

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ  
имени А.А. ЕЖЕВСКОГО**

Институт экономики, управления и прикладной информатики  
Кафедра информатики и математического моделирования

**Учебное пособие  
по информационно-коммуникационным  
ТЕХНОЛОГИЯМ**

для студентов направлений подготовки 13.03.01 Теплоэнергетика и  
теплотехника, 13.03.02 Электроэнергетика и электротехника, 35.03.06  
Агроинженерия

Молодежный 2020

УДК 004

Печатается по решению научно-методической комиссии  
ФГБОУ ВО Иркутского ГАУ. Протокол № 4 от 25.05.2020 г.

Полковская М.Н. Учебное пособие по информационно-коммуникационным технологиям для студентов энергетического факультета (направления подготовки 13.03.01 Теплоэнергетика и теплотехника, 13.03.02 Электроэнергетика и электротехника, 35.03.06 Агроинженерия ) / Т.С. Бузина, М.Н. Полковская // Молодежный: Изд-во Иркутского ГАУ, 2020. – 143 с.

Рецензент:

к.т.н., доцент Логибнов А.Ю.

д.т.н., профессор, профессор «Информационные системы и защита информации» ФГБОУ ВО ИрГУПС Краковский Ю.М.

В учебном пособии освещаются основные аспекты использования информационно-коммуникационных технологий при решении прикладных задач. В частности, рассмотрены вопросы создания базы данных в СУБД Microsoft Access. Кроме того, приведены теоретические основы и практические задания по созданию html-страниц.

Работа предназначена для бакалавров направлений подготовки 13.03.01 Теплоэнергетика и теплотехника, 13.03.02 Электроэнергетика и электротехника, 35.03.06 Агроинженерия. Кроме того, пособие может быть полезно магистрантам и аспирантам при изучении современных информационно-коммуникационных технологий.

© М.Н. Полковская, Т.С. Бузина  
© Издательство Иркутский ГАУ, 2020

## **1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

Цель освоения дисциплины:

формирование у обучающихся общекультурных и профессиональных компетенций, знаний, умений и навыков владения и эффективного использования информационных технологий.

Основные задачи освоения дисциплины:

- расширение профессионального кругозора бакалавров при автоматизации решения расчетных задач;
- умение адаптировать информационные технологии к решению задач конкретной предметной области.

Результатом освоения дисциплины «Информационно-коммуникационные технологии» является овладение бакалаврами по направлению подготовки 13.03.01 Теплоэнергетика и теплотехника научно-исследовательской деятельностью, в том числе компетенциями, заданными ФГОС ВО.

## **2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП**

Дисциплина «Б1.О.02.04 Информационно-коммуникационные технологии» находится в обязательной части Блока 1 учебного плана. Дисциплина изучается на 1 курсе во 2 семестре.

# СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ MICROSOFT ACCESS 2007

## Лабораторная работа №1. Функциональные возможности Microsoft Access и создание базы данных из шаблона форм.

### Общие сведения

**Базы данных.** Естественно, что в каждой программной системе по анализу информации дается свое определение БД. В **Microsoft Access** под базой данных понимается следующее. *База данных — это совокупность данных и объектов, относящихся к определенной задаче. База данных может содержать объекты: таблицы, запросы, формы, отчеты, макросы и модули.* Основным элементом БД **Microsoft Access** является таблица, а сама база относится к классу реляционных в виду того, что сама таблица относится к этому классу.

Таблица, в которой размещена информация, представленная в базе данных, состоит из строк и столбцов. Строки таблицы называются записями, а столбцы — полями. Каждое поле представляет собой отдельную характеристику данных, сохраненных в базе. Например, если в базе данных размещен список сотрудников, то полем может являться год рождения, фамилия, размер заработной платы, занимаемая должность. Запись базы данных — это набор данных, описывающих один элемент, размещенный в базе. В нашем примере записью является полная информация об одном сотруднике.

Создается впечатление, что базы данных **Access** и электронные таблицы **Excel** очень похожи, а соответствующие программы предназначены для решения одноподобных задач. Однако между ними существуют принципиальные отличия.

Основное преимущество **Access** состоит в удобных средствах поиска и выборки данных из базы данных. Кроме того, в **Access** есть удобная система защиты от несанкционированного доступа. Таким образом, если вам нужно выполнять вычисления и создавать сложные таблицы — используйте **Excel**.

В то же время **Access** -это идеальное решение для поиска и отбора данных.

**Объекты базы данных Access.** Когда вы раньше работали в **Word**, то имели дело с объектом, который назывался документом. В электронных таблицах вы проводили расчеты в объекте, который назывался рабочей книгой, состоящей из рабочих листов. В отличие от **Word** и **Excel** в базу данных **Access** может входить много различных объектов. Кроме таблиц, она

может содержать **запросы, формы, отчеты, макросы и модули.**

Однако именно **таблицы** предназначены для хранения данных. Они являются основным и обязательным элементом базы и создаются в первую очередь.

Следующий структурный элемент — **запросы**. Они представляют собой часть базы данных, отобранную по определенным критериям.

**Форма** — это электронный бланк, используемый для ввода информации в базу данных или для удобного ее просмотра. Еще одним предназначением формы является защита информации.

**Отчеты** служат для отображения итоговых данных из таблиц и запросов в удобном для просмотра и печати виде и помогают проанализировать содержимое исходных таблиц.

**Макросы** автоматизируют некоторые процессы при работе с базами данных, например процедуру заполнения и отбора информации.

**Модули** — это программы, написанные на специальном языке, которые позволяют решить задачи, недоступные для реализации с помощью макросов или стандартных возможностей **Access**.

В окне базы данных можно просмотреть все входящие в нее объекты.

В **Microsoft Access**, прежде чем создавать таблицы, формы и другие объекты необходимо задать структуру базы данных. Хорошая структура базы данных является основой для создания адекватной требованиям, эффективной базы данных. Но прежде чем приступить к процессу проектирования и создания **БД** рассмотрим вопросы загрузки **Access** и начала работы с ней.

## **Запуск и начало работы с СУБД Access**

После запуска **Access** вы увидите окно начала работы с программой (рис.1). Здесь вы можете задать создание базы данных одним из трех ниже описанных способов.

