» (View, Details) или нажмите кнопку «Таблица» (Details) на панели инструментов. Это описание является полезным для словаря данных, который используется для баз данных документов и приложений.

Условие на значение (Validation Rule). Определяет требования к данным, вводимым в поля записи. Используется для обеспечения целостности и непротиворечивости данных. Это свойство применяется ко всем полям таблицы.

Сообщение об ошибке (Validation Text). Позволяет указать текст сообщения, выводимого на экран, если введенные данные нарушают условие, определенное в свойстве «Условие на значение» (Validation Rule).

Фильтр (Filter). Определяет подмножество записей, выводимых после применения фильтра к таблице.

Порядок сортировки (Order By). Определяет порядок сортировки записей в таблице.

В бланке таблицы можно установить следующие свойства полей таблицы:

Имя поля (Field Name). Определяет имя поля в таблице. Имя поля должно удовлетворять соглашениям об именах объектов (набор правил, ограничивающих допустимые имена объектов Access). Имена объектов должны содержать не более 64 символов и могут включать любые комбинации букв, цифр, пробелов и специальных символов за исключением точки «.», восклицательного знака «!», квадратных скобок «[» и «]». Отметим, что имя не должно начинаться с пробела и содержать управляющие символы. Каждое поле таблицы должно иметь уникальное имя, но в различных таблицах можно использовать одинаковые имена полей. Хорошим тоном считается отказ от включения в имя поля пробелов и замена их на знаки подчеркивания «_». Имя

поля не должно быть очень длинным, особенно, если будут создаваться макросы или модули, использующие это поле.

Тип данных (Data Type). Определяет тип данных, выбираемый из раскрывающегося списка. В каждое поле допускается ввод данных только одного типа. В таблицах Access можно использовать следующие типы данных:

1. «Текстовый» (Text),
2. «Поле МЕМО» (Memo),
3. «Числовой» (Number),
4. «Дата/Время» (Date/Time),
5. «Денежный» (Currency),
6. «Счетчик» (AutoNumber),
7. «Логический» (Yes/No),
8. «Поле объекта OLE»(OLE Object),
9. «Гиперсвязь» (Hyperlink),
- 10.«Мастер подстановок» (Lookup Wizard).

Описание (Description). Определяет текст, содержащий описание поля таблицы. Этот текст выводится в строке состояния при добавлении или изменении данных в поле.

Ключевое поле (Primary Key). Чтобы сделать поле ключевым, выделите его и нажмите кнопку «Ключевое поле» (Primary Key) на панели инструментов.

Остальные свойства поля зависят от его типа данных и являются необязательными. Свойства поля, не включенные в бланк таблицы, заполняются в нижней части окна конструктора таблицы. Значение некоторых свойств выбирается из раскрывающихся списков, значения других свойств вводятся с клавиатуры. Вкладка «Общие» (General) содержит основные свойства поля. Во вкладке «Подстановка» (Lookup) можно задать поведение поля в форме.

В следующем списке приведены свойства полей таблицы, устанавливаемые на вкладке «Общие» (General):

Размер поля (Field Size). Определяет максимальный размер данных, которые могут сохраняться в полях с типом данных «Текстовый» (Text) или

«Числовой» (Number). Если свойство «Тип данных» (Data Type) имеет значение «Текстовый» (Text), то значением свойства «Размер поля» (Field Size) должно быть целое число в диапазоне от 0 до 255. По умолчанию задается размер 50. Если свойство «Тип данных» (Data Type) имеет значение «Числовой» (Number), то размер поля выбирается в раскрывающемся списке. Свойство «Размер поля» (Field Size) не задается для полей с типами данных «Дата/Время» (Date/Time), «Логический» (Yes/No), «Денежный» (Currency), «Поле МЕМО» (Memo), «Поле объекта OLE» (OLE Object) и «Гиперсвязь» (Hyperlink).

Формат поля (Format). Позволяет указать форматы вывода текста, чисел и дат на экран и на печать. Можно оставить формат, выбранный автоматически (исключая тип «Текстовый» (Text)), выбрать формат в раскрывающемся списке или ввести пользовательский формат. В любом случае формат поля не оказывает влияние на его значение, но влияет на отображение значения поля. Это свойство не задается для полей объектов OLE.

Число десятичных знаков (Decimal Places). Определяет число десятичных знаков, используемое при отображении чисел. Это свойство задается только для полей с типом данных «Числовой» (Number) и «Денежный» (Currency). Как и свойство «Формат поля» (Format), свойство «Число десятичных знаков» (Decimal Places) влияет на формат вывода значения поля, но не оказывает влияния на значение поля.

Маска ввода. Задает строку символов, облегчающую ввод данных в поле. Например, удобно создать следующую маску ввода для поля «Телефон», позволяющую вводить только цифры и автоматически добавляющую промежуточные символы:

+_(____) ____-__-__.

Для этого необходимо в поле *Маска ввода* записать формулу:

+9(999) 999-99-99,

где 9 – код, допускающий только цифровые символы.

Подпись (Caption). Указывает текст, который выводится в подписях, присоединенных к элементам управления, создаваемых путем переноса с

помощью мыши из списка полей, и который выводится для таблицы или запроса в заголовке столбца в режиме таблицы. Длина текста подписи может достигать 2048 символов, причем, в отличие от имени поля, можно использовать любые знаки пунктуации (кроме *).

Значение по умолчанию (Default Value). Позволяет указать значение, автоматически появляющееся в поле при создании новой записи. Например, в поле с типом данных «Дата/Время» (Date/Time) можно вводить по умолчанию текущую дату. Это свойство не задается для полей с типом данных «Счетчик» и полей объектов OLE (OLE object).

Условие на значение. Определяет требования к вводимым данным. Значение этого свойства представляет собой выражение, в зависимости от значения которого может быть выдано сообщение об ошибке. Это свойство не задается для полей с типом данных «Счетчик», «Поле МЕМО» (Memo) и полей объектов OLE (OLE object).

Сообщение об ошибке. Позволяет указать текст сообщения, выводимого на экран, если введенные данные нарушают условие, определенное в свойстве «Условие на значение».

Обязательное поле. Указывает, требует ли поле обязательного ввода значения. Если это свойство имеет значение «Да», то при вводе новой записи необходимо ввести значение в это поле или в любой присоединенный к нему элемент управления. Пустые значения в этом поле не допускаются. Свойство «Обязательное поле» ключевого поля должно иметь значение «Да» потому, что Access не допускает пустых значений в ключевых полях.

Пустые строки. Определяет, допускается ли ввод в данное поле пустых строк. Если свойства «Обязательное поле» и «Пустые строки» имеют значение «Да» (Yes), то при добавлении новой записи необходимо ввести в это поле хотя бы один символ. Пустая строка (« ») и значение Null — это не одно и то же. Это свойство можно определить только для полей с типом данных «Текстовый» (Text), «Поле МЕМО» (Memo) и «Гиперсвязь» (HyperLink).

Индексированное поле. Определяет индекс, создаваемый по одному полю. Это свойство может иметь значения: «Нет» (индекс не создается), «Да, допускаются совпадения» (в индексе допускаются повторяющиеся значения) и «Да, совпадения не допускаются» (повторяющиеся значения в индексе не допускаются) и не задается для полей с типом данных «Поле MEMO» (Memo) и полей объектов OLE (OLE object).

Новые значения. Определяет способ изменения значений поля счетчика при добавлении новых записей. Это свойство может принимать следующие значения: «Последовательные» — значение поля счетчика увеличивается на 1 в каждой новой записи и «Случайные» — поле счетчика в новой записи получает псевдослучайное значение типа «Длинное целое». Обычно значение «Случайные» используется в реплицируемых базах данных для присвоения записям таблиц, содержащихся в различных репликах, уникальных идентификаторов. Свойство «Новые значения» задается только для полей с типом данных «Счетчик».

Отношения между таблицами

Связь между таблицами устанавливает отношения между совпадающими значениями в ключевых полях — обычно между полями разных таблиц. В большинстве случаев связывают *ключевое поле* одной таблицы с соответствующим ему полем (часто имеющим то же имя), которое называют полем *внешнего ключа* во второй таблице. Таблица, содержащая ключевое поле, называется *главной*, а таблица, содержащая внешний ключ — *связанной*.

Существует четыре вида отношений:

1. *Один-к-одному.* Запись в таблице «А» (главной таблице) может иметь не более одной связанной записи в таблице «В» (связанной таблице) и наоборот. Этот тип связи используют не очень часто, поскольку такие данные могут быть помещены в одну таблицу. Связь с отношением «один-к-одному» используют для разделения очень широких таблиц или для отделения части таблицы по соображениям защиты. В таблицах, связанных

отношением «один-к-одному», ключевые поля должны быть уникальными (в этих полях не допускаются повторяющиеся данные).

2. *Многие-к-одному*. Одной записи в таблице «А» может соответствовать одна запись в таблице «В», а одной записи в таблице «В» - несколько записей в таблице «А». В таблице «В» ключевое поле должно быть уникальным.
3. *Один-ко-многим*. Связь с отношением «один-ко-многим» является наиболее часто используемым типом связи между таблицами. Каждой записи в таблице «А» могут соответствовать несколько записей в таблице «В», а запись в таблице «В» не может иметь более одной соответствующей ей записи в таблице «А». В таблице «А» ключевое поле должно быть уникальным. Связь с отношением «один-ко-многим» таблиц «А» и «В» можно рассматривать как связь с отношением «многие-к-одному» таблиц «В» и «А».
4. *Многие ко многим*. Одной записи в таблице «А» могут соответствовать несколько записей в таблице «В», а одной записи в таблице «В» несколько записей в таблице «А». В этом случае не требуется уникальности ключевых полей таблиц «А» и «В». Этот вид отношения используется для связывания таблиц, имеющих только внешние ключи.

Обеспечение ссылочной целостности

Автоматическое обеспечение ссылочной целостности данных является важной особенностью Access. Если на связь между таблицами наложены условия ссылочной целостности, то Access не позволяет добавлять в связанную таблицу записи, для которых нет соответствующих записей в главной таблице, и изменять записи в главной таблице таким образом, что после этого в связанной таблице появятся записи, не имеющие главных записей; а также удалять записи в главной таблице, для которых имеются подчиненные записи в связанной таблице.

ЗАПРОСЫ К БАЗЕ ДАННЫХ ACCESS

Понятие запросов, виды и назначение запросов Access

Источником запроса могут быть:

- одна или несколько исходных таблиц базы данных,
- сохраненные таблицы, полученные в результате выполнения других запросов,
- другой запрос с использованием его временной таблицы с результатами.

Запрос позволяет:

- выбирать данные из одной или нескольких взаимосвязанных таблиц,
- производить вычисления,
- получать результат в виде таблицы,
- обновлять данные в таблицах,
- добавлять и удалять записи.

Структура запроса создается с помощью графического средства формирования запроса по образцу — QBE (Query By Example). Запрос QBE включает подсхему данных с используемыми таблицами и бланк запроса. При конструировании запроса достаточно, работая мышью, выделить и перетащить необходимые поля из источников, представленных в схеме данных запроса, в бланк запроса и ввести условия отбора записей.

Средствами запроса можно:

- выбрать записи, удовлетворяющие условиям отбора;
- включить в результирующую таблицу запроса нужные поля;
- произвести вычисления в каждой из полученных записей;
- сгруппировать записи с одинаковыми значениями в одном или нескольких полях и выполнить над ними групповые функции;
- произвести обновление полей в выбранном подмножестве записей;
- создать новую таблицу базы данных, используя данные из существующих таблиц;

- удалить выбранное подмножество записей из таблицы базы данных;
- добавить выбранное подмножество записей в другую таблицу.

Многотабличный запрос позволяет сформировать новую таблицу, записи которой образуются путем объединения взаимосвязанных записей из разных таблиц базы данных и включения нужных полей из этих таблиц.

Последовательное выполнение ряда запросов позволяет решать достаточно сложные задачи, не прибегая к программированию.

В Access может быть создано несколько видов запросов.

1. Запрос на выборку — выбирает данные из взаимосвязанных таблиц и других запросов. Результатом его является таблица, которая существует до закрытия запроса. Запрос на выборку играет особую роль, так как на его основе строятся запросы другого вида.
2. Запрос на создание таблицы — основан на запросе выборки, но, в отличие от него, результат запроса сохраняется в новой таблице.
3. Запросы на обновление, добавление, удаление — это запросы-действия, в результате выполнения которых изменяются данные в таблицах.

Основы конструирования запроса

Основные принципы конструирования запроса заложены в технике конструирования запроса на выборку. Запрос на выборку позволяет выбрать данные из одной или нескольких взаимосвязанных таблиц. Результаты выполнения запроса отображаются в виде таблицы, существующей до ее закрытия. Записи этой таблицы формируются на основе задаваемых в запросе условий отбора записей в исходных таблицах и связей между этими таблицами, имеющимися в схеме данных запроса. Поля, включаемые в записи результата, указываются пользователем в бланке запроса.

Таблица с результатами запроса может применяться при дальнейшей обработке данных. В запросе на выборку могут использоваться не только таблицы базы данных, но и ранее созданные запросы. При этом нет

необходимости сохранять таблицы, получаемые в результате выполнения ранее созданных запросов.

Результаты выполнения запроса выводятся в режиме таблицы. Окно запроса в режиме таблицы аналогично окну просмотра таблицы базы данных.

Разработка запроса производится в режиме Конструктора запросов.

Окно запроса

В основном окне базы данных в закладке «Запрос» выбрать опцию «Создание запроса в режиме конструктора». В окне «Добавление таблицы» выбрать используемые в запросе таблицы. Закрыть окно. В результате появится окно Конструктора запросов — <имя запроса>: запрос на выборку.