1.Чтобы создать базу данных на основе одного из стандартных шаблонов, выберите название шаблона в разделе **Локальные шаблоны** (рис.2), а затем щелкните по одному из шаблонов, например, **Контакты**. Кроме того, можно взять за основу один из онлайн - шаблонов (**Microsoft Office Online**), если ваш компьютер подключён к Интернет.

2. Если вы создавали базу данных раньше, выберите ее название в разделе **Открыть последнюю базу данных**. Если в этом списке нет имени вашей базы, то щелкните кнопкой мыши на ссылке **Другие** и в появившемся окне **Открытие файла базы данных** укажите нужный файл.

3.Самый простой способ создания новой базы данных — на основе

пустого шаблона в окне **Новая пустая база данных**. В этом случае вам будет доступно больше всего возможностей для формирования ее структуры и ввода записей. Щелкните кнопкой мыши на значке **Новая база данных**. В левой части окна **Новая база данных** отобразится поле **Имя файла**. Введите в него имя новой базы и нажмите кнопку **Создать**. Новая база данных будет сохранена в папке **Мои документы**.

Общий принцип использования интерфейса **Access 2007** — такой же, как у других приложений пакета **Microsoft Office 2007**.

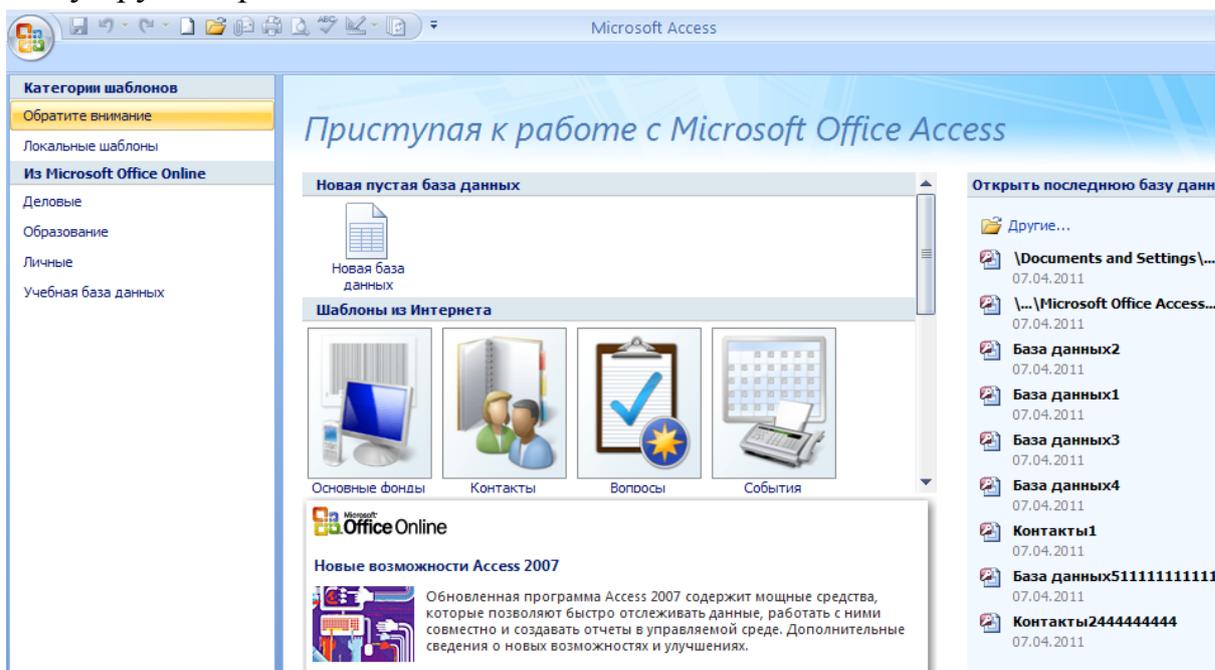


Рисунок 1 – Окно начала работы с программой

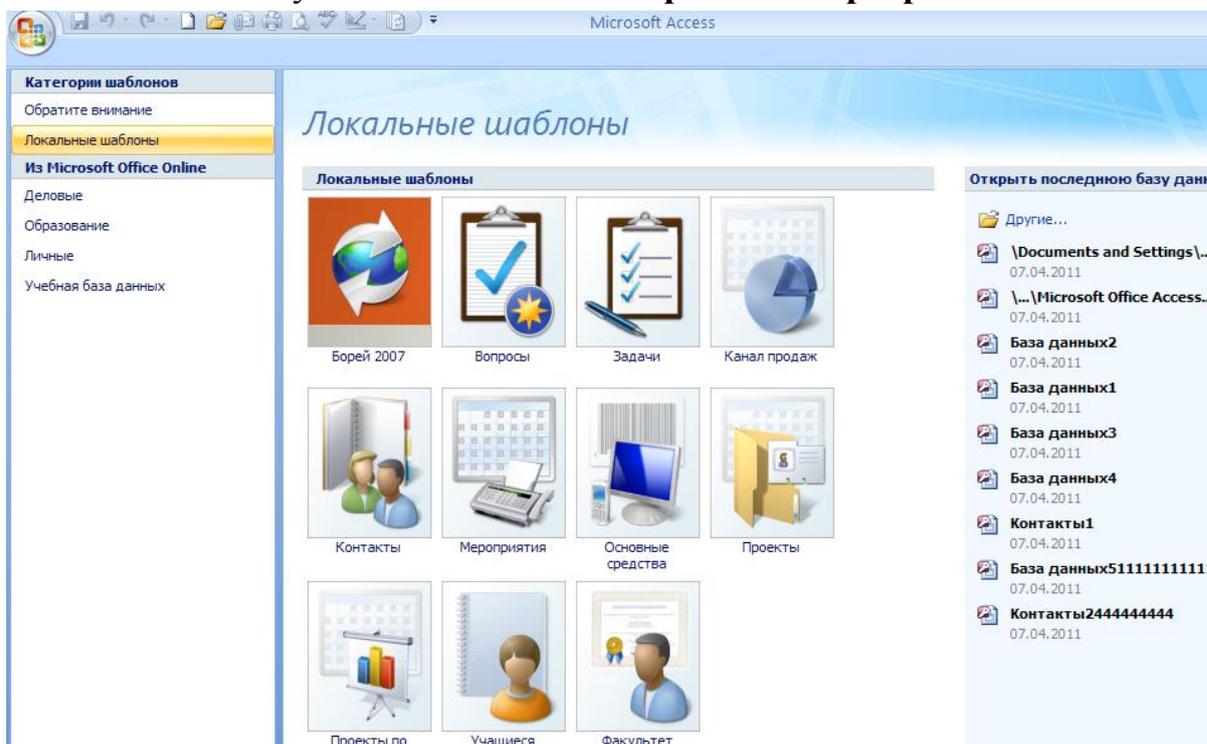


Рисунок 2 – Окно локальных шаблонов

В центре пустой базы данных находится окно активной базы данных (рис.3). Оно состоит из двух частей: области переходов, содержащей список объектов базы данных, и области просмотра. Левую часть окна можно скрыть с помощью кнопки **Открыть/Заккрыть границу области переходов** «<<». Нажав повторно эту кнопку, вы снова отобразите область переходов.

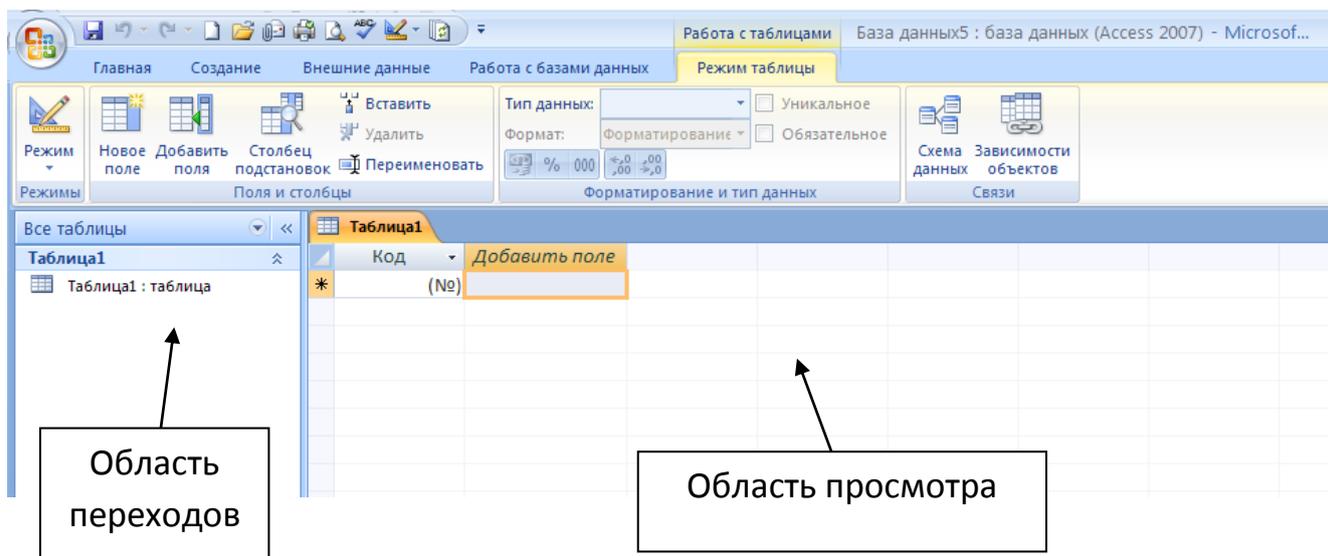


Рисунок 3 – Окно пустой базы данных.

### Технология создания баз данных

Процесс создания базы данных включает в себя:

- проектирование базы данных с целью конкретизации информации;
- создание таблиц;
- формирование необходимых связей между таблицами;
- ввод и импорт информации в таблицы.

Иногда необходимо изменить объем и структуру хранимых данных. В **Access** эти операции выполняются очень просто.