Окно Конструктора запросов разделено на две панели. Верхняя панель содержит схему данных запроса, которая включает выбранные для данного запроса источники данных (таблицы). Таблицы представлены списками полей. Схема данных отображает связи между таблицами. Кроме того, пользователь сам может установить нужные связи, перетаскивая задействованные связи поля из одного списка полей в другой.

Нижняя панель является бланком запроса по образцу, который нужно заполнить. Каждый столбец бланка относится к одному полю, с которым нужно работать в запросе. Поля могут использоваться для включения их в результат выполнения запроса, для задания сортировки по ним и для задания условия отбора записей.

При заполнении бланка запроса необходимо:

- в строку «Поле» включить имена полей, используемых в запросе,
- в строке «Вывод на экран» отметить поля, которые должны быть включены в результирующую таблицу,
- в строке «Условие отбора» задать условия отбора записей,
- в строке «Сортировка» выбрать порядок сортировки записей результата.

Каждый столбец бланка запроса соответствует одному из полей таблиц. Кроме того, здесь может размещаться вычисляемое поле, значение которого

вычисляется на основе значений других полей, или итоговое поле, использующее одну из встроенных групповых функций Access.

Для включения нужных полей из таблиц в соответствующие столбцы запроса достаточно:

- перетащить нужное поле из списка полей таблицы в схеме данных запроса в первую строку бланка запроса.
- установить нужные связи (объединения), перетащив задействованные в связи поля из одного списка полей в другой.

Условия отбора записей

Условия отбора записей могут задаваться для одного или нескольких полей в соответствующей строке бланка запроса. Условием отбора является выражение, которое состоит из операторов сравнения и операндов, используемых для сравнения. В качестве операндов выражения могут использоваться некоторые заданные значения и идентификаторы (ссылки). Идентификатор представляет собой ссылку на значение поля, элемента управления или свойства. Идентификаторами могут быть имена полей, таблиц, запросов, форм, отчетов и т. д. Если необходимо указать ссылку на поле в конкретной таблице, форме, отчете, то перед именем поля ставится имя таблицы, заключенное в квадратные скобки и отделенное от имени поля восклицательным знаком. Например, [Имя таблицы]![Имя поля].

Вычисляемые поля

В запросе над полями могут производиться вычисления. Результат вычисления образует вычисляемое поле в таблице, создаваемой по запросу. При каждом выполнении запроса производится вычисление с использованием текущих значений полей, арифметических выражений и встроенных функций Access.

Выражение вводится в бланк запроса в пустую ячейку строки «Поле». В этой ячейке формируется имя поля *Выражение N* , где N — целое число, увеличивающееся на единицу для каждого нового создаваемого вычисляемого

поля в запросе. Имя вычисляемого поля выводится перед выражением и отделяется от него двоеточием. Например:

Выражение1:[Цена][Количество],*

где *Цена* и *Количество* — имена полей.

Имя вычисляемого поля *Выражение1* становится заголовком столбца в таблице с результатами выполнения запроса. Это имя можно изменить. Пользователь имеет возможность переименовать вычисляемое поле. Для этого достаточно в бланке запроса вместо *ВыражениеN* ввести новое имя.

Параметры запроса

Конкретное значение поля в условии отбора может вводиться непосредственно в бланк запроса или задаваться пользователем при выполнении запроса в диалоговом окне. Чтобы выводилось диалоговое окно для ввода конкретного значения поля в условии отбора, нужно определить параметр запроса. Имя параметра запроса может вводиться непосредственно в условия отбора в квадратных скобках. При выполнении запроса это имя появится в диалоговом окне «Введите значение параметра». Если в запрос вводится несколько параметров, то порядок их ввода через диалоговые окна определяется порядком расположения полей с параметрами в бланке запроса.

Мастера создания запросов

Простейшие запросы некоторых видов могут быть созданы с помощью Мастеров Access. Мастер запросов ускоряет процесс создания запроса, автоматически выполняя первоначальные простейшие действия по подготовке запроса. Вызванный Мастер запросов запрашивает у пользователя сведения и создает запрос на основе ответов пользователя. При необходимости можно в режиме Конструктора отредактировать запрос.

Создание запроса с помощью Мастера начинается с выбора в окне базы данных вкладки «Запросы» опции «создание запроса с помощью Мастера». Следует выполнять инструкции, выводящиеся в окнах диалога Мастера.

С помощью Мастера можно создать:

– простой запрос на выборку;

- запрос для поиска повторяющихся записей;
- запрос для поиска записей, не имеющих подчиненных записей;
- перекрестный запрос.

Заметим, что запрос для поиска повторяющихся записей и запрос для поиска записей, не имеющих подчиненных, являются элементарными запросами на выборку.

ФОРМЫ И ОТЧЕТЫ В ACCESS

Принципы создания экранных форм

В последнее десятилетие в программировании для операционной системы Windows наметился так называемый *визуальный подход*. До этого серьезным препятствием для разработки графических приложений была сложность создания различных элементов управления и контроля их работы. Достаточно взглянуть на окно любой Windows-программы. В нем имеется множество стандартных элементов управления (кнопки, пункты меню, списки, переключатели и т. д.). Очень трудоемко вручную описывать процесс создания этих элементов в соответствии с требованиями Windows, на глаз определять координаты, отслеживать их состояние с помощью специальных команд. Например, для простой программы, складывающей два числа, потребуется один оператор (одна строка исходного текста) для выполнения нужного вычисления и сотни строк кода для подготовки приложения к работе в Windows, создания кнопки и пары полей ввода.

Этот процесс автоматизирован в *средах быстрого проектирования* (Rapid Application Development, RAD-среды). Все необходимые элементы оформления и управления создаются и обслуживаются с помощью готовых визуальных компонентов, которые с помощью мыши перетаскиваются в проектируемое окно. Их свойства и поведение затем настраиваются с помощью простых редакторов, визуально показывающих характеристики соответствующих элементов. При этом вспомогательный исходный текст программы, ответственный за создание и работу этих элементов, генерируется *RAD-средой* автоматически, что позволяет сосредоточиться только на логике решаемой задачи. В результате программирование во многом заменяется проектированием — подобный подход называется *визуальным программированием*.

Компоненты достаточно легко создавать самостоятельно, поэтому в мире сегодня распространяются тысячи бесплатных и платных компонентов для наиболее известных *RAD-сред*, из них формируются библиотеки компонентов

— *объектные репозитории*. Компоненты выступают в роли «строительных кирпичиков», позволяющих собирать готовое приложение с богатыми возможностями, написав всего десяток строк исходного кода, и такой *компонентный подход* к созданию программ считается очень перспективным, потому что без лишних усилий и на законных основаниях допускает повторное использование чужого труда.

Формы являются ключевыми элементами в приложениях Access для доступа к данным.

Типы элементов управления

В формах Access существует три типа элементов управления:

1. *Присоединенные элементы управления* связаны с полем источника данных для формы или подчиненной формы. Источником данных может быть таблица или запрос. Присоединенные элементы управления отображают и обновляют в ячейке данных текущей выделенной записи значение поля, с которым они связаны. Текстовые поля (Text Boxes) являются наиболее общими присоединенными элементами управления. С помощью присоединенного элемента управления OLE можно отображать на экране содержимое графического объекта или проигрывать звуковой аудио-файл. Можно связать переключатели, выключатели и флажки с логическими полями типа «Да/Нет» (Yes/No). Все присоединенные элементы управления имеют связанные с ними метки, отображающие свойство поля «Имя» (Caption). Эту метку можно изменить или удалить, не влияя на присоединенный элемент управления.
2. *Свободные элементы управления* отображают данные, не зависящие от источника данных формы или подчиненной формы. Свободные элементы управления OLE используются для добавления графики в форму. Линии и прямоугольники можно использовать для разделения формы на логически связанные части или для имитации на экране внешнего вида формы на бумаге. Свободные текстовые поля используются для ввода данных без обновления поля в источнике данных, но возможно использование их для

других целей, например, для установки значения, используемого в выражении. Некоторые свободные элементы управления, например текстовые поля, имеют связанные с ними метки, а некоторые, например свободные объекты OLE, не имеют.

3. *Вычисляемые элементы управления* в качестве своего источника данных используют выражения. Обычно в выражения, используемые вычисляемыми элементами управления, включается значение поля, но в них можно также использовать значения, созданные с помощью свободных текстовых полей.

Создание текстовых полей

Access использует четыре основных типа текстовых полей:

1. *Однострочные текстовые поля* обычно являются присоединенными к элементам управления формы или полям таблицы или запроса.
2. *Многострочные текстовые поля* обычно являются присоединенными к мемо-полям и включают вертикальную полосу прокрутки для просмотра текста, не поместившегося в пределах поля.
3. *Вычисляемые поля* получают значения из выражений, которые начинаются со знака равенства (=) и обычно являются однострочными текстовыми полями. Если в выражение для вычисляемого поля включить значение какого-либо поля, например, поля «ДатаПлан», то текстовое вычисляемое поле будет присоединенным к этому полю. В противном случае вычисляемые текстовые поля являются свободными. Значение вычисляемого текстового поля нельзя редактировать.
4. *Свободные текстовые поля*, не являющиеся вычисляемыми текстовыми полями, могут быть использованы, например, для таких вспомогательных значений как граничные даты в процедурах Access VBA.

Большая часть способов создания форм в Access применима и к отчетам. Ниже перечислены основные различия между отчетами и формами:

1. Отчеты предназначены только для печати и в отличие от форм не предназначены для вывода в окне. При просмотре документа с увеличением в окне предварительного просмотра помещается только часть отчета.

2. Отчеты не обеспечивают просмотра в режиме таблицы. Возможен только предварительный просмотр перед печатью и просмотр макета отчета в режиме конструктора.

Отчетам Access присущи многие черты форм, в том числе и следующие:

1. Мастера отчетов могут создавать отчеты разных типов, в том числе:
 - отчет в одну колонку,
 - групповой/итоговый
 - почтовые наклейки.
2. Можно произвольным образом изменять отчеты, созданные Мастером отчетов (Report Wizard) в режиме Конструктора.
3. *Разделы (Sections)* отчета включают заголовок и примечание, появляющиеся в начале и в конце отчета, а также верхний и нижний колонтитулы страниц. В примечании часто указываются итоговые значения. Разделы отчетов соответствуют разделам форм с теми же именами.
4. *Разделы групп* отчетов являются эквивалентами разделов групп в формах. К каждой группе можно добавлять заголовок, включающий заглавие, и примечание для вывода итоговых значений группы. Можно размещать статическую (несвязанную) графику в разделах заголовка и примечания, а связанную графику — в разделах групп.
5. *Элементы управления* добавляются к отчетам из панели элементов Access, а затем выбирается их положение и размер.
6. *Подчиненные отчеты* могут добавляться в отчеты так же, как добавляются элементы управления подчиненных форм в основных формах.

ПРИЛОЖЕНИЕ 2

ЯЗЫК РАЗМЕТКИ ТЕКСТА HTML

Характеристика HTML как языка программирования для Интернет

HTML является наиболее популярным языком разметки документов. В исходном виде файл HTML содержит инструкции форматирования (теги), чередующиеся с текстом документа. Главной особенностью разметки HTML является возможность вставки ссылок на внешние документы или на внутренние разделы того же документа.

Язык HTML отличается от других широко распространенных языков программирования тем, что он позволяет отображать содержимое документа на компьютерах с различными аппаратными платформами, операционными системами и графическими характеристиками. Достигается это тем, что файл с документом WWW представляет собой набор инструкций для программы-обозревателя (браузера), которая выводит содержимое документа, интерпретируя его построчно. При этом задача выбора точных размеров окна, размера и цвета шрифтов и т. п. полностью ложится на браузер, который решает их в соответствии со своими возможностями и с возможностями системы. Этим достигается большая гибкость в выборе технических средств вывода изображения документа, но теряется точность. На практике это означает, что изображения одного и того же документа на одном компьютере, выведенные при помощи различных браузеров, будут отличаться.

В соответствии со спецификациями языка HTML текстовые документы сохраняются в файлах, имеющих расширение .htm или .html. Однако документ, открытый на сервере, редко бывает полностью текстовым, чаще всего он содержит графические изображения, создающие фон под текстом, либо имеющие определенную смысловую нагрузку. В документе могут

присутствовать и звуковые файлы, и движущиеся изображения. Все эти составные части документа являются разными файлами, содержащими логически связанную между собой информацию и объединенными локальными ссылками, которые позволяют управлять просмотром графики, прослушиванием звука, перемещением по тексту, либо выполняют другие функции.

Наряду с неоспоримыми достоинствами язык HTML обладает рядом слабых сторон. По сути, HTML - это технология представления информации, описывающая то, как браузер должен скомпоновать текст и графику на странице. В результате справедливо утверждение: «То, что вы видите, это все, что вы получаете»:

- невозможно описать данные независимо от отображения этих данных, что является главной проблемой поиска нужной информации;
- клиент не имеет средств извлечения данных со страницы Web для дальнейшей обработки;
- на конкретной странице Web-клиент получает только одно представление конкретного множества данных.

Например, для получения данных, отсортированных иным образом, нежели тот, что представлен на текущей странице, браузер должен посылать новый запрос серверу. В свою очередь серверу придется заново отправлять полную страницу HTML со списком данных. Такое манипулирование данными ведет к значительному увеличению числа обращений к Web-серверам и затрудняет их дальнейшее масштабирование.

Другой недостаток языка HTML заключается в его «плоскости», т. е. невозможности предоставления информации об иерархии данных.