### Проектирование баз данных

Это первый и наиболее ответственный этап. Хотя исправления всегда можно внести в **БД** и ее структуру, но этот процесс может занять очень много времени, тогда как проектирование позволит избежать ненужных затрат. Например, при создании таблицы всегда задается тип данных, которые будут храниться в каждом поле. В процессе разработки **БД** тип данных может измениться (например, с текстового на числовой), что потребует заново вводить информацию в это поле во все записи. Поэтому лучше заранее спроектировать базу данных и возможные варианты использования информации. В этой связи необходимо выделить данные,

которые требуют постоянного обновления, а также определить спектр вопросов, которые придется решать при работе с данными.

Основные этапы проектирования базы данных в **Microsoft Access** сводятся к следующему.

- 1) Определение цели создания базы данных.
- 2) Определение таблиц, которые должна содержать база данных.
- 3) Определение необходимых в таблице полей.
- 4) Задание индивидуального значения каждому полю.
- 5) Определение связей между таблицами.
- 6) Обновление структуры базы данных.
- 7) Добавление данных и создание других объектов базы данных.
- 8) Использование средств анализа в **Microsoft Access**.

Ниже приводится описание выше указанных этап проектирования баз данных в **Microsoft Access**.

**1) Определение цели создания базы данных.** На первом этапе проектирования базы данных необходимо определить цель создания базы данных, основные ее функции и информацию, которую она должна содержать. То есть нужно очертить основные темы таблиц базы данных и информацию, которую будут содержать поля таблиц.

База данных должна отвечать требованиям тех, кто будет непосредственно с ней работать. Для этого нужно определить темы, которые должна покрывать база данных; отчеты, которые она должна выдавать; проанализировать формы, используемые в настоящий момент для записи данных; сравнить создаваемую базу данных с хорошо спроектированной, подобной ей БД.

**2) Определение таблиц, которые должна содержать база данных.** Одним из наиболее сложных этапов в процессе проектирования базы данных является разработка таблиц, так как результаты, которые должна выдавать база данных (отчеты, выходные формы и др.) не всегда дают полное представление о структуре таблицы.

При проектировании таблиц вовсе не обязательно использовать **Microsoft Access**. Сначала лучше разработать структуру на бумаге.