Непоследовательность языка HTML (неоднозначность синтаксиса) затрудняет разбор текста программным обеспечением браузеров. Примером такой непоследовательности может служить необязательное наличие закрывающих тегов для соответствующих открывающих. Для обозначения

тегов используются символы «<» и «>», внутри которых помещаются названия инструкций и их параметры.

Базовая структура страницы WEB

1. Для создания Web-страницы требуются текстовый редактор, например, стандартное приложение ОС Windows Блокнот (Пуск-> Программы-> Стандартные -> Блокнот). В блокноте создается новый файл и сохраняется с расширением .html.
2. Язык HTML — это набор меток с форматом <TAG>текст</TAG>.
3. Все Web-страницы начинаются с метки <HTML> и заканчиваются меткой </HTML>.
4. HTML-документ состоит из двух секций. Первая из них, голова, определяется метками <HEAD> и </HEAD>.
5. Вторая секция, тело страницы, обрамляется метками <BODY> и </BODY>.
6. Заглавие страницы указывается между метками <TITLE> и </TITLE> и должно находиться в секции головы.
7. Текст Web-страницы располагается в секции тела. Для превращения текста в абзацы используется метка <P>.

Стили форматирования

1. Для введения стилей текста предназначены четыре основные метки: — жирный, <I> — курсивный, <U> — подчеркнутый и <TT> — моноширинный. Допускается комбинировать эти метки, определяя для одного и того же текста несколько стилей.

Стиль текста	Начальная метка	Конечная метка
Жирный		
<i>Курсив</i>	<I>	</I>
<u>Подчеркивание</u>	<U>	</U>
Моноширинный	<TT>	</TT>

2. Для вывода нестандартных символов, например знака авторского права, применяются специальные коды (символьные ссылки или названия).

Знак	Символьная ссылка	Название
<	<	<
>	>	>
¢	¢	¢
£	£	£
¥	¥	¥
©	©	©
®	®	®
°	°	°
¼	¼	¼
½	½	½
¾	¾	¾
×	×	×

3. При наличии на странице нескольких разделов (глав) для форматирования их названий и определения структуры страницы применяются метки заголовков от <H1> до <H6>.
4. Для вывода белого пространства и возврата каретки в предформатированном тексте предназначены метки <PRE> и </PRE>.
5. Конец строки определяет метка
.
6. Чтобы провести поперек страницы горизонтальную линию, применяется метка <HR>.

Списки

1. Нумерованные списки удобны для ранжирования элементов и поэтапных процедур. Перед каждым элементом указывается метка , а весь список обрамляется метками и .
2. Маркированные списки позволяют выводить на странице Web бланочные комментарии. Перед каждым элементом списка указывается метка , а весь список находится внутри меток и .
3. Списки определений удобно применять для элементов типа словаря или секций документа (где каждый термин служит заголовком секции, а

определение — текстом секции). Для указания термина применяется метка <DT>, для определения — метка <DD>, а весь список обрамляется метками <DL> и </DL>.

4. Допускается вставлять список одного типа в список другого типа и даже того же самого типа.

Гипертекстовые связи

1. Адреса URLs можно разбить на четыре части: «как» (протокол, например http, с последующими символами ://), «кто» (имя хост-компьютера, например www.logophilia.com), «где» (каталог, например /Home/) и «что» (имя файла, например, homepage.html).

2. Базовая структура гипертекстовой связи имеет такой вид:

`Текст связи`

3. Если страница, к которой создается связь, находится в том же каталоге, что и текущий документ, в адресе URL связи можно оставить только имя файла.

4. Для создания якоря применяется такая разновидность метки <A>:

`Текст якоря`

5. Для образования связи к якорю применяется такая метка:

`Текст связи`

6. В связях электронной почты применяется разновидность mailto метки <A>:

`Текст связи`

Графика

1. Встроенные изображения появляются непосредственно на странице Web, а внешние изображения можно просмотреть только с применением отдельной графической программы.
2. Имеется множество графических форматов, но в Web применяются только два — GIF (самый распространенный) и JPEG.

3. Графику можно получить из библиотек клипартов, других страниц, архивов Интернет, с помощью сканера или факса.
4. Для образования графического файла в формат GIF из другого формата применяется программа LView.
5. Для добавления изображения на страницу Web применяется метка

``,

где *имя_файла* есть имя графического файла.

6. Чтобы превратить изображение в гипертекстовую связь, применяется метка `<A>`, в которой обычный текст связи заменяется меткой ``:

``

Дополнительные возможности форматирования

1. Метка `` определяет размер шрифта. Для установки размера всего текста предназначена метка `<BASEFONT=размер>`
2. Цвет текста определяет метка `<BODY TEXT= «#nnnnnn»>`. Кроме того, метка `<BODY>` поддерживает атрибут `BGCOLOR`, определяющий цвет фона.

Таблица цветов (в PhotoShop есть полная палитра цветов в кодах)

Формат	Цвет
#000000	Черный
#FFFFFF	Белый
#FF0000	Красный
#00FF00	Зеленый
#0000FF	Синий
#FF00FF	Вишневый
#00FFFF	Голубой
#FFFF00	Желтый

3. Способ нумерации в нумерованных списках задает метка `<OL TYPE=тип>`, а метка `<UL TYPE=тип>` определяет маркер для маркированных списков. Тип может принимать значения: `disc`, `circle`, `square`.
4. Графика страницы загружается быстрее, если в метке `` определить высоту `HEIGHT` и ширину `WIDTH`.

5. Для центрирования текста, абзацев, заголовков и графики применяется метка <CENTER>.
6. Форматирование линии производится в соответствии с таблицей:

Расширение <HR>	Действие
<HR WIDTH=x>	Определяет ширину линии в x пикселей.
<HR WIDTH=x%>	Определяет ширину линии в x процентов экрана.
<HR SIZE=n>	Определяет толщину линии в n единиц (принятая по умолчанию толщина соответствует одной единице).
<HR ALIGN=LEFT>	Выравнивает линию по левому полю.
<HR ALIGN=CENTER>	Центрирует линию
<HR ALIGN=RIGHT>	Выравнивает линию по правому полю.
<HR NOSHADE>	Выводит линию плоской (а не объемной на экране).

Создание таблиц

1. Таблица представляет собой прямоугольную сетку строк и столбцов.
2. Таблицы лучше структурируют текст, чем метки <PRE>, так как с таблицами проще работать, в них можно включать изображения и связи и допускается применять метки для форматирования текста, например вывода его жирным или курсивным шрифтом.
3. Текст и метки таблицы находятся внутри меток <TABLE> и </TABLE>. Чтобы вокруг таблицы выводилась рамка, применяется метка <TABLE BORDER>
4. Каждая строка определяется метками <TR> и </TR>, а для создания клеток применяются метки <TD> и </TD>.
5. Для превращения клетки в надпись (жирный центрированный текст) вместо меток <TD> и </TD> используются метки <TH> и </TH>.
6. Чтобы указать для таблицы содержательное название, применяется метка <CAPTION ALIGN=где>, в которой где может быть TOP или BOTTOM.
7. Метки <TD> и <TH> позволяют определить горизонтальное и вертикальное выравнивания. Для горизонтального выравнивания предназначена метка <TD ALIGN=выравнивание> или <TH ALIGN=выравнивание>, где выравнивание может быть LEFT, CENTER или RIGHT. Для вертикального выравнивания предназначена метка <TD

VALIGN=*вертикальное*> или <TH VALIGN=*вертикальное*>, где *вертикальное* может быть TOP, MIDDLE или BOTTOM.

8. Чтобы клетка перекрывала несколько столбцов, применяется метка <TD COLSPAN=*столбцы*>, где *столбцы* показывают число перекрываемых столбцов. Чтобы клетка перекрывала несколько строк, используется метка <TD ROWSPAN=*строки*>, где *строки* показывает число перекрываемых строк.
9. Расширения действуют с меткой <TABLE> и включают в себя атрибуты BORDER (для изменения размера рамки), WIDTH (для определения ширины таблицы), CELLSPACING (для изменения промежутка между каждой клеткой) и CELLPADDING (для изменения промежутка между содержимым клетки и ее рамкой).

Бланк

1. Бланк представляет собой страницу, заполненную разновидностями диалоговых боксов — текстовыми боксами, контрольными боксами и командными кнопками, которые применяются для получения информации от читателя.
2. Базовый бланк обрамлен метками <FORM> и </FORM>, причем в метке <FORM> имеются атрибуты ACTION и METHOD.
3. Чтобы пользователь мог послать данные бланка, применяется кнопка submit:

<INPUT TYPE=SUBMIT VALUE= «*Отметка*»>

4. Для очистки данных бланка и нового его заполнения применяется кнопка reset:

<INPUT TYPE=RESET VALUE= «*Отметка*»>

5. Ввод читателем простого текста производится в текстовые боксы:

<INPUT TYPE=TEXT NAME= «*Имя_поля*»>

6. Для ввода длинного или многострочного текста применяется текстовая область:

```
<TEXTAREA NAME=«Имя_поля» ROWS=Всего_строк COLS =
Всего_столбцов WRAP> </TEXTAREA>
```

7. Для ввода информации типа Да/Нет или Истина/Ложь предназначены контрольные боксы:

```
<INPUT TYPE=CHECKBOX NAME=«Имя_поля»> Отметка
```

8. Радиокнопки предоставляют читателю несколько опций:

```
<INPUT TYPE=RADIO NAME=«Имя_поля» VALUE=Значение»> Отметка
```

9. Для большого числа опций применяется список выбора:

```
<SELECT NAME=«Имя_поля» SIZE=Элементы>
<OPTION>Текст первого элемента</OPTION>
<OPTION>Текст второго элемента</OPTION>
<OPTION> и т.д. </OPTION>
</SELECT>
```

СПРАВОЧНИК ЯЗЫКА HTML

Скелет страницы Web

```
<HTML>
<HEAD>
<TITLE>Название окна страницы сайта</TITLE>
</HEAD>
<BODY>
    Все наполнение страницы сайта (тело)
</BODY>
</HTML>
```

Метки для абзацев и др.

```
Новый абзац      <P>
Горизонтальная линия <HR>
Конец строки     <BR>
Заголовки        <H1> (наибольший) <H6> (наименьший)
```

Форматирующие метки

```
Жирный          <B>Текст будет жирным</B>
Курсив           <I>Текст будет курсивным</I>
Подчеркивание   <U>Текст будет подчеркнутым</U>
Моноширинный    <TT>Текст будет похож на печать пишущей машинки</TT>
```

Форматированный `<PRE>Текст сохраняет все пробелы и табуляции</PRE>`

Маркированный список

``

`Первый элемент`

`Второй элемент`

и т.д.

``

Список определений

`<DL>`

`<DT>Первый термин`

`<DD>Первое определение`

`<DT>Второй термин`

`<DD>Второе определение и т.д.`

Нумерованный список

``

`Первый маркер`

`Второй маркер`

и т.д.

``

Метки связей

Внешняя связь	<code>Текст связи</code>
Якорь	<code>Текст якоря</code>
Внутренняя связь	<code>Текст связи</code>
Электронная почта	<code>Текст связи</code>
FTP (каталог)	<code>Текст связи</code>
FTP (файл)	<code>Текст связи</code>

Метки изображений

Базовое изображение	<code></code>
Изображение, выровненное с верхом текста	<code></code>
Изображение, выровненное с серединой текста	<code></code>
Изображение, выровненное с низом текста	<code></code>
Изображение с альтернативным текстом	<code></code>

Метки таблицы

`<TABLE BORDER>`

`<CAPTION ALIGN=TOP | BOTTOM>Текст надписи.</CAPTION>`

`<TR>`

`<TD>Первая строка, первый столбец</TD>`

`<TD>Первая строка, второй столбец</TD>`

`<TD>Первая строка, третий столбец</TD>`

`</TR>`

`<TR>`

`<TD>Вторая строка, первый столбец</TD>`

`<TD>Вторая строка, второй столбец</TD>`

`<TD>Вторая строка, третий столбец</TD>`

`</TR>`

и т.д.

`</TABLE>`

Заголовки

`<TR>`

`<TH>Заголовок первого столбца</TH>`

`<TH>Заголовок второго столбца</TH>`

`<TH>И так далее до конца</TH>`

`</TR>`

Горизонтальное выравнивание

<TD ALIGN=LEFT|CENTER|RIGHT>

<TH ALIGN=LEFT|CENTER|RIGHT>

Вертикальное выравнивание

<TD VALIGN=TOP|MIDDLE|BOTTOM>

<TH VALIGN=TOP|MIDDLE|BOTTOM>

Перекрывающиеся столбцы

<TD COLSPAN=Число_столбцов>

<TH COLSPAN=Число_столбцов>

Перекрывающиеся строки

<TD ROWSPAN=Число_строк>

<TH ROWSPAN=Число_строк>

Расширение HTML

Размер шрифта	Текст
Размер базового шрифта	<BASEFONT SIZE=размер>
Цвет текста	<BODY TEXT= «#nnnnnn»>
Стиль нумерованного списка	<OL TYPE=тип> тип м.б. <i>l, a, A, i, I</i>
Стиль маркированного списка	<UL TYPE=тип>, тип м.б. <i>disc, circle, square</i>
Размеры изображения	
Рисунок как гиперссылка	
Цвет фона	<BODY BGCOLOR= «#nnnnnn»> Цвет м.б. <i>yellow, green</i>
Фоновое изображение	<BODY BACKGROUND= «имя_файла»>
Размер рамки таблицы	<TABLE BORDER=размер>
Ширина таблицы	<TABLE WIDTH=ширина>
Ширина клетки таблицы	<TD WIDTH=ширина>
Интервал клетки таблицы	<TABLE SPACING =интервал>
Заполнение клетки таблицы	<TABLE CELLPADDING=заполнение>
Центрирование абзацев	<CENTER> Здесь центрируемые заголовки, текст, графика </CENTER>