**3) Определение необходимых в таблице полей.** Каждая таблица содержит информацию на отдельную тему, а каждое поле в таблице содержит отдельные сведения по теме таблицы. Например, в таблице с данными о клиенте могут содержаться поля с названием компании, адресом, городом, страной и номером телефона. При разработке полей для таблицы необходимо помнить, что:

- *каждое поле должно быть связано с темой таблицы;*

➤ не рекомендуется включать в таблицу данные, которые являются результатом выражения;

➤ в таблице должна присутствовать вся необходимая информация;

➤ информацию следует разбивать на наименьшие логические единицы (например, поля «Имя» и «Фамилия», а не общее поле «Имя»).

**4) Задание индивидуального значения каждому полю.** Для того чтобы **Microsoft Access** мог связать данные из разных таблиц, например, данные о клиенте и его заказы, каждая таблица должна содержать поле или набор полей, которые будут задавать индивидуальное значение каждой записи в таблице. Такое поле или набор полей называют основным ключом.

**5) Определение связей между таблицами.** После распределения данных по таблицам и определения ключевых полей необходимо выбрать схему для связи данных в разных таблицах. Для этого нужно определить связи между таблицами.

**6) Обновление структуры базы данных.** После проектирования таблиц, полей и связей необходимо еще раз просмотреть структуру базы данных и выявить возможные недочеты. Желательно это сделать на данном этапе, пока таблицы не заполнены данными. Для проверки следует создать несколько таблиц, определить связи между ними и ввести небольшое количество записей в каждую таблицу, затем посмотреть, отвечает ли база данных поставленным требованиям. Рекомендуется также создать черновые выходные формы и отчеты и проверить, выдают ли они требуемую информацию.

**7) Добавление данных и создание других объектов базы данных.** Если структуры таблиц отвечают поставленным требованиям, то можно вводить данные. Затем можно создавать любые запросы, формы, отчеты, макросы и модули.

**8) Использование средств анализа в Microsoft Access.** В **Microsoft Access** существует два средства для усовершенствования структуры баз данных. **Мастер анализа таблиц** исследует таблицу и, в случае необходимости, предлагает новую ее структуру и связи, а также вносит в нее изменения. **Анализатор быстрого действия** просматривает всю базу данных, дает рекомендации по ее улучшению, а также реализует их.

После ознакомления с основными этапами проектирования базы данных в **Access** перейдем непосредственно к их реализации. Прежде всего, рассмотрим способы создания БД.

## Создание базы данных на основе шаблона форм

В **Microsoft Access** реализовано два основных способа создания баз данных. Самый простой способ, что описан как способ 3 в п. 3.2 методических указаний, позволяет создать пустую базу данных на основе пустого шаблона, а затем добавить в нее таблицы, формы, отчеты и другие объекты. Такой способ является наиболее гибким, но требует определения каждого элемента базы данных.

Имеется также возможность сразу создать базу данных с помощью **Мастера** форм определенного типа (первый способ создания БД, что описан как способ 1 в п. 3.2 методических указаний) со всеми необходимыми таблицами, формами и отчетами. Это простейший способ начального создания базы данных. В обоих случаях останется возможность в любое время изменить и расширить созданную базу данных.

Рассмотрим последний способ создания БД.

После запуска **Microsoft Access** на экране появится диалоговое окно **Access** с помощью, которого можно создать или открыть существующую базу данных (рис.1).

Чтобы создать базу данных на основе одного из стандартного шаблона, выберите название шаблона в разделе **Локальные шаблоны** (рис.2), а затем щелкните по одному из шаблонов, например, **Контакты**. Укажите имя файла вашей базы данных, например, **Контакты-ваша фамилия-№ группы** и щелкните кнопку **Создать**.

Откроется окно формы **Список контактов**, которую необходимо заполнить необходимыми данными (рис. 4) и сохранить их с помощью кнопки

**Сохранить** . Следует заметить, что поле **ИД** (индивидуальные данные) является ключевым и заполняется **Access** самостоятельно.

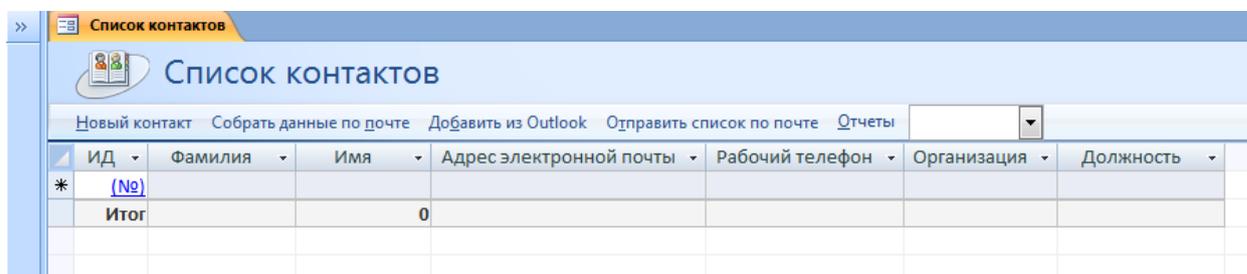


Рисунок 4 – Окно формы **Список контактов**

Кроме этого, следует открыть форму **Сведения о контактах** в области переходов с помощью кнопки **Открыть/Закрыть границу области переходов** и заполнить её данными (рис. 5). Причём в этой форме также

отобразятся однотипные данные по алфавиту, которые вы раньше занесли в форме **Список контактов**. Дополните эту форму необходимыми данными и сохраните их с помощью кнопки **Сохранить** .