Основные теги HTML

Тег	Назначение	Атрибуты	Значения атрибутов
все теги	-	ID	Уникальный идентификатор объекта, используется динамическими сценариями и в качестве селектора CSS
		NAME	Имя элемента. Используется при указании целевого кадра, создании закладок (якорей) и т.д.
		CLASS	Указывает принадлежность элемента к классу, используется в CSS
		TITLE	Всплывающая подсказка
<!-...—>	Комментарий -текст, который есть в документе, но не отображается при просмотре		
A	Гиперссылка	HREF	Целевой документ ссылки
B	Полужирное начертание	-	-
BASE	Базовые свойства страницы	TARGET	Кадр назначения по умолчанию
		HREF	Базовый адрес для всех относительных ссылок на странице
BODY	Секция тела документа,	LINK	Цвет гиперссылок

	содержит отображаемую часть страницы	ALINK	Цвет активных гиперссылок
		VLINK	Цвет посещенных гиперссылок
		TEXT	Цвет текста
		BACKGROUND	Фоновое изображение
		BGCOLOR	Цвет фона
BR	Перевод строки	-	-
DIV	Обобщенный тег блочной разметки	-	-
FONT	Параметры шрифта	COLOR	Цвет шрифта
		FACE	Гарнитура шрифта
		SIZE	Размер шрифта
FORM	Логическое объединение элементов	ACTION	Адрес сценария, обрабатывающего форму
		METHOD	Метод отправки данных на сервер
FRAME	Задание свойств кадра	NORESIZE	Запрещение изменения размера кадра при просмотре
		SCROLLING	Наличие полос прокрутки у кадра
		SRC	Страница, открываемая в кадре
FRAMESET	Задание набора кадров и его свойств	BORDER	Толщина бордюра
		BORDERCOLOR	Цвет бордюра
		COLS	Количество кадров по горизонтали
		FRAMEBORDER	Отображение бордюра у кадров
		FRAMES PACING	Расстояние между кадрами
		ROWS	Количество кадров по вертикали
Hn (n=1..6)	Заголовок уровня n	ALIGN	Выравнивание
HEAD	Секция заголовка - содержит служебную информацию документа	-	-
HTML	Содержит весь документ	-	-
I	Наклонное начертание	-	-
Тег	Назначение	Атрибуты	Значения атрибутов
IMG	Изображение	ALIGN	Выравнивание
		HEIGHT	Высота
		SRC	Адрес графического файла
		WIDTH	Ширина
INPUT	Поле ввода	TYPE	Тип элемента формы. Он определяет внешний вид и функции элемента, и в зависимости от типа имеются различные атрибуты
LI	Элемент списка	VALUE	Значение элемента списка
NOFRAMES	Содержимое без кадров	-	
OL	Упорядоченный список	START	Начальное значение списка
		TYPE	Тип нумерации: 1, a, A, i, I
P	Параграф	ALIGN	Выравнивание
PRE	Сохранение исходного форматирования	-	-
SPAN	Обобщенный тег строковой разметки	-	-
SUB	Верхний индекс	-	-

SUP	Нижний индекс	-	-
TABLE	Таблица	ALIGN	Выравнивание
		BACKGROUND	Фоновое изображение
		BGCOLOR	Цвет фона
		BORDER	Толщина бордюра
		BORDERCOLOR	Цвет бордюра
		CELLPADDING	Расстояние от границы ячейки до содержимого
		CELLSPACING	Расстояние между ячейками
		WIDTH	Ширина
TD	Ячейка таблицы	ALIGN	Выравнивание содержимого
		BACKGROUND	Фоновое изображение
		BGCOLOR	Цвет фона
		BORDERCOLOR	Цвет бордюра
		COLSPAN	Расширение на несколько столбцов
		HEIGHT	Высота
		NOWRAP	Запрещение сдвига текста
		ROWSPAN	Расширение на несколько строк
		VALIGN	Выравнивание содержимого по вертикали
		WIDTH	Ширина
TEXTAREA	Текстовый блок	ALIGN	Выравнивание
		COLS	Ширина (в символах)
		ROWS	Высота (в строках)
TITLE	Заголовок документа	-	-
TR	Строка таблицы	ALIGN	Выравнивание содержимого
		BGCOLOR	Цвет фона
		BORDERCOLOR	Цвет бордюра
		VALIGN	Выравнивание содержимого по вертикали
U	Подчеркнутое начертание	-	-
UL	Маркированный список	TYPE	Тип маркера списка: disc, circle, square.

Закрывающие части тегов HTML

Тег	Закрывающая часть	Тег	Закрывающая часть	Тег	Закрывающая часть
<i— —>	Запрещена	Hn	Обязательна	SPAN	Обязательна
A	Обязательна	HEAD	Обязательна	SUB	Обязательна
B	Обязательна	HTML	Обязательна	SUP	Обязательна
BASE	Обязательна	I	Обязательна	TABLE	Обязательна
BODY	Обязательна	IMG	Запрещена	TD	Обязательна
BR	Запрещена	INPUT	Запрещена	TEXTAREA	Обязательна
DIV	Обязательна	LI	Необязательна	TITLE	Обязательна
FONT	Обязательна	NOFRAMES	Обязательна	TR	Обязательна
FORM	Обязательна	OL	Обязательна	U	Обязательна
FRAME	Запрещена	P	Необязательна	UL	Обязательна
FRAMESET	Обязательна	PRE	Обязательна		

Данную таблицу нельзя считать полным справочником по HTML, но она содержит основные теги, с помощью которых можно создать полнофункциональную страницу HTML.

В случае, когда закрывающая часть тега необязательна, тег автоматически считается закрытым при встрече следующего открывающего тега (например, два параграфа подряд) или при встрече закрывающей части родительского тега (например, тег `` автоматически закрывает последний тег ``).

В случае, если закрывающая часть тега обязательна, ее отсутствие может привести как к некорректному отображению элементов (два подряд тега `<DIV>` будут восприниматься не как последовательные, а как вложенные), так и к потере содержимого (незакрытый тег `<TITLE>` может вызвать неотображение содержимого страницы, т.к. браузер будет считать, что это все еще заголовок документа).

Пример страницы HTML

```
<HTML>
<HEAD>
<TITLE>Пример Интернет-страницы</TITLE>
</HEAD>
<BODY BGCOLOR= «yellow»>
<H1>Это заголовок. Он отображается крупным жирным шрифтом. Фон страницы задан желтым цветом</H1>
<P> Это новый абзац
<IMG SRC= «имя_файла_с_ картинкой.jpg»> Так мы помещаем картинку на страницу
<P>
<A HREF= «файл_другой_страницы.html»>Кликнув сюда, мы перейдем на другую страницу</A>
<P>
</BODY>
</HTML>
```

ПРИЛОЖЕНИЕ 3

ПРИМЕР СПРАВОЧНИКА ГОТОВОЙ ПРОДУКЦИИ ПРЕДПРИЯТИЯ
МАШИНОСТРОЕНИЯ

Таблица. Электромеханические модули (ЭММ)

Код	Наименование	Ед. изм.	Цена	Технические характеристики ЭММ						
				Мощность	Напряжение	Частота вращения	Номинальный момент	Момент инерции	КПД	Масса
Д812	ЭММ	шт.	1200,00	55	220	550	955	3.125	89.5	1215
Д812	ЭММ	шт.	1270,00	75	220	515	1391	5.750	89.8	1570
Д814	ЭММ	шт.	1210,00	115	220	500	2197	10.000	91.2	2240
Д816	ЭММ	шт.	1290,00	150	220	480	2984	14.750	92.5	2860
Д818	ЭММ	шт.	1200,00	186	220	450	3947	26.750	91.4	3745
Mkr050	ЭММ	шт.	1650,00	225	220	450	4826	28.300	90.5	3640
Mkr061	ЭММ	шт.	1680,00	355	220	340	342	90.000	91.3	8260
Mkr062	ЭММ	шт.	1750,00	380	220	340	678	98.000	91.8	8100
MKS315H	ЭММ	комплект	1850,00	75	220	550	1316	3.800	89.9	1120
MKS355L	ЭММ	комплект	1850,00	110	220	500	2126	10.600	90.3	1910
MKS400M	ЭММ	комплект	1850,00	150	220	500	2896	14.500	92.1	2040
GMK400.1	ЭММ	комплект	2530,00	55	220	550	965	3.200	89.0	1100
GMK400.2	ЭММ	комплект	2530,00	125	220	515	2342	12.200	90.6	2180
MFD180L0	ЭММ	шт.	3160,00	75	220	550	1310	3.800	88.9	1180
MFD180M	ЭММ	шт.	3200,00	63	220	550	1110	3.300	89.2	1200
MFD355M	ЭММ	шт.	3510,00	110	220	500	2125	7.800	92.0	1705
MFD400L	ЭММ	шт.	3180,00	132	220	500	2548	13.500	91.1	2200
MFD450L	ЭММ	шт.	3360,00	200	220	435	4438	24.500	91.9	3610
MFD450M	ЭММ	шт.	3500,00	220	220	435	4881	27.200	91.8	3530
MFD450L.1	ЭММ	шт.	3760,00	280	220	360	7508	62.400	92.6	5830
GN315M	ЭММ	шт.	3970,00	90	220	515	1686	5.900	91.3	1650
GN355L	ЭММ	шт.	3900,00	110	220	515	2064	7.500	91.8	1720
V160L	ЭММ	комплект	4200,00	55	220	550	970	3.100	88.9	1130
V315L	ЭММ	комплект	4320,00	90	220	500	1738	6.200	90.3	1170
V355L0	ЭММ	комплект	4430,00	132	220	515	2474	12.800	90.7	2260
YEK030	ЭММ	комплект	4500,00	185	220	480	3720	18.900	91.2	2880
YEK032	ЭММ	комплект	4320,00	200	220	480	4022	19.600	92.3	3120
GRA	ЭММ	шт.	4670,00	185	220	480	3720	19.200	91.8	2840
GRA7142	ЭММ	шт.	4700,00	225	220	435	4992	31.800	91.4	3730
GRA7142.1	ЭММ	шт.	4850,00	365	220	360	9786	126.000	92.4	8070
GRA6340	ЭММ	шт.	4910,00	300	220	400	7239	78.000	93.0	5920
GRA6345	ЭММ	шт.	4950,00	250	220	360	6703	65.400	92.4	5860
GRC8040	ЭММ	шт.	4980,00	280	220	360	7508	68.500	92.3	6240
YD822	ЭММ	шт.	5200,00	280	220	360	7428	75.000	92.5	7030
YD624	ЭММ	шт.	5250,00	370	220	340	3389	144.000	91.3	8880
MT52	ЭММ	шт.	5980,00	375	220	340	1239	102.000	92.6	8025
GMD69	ЭММ	шт.	6100,00	300	220	360	8043	67.800	92.8	5890
GMD79	ЭММ	шт.	6230,00	370	220	360	9920	126.800	91.2	8340
GMD79	ЭММ	шт.	6500,00	370	220	340	1239	118.000	91.6	8120
GMg810	ЭММ	шт.	6720,00	52	230	550	903	3.300	88.3	1080
GMg812	ЭММ	шт.	6780,00	75	230	515	1391	5.250	90.6	1570
GMg814	ЭММ	шт.	6800,00	112	230	500	2139	7.750	90.2	1940
GMg816	ЭММ	шт.	6900,00	149	230	480	2964	13.500	90.6	2440
GMg818	ЭММ	шт.	7000,00	186	230	435	4083	21.000	90.8	3520

ПРИЛОЖЕНИЕ 4

ПРИМЕР СПРАВОЧНИКА ЗАКАЗЧИКОВ

Таблица. Справочник заказчиков готовой продукции предприятия

Код	Наименование	Адрес	Телефон	E-mail
1	ООО ВИНДОР	192238, СПб, Пражская ул, д.16	(812) 520 98 28	
2	ООО Реклам. Агент-во Норма	198216, СПб, Бульвар Новаторов	(812) 283 64 54	
3	ООО Сафит	93012, СПб, ул. Бабушкина, д. 135	(812) 763 05 45	
4	ООО Техноплац	189630, СПб, Колпино, ул. Пролетарская, д. 46	(812) 108 36 47	
5	ООО Издательство Атлант	196240, 5-ый Предпортовый пр. д. 1	(812) 567 85 31	
6	ООО Ловен АГ	193036, СПб, 5-я Советская, д. 17/12	(812) 127 00 91	171282@mail.com
7	ЗАО ИНФОРМЛАЙН	189630, СПб, Колпинский р-н, Госненский пер.,1	(812) 346 16 20	
8	ЗАО ХИЛТИ Дистрибьюшн Лтд.	189630, СПб, Колпинский р-н, пр. Ленина, 39	(812) 520 33 77	hilty@yahoo.com
9	ООО НеваРеактив	196242, 5-ый Предмостовой пр., д. 12	(812) 444 43 09	
10	ЗАО Бонниер Бизнес Пресс	196240, 5-ый Предпортовый пр. д., 10/2	(812) 156 05 50	
11	ЗАО Евразия	196236, Смольная ул., д. 102	(812) 103 84 88	
12	ООО Компания Строй-Инвест	193241, Средний пр., д. 17	(812) 305 74 12	
13	ЗАО Издательский дом ЭКСТРА-БАЛ	196260, Трамвайный пр., д. 15	(812) 100 02 03	
14	ООО ЛИСТ СПб	196233, Басков пер., д. 14	(812) 378 40 98	