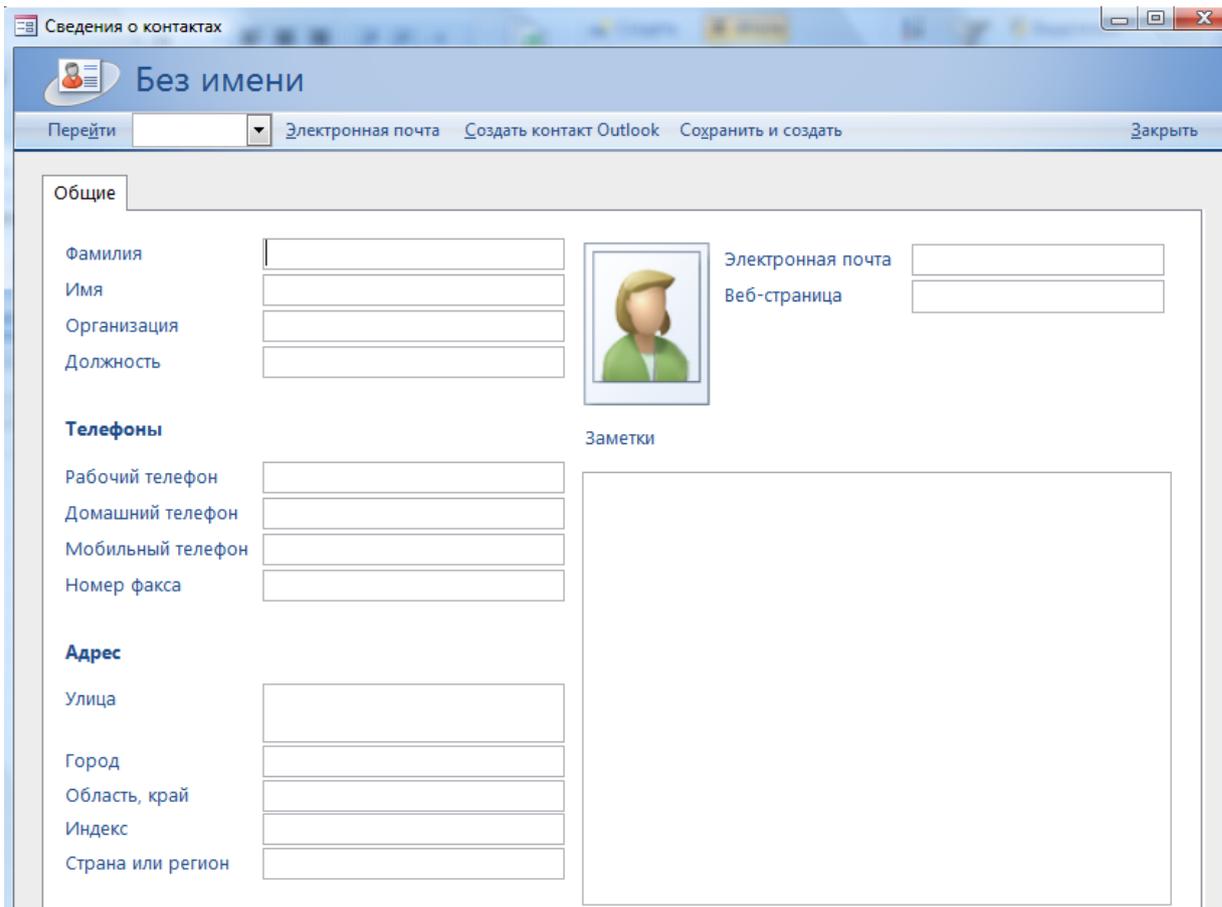


Рисунок 5 – Окно формы **Сведения о контактах**

В области переходов также содержится отчеты **Адресная книга контактов** и **Список телефонов контактов**, которые также можно рассмотреть в конце создания БД.

После окончания работы с окнами форм **Список контактов** и **Сведения о контактах** и отчетов **Адресная книга контактов** и **Список телефонов контактов** нажимаем кнопку **Заккрыть** , чтобы закрыть эти окна. Ваша БД будет сохранена в папке **Мои документы**.

### Практическое задание

1. Откройте **Microsoft Access**. В разделе **Локальные шаблоны** выберите шаблон **Контакты**. Создайте вашу базу данных под именем **Контакты-ваша фамилия-№ группы**.

2. Заполните форму **Список контактов** данными из таблицы 1. Сохраните их.

3. Дополните форму **Сведения о контактах** данными адреса, рабочего

телефона и заметками из таблицы 2. Страну в форме заполните в зависимости от указанного адреса.

4. Результаты работы по созданию базы данных рассмотрите в отчетах **Адресная книга контактов и Список телефонов контактов.**

5. Сохраните вашу базу данных в папке **Мои документы.**

Таблица 1 – Список контактов

ИД	Фамилия	Имя	Адрес электронной почты	Рабочий телефон	Организация	Должность
1	Корнев	Олег	Kor@mail.ru	5767887	НИИССУ	Инженер
2	Сычев	Иван	Sich@mail.ru	3214566	НИИТ	Старший инженер
3	Иванов	Петр	Ivan@mail.ru	3145566	МИФИ	Инженер
4	Кротов	Петр	Krot@mail.ru	1876543	НИИССУ	Инженер
5	Котов	Иван	Kot@mail.ru	1235555	Завод “Рубин”	Инженер
6	Быков	Иван	Bic@mail.ru	044556324	ОКБ “Вымпел”	Старший инженер
7	Тощев	Олег	Toch@mail.ru	057567543	НИИССУ	Инженер
8	Громов	Петр	Grom@mail.ru	863567345	МТЗ	Инженер
9	Бабина	Анна	Bab@mail.ru	863567345	НИИССУ	Инженер
10	Хрущ	Саша	Hru@mail.ru	3456665	НИИВТ	Начальник отдела

Таблица 2 – Сведения о контактах

Фамилия	Имя	Адрес	Домашний телефон	Заметки
Корнев	Олег	115477 Москва Пролетарский пр. д. 9	2547887	Заказ на поставку компьютеров
Сычев	Иван	110877 Москва ул. Луганская д.3	5478912	Заказ на поставку мониторов
Иванов	Петр	114777 Москва Каширское ш. д.1	1254784	Заказ на поставку мониторов и ПУ
Кротов	Петр	100333 г.Александров Московской обл. ул. Ленина д.34	1245784	Заказ на поставку телевизоров
Котов	Иван	133444 Москва, ул. Кораб- лестроителей д.66	5421025	Заказ на поставку кинескопов
Быков	Иван	121355 Киев ул. Рыжова д.1	044142587	Заказ на поставку компьютеров
Тощев	Олег	234543 Харьков ул. Жукова д.55	057142635	Заказ на поставку мониторов
Громов	Петр	565776 Ростов ул. Стахова д.89	863120369	Заказ на поставку процессоров
Бабина	Анна	565776 Ростов ул. Федорова д.78	862225478	Заказ на поставку мониторов
Хрущ	Саша	123456 Москва Варшавское ш. д.98	3456677	Заказ на поставку клавиатуры

## **Контрольные вопросы**

1. Что включает в себя база данных?
2. С какими объектами база данных работает?
3. Для каких целей предназначены различные объекты базы данных?
4. Из чего состоит таблица базы данных?
5. Какие основные этапы проектирования баз данных содержит

## **Microsoft Access?**

6. Как сохранить базу данных в папке **Мои документы**?
7. Какие вы знаете способы создания баз данных?

## Лабораторная работа №2. Создание таблицы в базе данных в режиме таблицы

### Способы создания таблиц в базе данных

После создания базы данных переходим к проектированию таблиц, которые входят в БД **Контакты-ваша фамилия-№ группы**. В Access существует несколько методов создания таблиц, некоторые из них приведены в табл. 3.

В зависимости от выбранного метода процесс создания таблицы включает одну или несколько нижеследующих операций:

- 1) определение полей таблицы;
- 2) определение типа данных для каждого поля;
- 3) задание размера поля (только для текстовых и числовых полей);
- 4) установка ключевого поля;
- 5) присвоение имени таблице и ее сохранение.

Таблица 3 – Методы создания таблиц

<b>МЕТОДЫ СОЗДАНИЯ ТАБЛИЦ</b>	
<b>Метод</b>	<b>Описание</b>
<b>Режим таблицы</b>	Можно сразу вводить данные. Access автоматически назначает тип поля по виду информации, которая вводится, а также устанавливает размеры текстовых и числовых полей. После сохранения таблицы Access предложит либо установить имя таблицы самостоятельно, либо принять предложенное Access.
<b>Режим конструктора</b>	Позволяет задать поля, тип данных для каждого из них и размеры для текстовых и числовых полей, установить ключевое поле и сохранить макет таблицы вам самостоятельно. Затем перейдя в <b>Режим таблицы</b> можно вводить данные в таблицу
<b>Шаблоны таблиц</b>	Позволяет выбрать поле из предлагаемых образцов таблиц. Тип данных и другие свойства уже определены. Затем в режиме таблицив можно вводить данные в таблицу
<b>Импорт таблицы</b>	Импортируются вместе с данными все имена полей и типы данных. Однако новой таблице надо присвоить имя, определить ключевое слово или предоставить это сделать Access. Недостаток — надо изменить свойства и параметры таблицы.

## Создание таблиц в Режиме таблицы

Чтобы создать таблицу в **Режиме таблицы** щелкните кнопку **Таблица** в группе **Таблица** из вкладки **Создание** (рис.6.). После этого программа будет работать в режиме таблицы, или ввода данных. Вы можете сразу же вводить записи в таблицу, помня при этом, что они представляют собой строки, а поля — это столбцы.

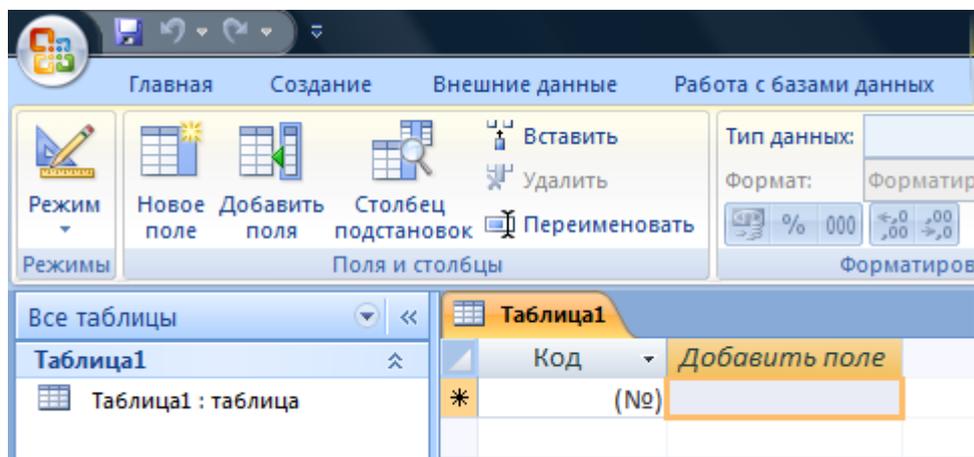


Рисунок 6 – Окно для создания таблицы в Режиме таблицы

Первый столбец таблицы обозначен как **Код**. Это поле заблокировано для пользователя. При вводе данных **Access** автоматически проставляет сюда последовательные номера. При добавлении сведений в текущую запись всегда отображается новая пустая запись, помеченная звездочкой. В **Access** новые записи всегда добавляются в конец таблицы.

Обратите внимание: в нижней части окна **Access** в строке **Запись** присутствует поле, позволяющее быстро переходить к нужным записям.

В режиме ввода данных новые поля обозначаются как Поле1, Поле2, Поле3 и т. д. Вы можете изменить присвоенные по умолчанию названия и задать свои. Для этого воспользуйтесь пунктом **Переименовать столбец** контекстного меню поля или кнопкой **Переименовать**, расположенной в группе **Поля и столбцы** на вкладке **Режим таблицы** контекстного инструмента **Работа с таблицами**.

С помощью кнопок этой же группы (или пунктов упомянутого контекстного меню) можно удалять существующие поля (**Удалить**) и вставлять новые (**Вставить**).

Ширина полей таблицы и величина строки для ввода записей изменяются так же, как в **Excel**.

Чтобы сохранить таблицу щелкните кнопку **Сохранить** . В окне **Сохранение** наберите имя таблицы, например, **Таблица 1-ваша фамилия-№ группы** и щелкните кнопку **ОК**.

Для изменения имени таблицы закройте её, щелкните правой мышкой по её имени в **Области переходов** и воспользуйтесь из контекстного меню кнопкой **Переименовать**.

### Практическое задание

1.Создайте в вашей базе данных **Контакты-ваша фамилия-№ группы**, созданной в предыдущей лабораторной работе, таблицу в **Режиме таблица**. Для создания этой таблицы воспользуйтесь данными из таблицы 4.

Таблица 4 – **Таблица 1-ваша фамилия-№ группы**

Код контакта	Дата звонка	Продолжительность звонка	Стоимость минуты звонка
1	7.02.11	1	10,55
2	7.02.11	3	5,1
3	7.02.11	12	7
4	7.02.11	7	5,1
5	2.02.11	9	7
6	3.02.11	4	7
7	4.02.11	9	5,1
8	5.02.11	14	7
9	7.02.11	23	5,1
10	7.02.11	21	10,55

2.Переименуйте поля **Продолжительность звонка** на **Продолжительность звонка (мин)** и **Стоимость минуты** на **Стоимость минуты звонка (руб.)**.

3.Сохраните эту таблицу в вашей базе данных под именем **Таблица 1-ваша фамилия-№ группы**.

### Контрольные вопросы

1. Какие существуют методы (способы) создания таблиц в базе данных **Microsoft Access**?

2.Как переименовать столбец в таблице в **Режиме таблицы**?

3.Как удалить столбец в таблице в **Режиме таблицы**?

4.Как изменить имя таблицы в базе данных?

## Лабораторная работа №3. Создание таблиц в базе данных в режиме конструктора и с помощью шаблонов таблиц

### Создание таблицы в режиме Конструктор таблиц

Прежде чем рассматривать создание таблицы в режиме конструктора, кратко остановимся на типах данных и их характеристике (табл. 5).

Таблица 5 – Типы данных

Тип данных	Используется	Размер
1	2	3
<b>Текстовый</b>	Текст или комбинация текста и чисел, например, адрес, а также числа, не требующие вычислений, например, номера телефонов, номенклатурные номера или почтовый индекс.	До 255 символов. Хранятся только введенные в поле символы. Позиции, не использованные в текстовом поле, не хранятся. Для управления максимальным числом вводимых символов
<b>Поле МЕМО</b>	Длинный текст или числа, например, пометки или описание.	До 64000 символов.
<b>Числовой</b>	Числовые данные, используемые для математических вычислений, за исключением вычислений, включающих денежные операции (используется денежный тип). Для определения числового типа следует определить свойство <b>Размер поля</b> .	1, 2, 4 или 8 байт. 16 байт только для кодов репликации.
<b>Дата/время</b>	Даты и время.	8 байт.
<b>Денежный</b>	Значения валют. Денежный тип используется для предотвращения округлений во время вычислений, Предполагает до 15 символов в целой части числа и 4 — в дробной.	8 байт.

1	2	3
<b>Счетчик</b>	Автоматическая вставка последовательных (отличающихся на 1) или случайных чисел при добавлении записи.	4 байта.
<b>Логический</b>	Поля, содержащие только одно или два значения, таких как «Да/Нет», «Истина/Ложь», «Включено/Выключено».	1 бит.
<b>Объекты OLE и Вложение</b>	Объекты (например, документы <b>Microsoft Word</b> , электронные таблицы <b>Microsoft Excel</b> , рисунки, звуки и другие данные}, созданные в других программах, использующих протокол OLE. Объекты могут быть связанными или внедренными в таблицу <b>Microsoft Access</b> . Для отображения объекта OLE в форме или отчете необходимо использовать присоединенную рамку объекта.	До 1 гигабайта (ограничено объемом диска).

Создание таблицы в режиме **Конструктор** происходит в два этапа. На первом этапе создается макет таблицы, в который вводятся имена столбцов таблицы и их типы данных, назначается ключевое поле и этот макет сохраняется под именем, например, **Таблица 2-ваша фамилия-№ группы**. На втором этапе этот макет таблицы заполняется данными и снова сохраняется. Для перехода из режима **Конструктор** в **Режим таблицы** и

наоборот используется кнопка **Режим**  из группы **Режимы** из вкладки **Конструктор** или **Режим таблицы**.

Для создания макета таблицы в режиме **Конструктора** необходимо:

-щёлкнуть кнопку **Конструктор таблиц** в группе **Таблицы** из вкладки **Создание** (рис. 7);

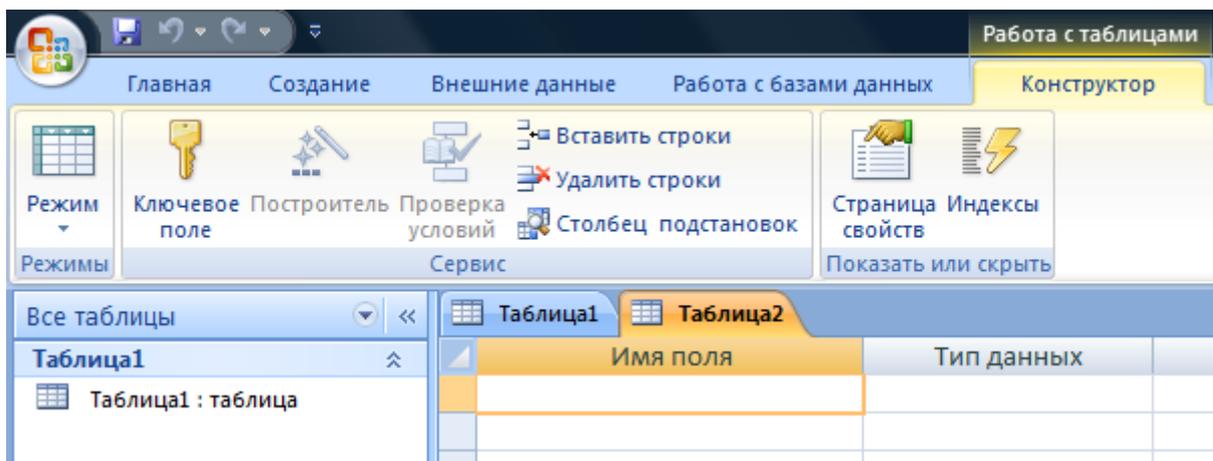


Рисунок 7 – Окно для создания макета таблицы в Режиме конструктор таблиц

-заполнить имена полей, которые представляют собой имена столбцов из обычной таблицы **Word**, и выбрать из всплывающего меню типы данных этих полей, воспользовавшись кнопкой  в столбце **Тип данных**;

-задать ключевое поле путем выделения строки макета таблицы и щелчка мышью по значку ключевого поля  (это поле является уникальным и необходимо для связывания таблиц между собой; когда оно заполняется, то данные в нём не повторяются);

-сохранить макет таблицы под именем, например, **Таблица 2-ваша фамилия-№ группы**, щелкнув кнопку **Сохранить** ;

- перейти из режима **Конструктор** в **Режим** таблицы, воспользовавшись кнопкой **Режим**  из группы **Режимы** из вкладки **Конструктор**;

-заполнить таблицу данными и снова сохранить таблицу.

Для изменения имени таблицы закройте её, щелкните правой мышкой по её имени в **Области переходов** и воспользуйтесь из контекстного меню кнопкой **Переименовать**.

Пример заполнения таблицы **Таблица 2-ваша фамилия-№ группы** показан на рис. 8.

Имя поля		Тип данных
№ звонка		Счетчик
Код звонка		Числовой
Организация		Текстовый
Дата звонка		Дата/время
Время звонка		Дата/время
Тема		Текстовый
Количество (шт)		Числовой

Рисунок 8 – Заполнение макета таблицы Таблица 2-ваша фамилия-№ группы

Для удаления или вставки новых полей в макет таблицы (режим **Конструктор таблиц**) предназначены кнопки **Вставить строки** и **Удалить строки** группы **Сервис** на вкладке **Конструктор** контекстного инструмента **Работа с таблицами**.

Чтобы сохранить таблицу щелкните кнопку **Сохранить** . В окне **Сохранение** наберите имя таблицы, например, **Таблица 2-ваша фамилия-№ группы** и щелкните кнопку **ОК**.

В случае необходимости, для изменения имени таблицы закройте её, щелкните правой мышкой по её имени в **Области переходов** и воспользуйтесь из контекстного меню кнопкой **Переименовать**.

### Создание таблицы с помощью шаблонов таблиц

Процесс создания таблицы с помощью **Шаблонов таблиц** сводится к заданию параметров таблицы на основе образцов, которые **Access** предлагает в появляющихся диалоговых окнах. Учитывая, что **Мастер** создан для широкого класса задач, которые могут полностью не совпадать с потребностями пользователя, можно выбрать подходящий тип таблицы и ее поля, а затем присвоить им другие имена.

В **Access** предусмотрены следующие варианты шаблонов:

- **Контакты** (Contacts) — таблица, созданная на основе этого шаблона, будет включать поля для ввода контактных данных: фамилии, номеров телефонов, адресов, индекса, факса и т. д.;
- **Задачи** (Tasks) — шаблон, содержащий поля, предназначенные для указания дат начала и окончания задачи, для ввода информации о ходе выполнения задачи, а также о ее статусе или приоритете;
- **Вопросы** (Issues) — включает набор полей для ввода состояния, категории, даты открытия, приоритета, описания и ряда других данных;
- **События** (Events) — в таблице, созданной на основе этого шаблона, будут

присутствовать поля для ввода дат начала и окончания события, а также места, описания и названия;

- **Основные фонды (Assets)** — такая таблица будет содержать информацию о категории, условиях, цене, дате приобретения, модели товара, изготовителе, текущей стоимости, дате списания и т. д.

Для создания таблицы в режиме **Шаблонов таблиц** необходимо:

-щёлкнуть кнопку **Шаблоны таблиц** в группе **Таблицы** из вкладки **Создание**;

-из предлагаемого списка шаблонов (**Контакты, Задачи, Вопросы, События** и **Основные фонды**), выбрать необходимый шаблон, например, **События** (рис. 9);

-при необходимости ввести изменения в названии полей;

-заполнить таблицу записями и сохранить её.

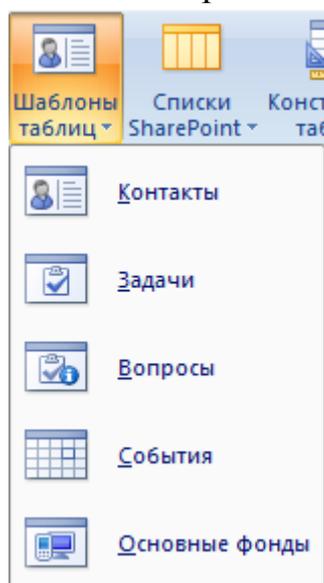


Рисунок 9 – Шаблоны таблиц

Чтобы сохранить таблицу щелкните кнопку **Сохранить** . В окне **Сохранение** наберите имя таблицы, например, **События-ваша фамилия-№ группы** и щелкните кнопку **ОК**.

В случае необходимости, для изменения имени таблицы закройте её, щелкните правой мышкой по её имени в **Области переходов** и воспользуйтесь из контекстного меню кнопкой **Переименовать**.

### Практическое задание

1. Создайте в базе данных **Контакты-Ваша фамилия-№ группы** в режиме **Конструктор таблиц** таблицу с именем **Таблица 2-ваша фамилия-№ группы**. В этой таблице используйте типы данных полей о звонках, представленные на рис. 8 методических указаний, а также введите в эту

таблицу данные нижеприведенной таблицы 6. В качестве ключевого поля используйте **Код звонка**. Сохраните эту таблицу в вашей базе данных под именем **Таблица 2-ваша фамилия-№ группы**.

Таблица 6 – **Таблица 2-ваша фамилия-№ группы**

Код звонка	№ звонка	Организация	Дата звонка	Время звонка	Тема	Количество
1	1767887	НИИССУ	01.02.11	12:12:00	Заказ на поставку компьютеров	150
2	3214566	МИФИ	01.02.11	13:23:56	Заказ на поставку мониторов	70
3	3145566	НИИВТ	05.02.11	17:34:00	Заказ на поставку мониторов	100
4	1876543	ОКБ "Стрела"	07.02.11	10:23:00	Заказ на поставку телевизоров	30
5	1235555	НИИТ	20.02.11	09:00:00	Заказ на поставку мониторов	150
6	556324	МИФИ	01.03.11	14:45:56	Заказ на поставку компьютеров	140
7	567543	Завод "Молния"	07.03.11	16:34:00	Заказ на поставку комплектующих	150
8	567345	НИИВТ	12.03.11	13:12:55	Заказ на поставку процессоров	80
9	567005	НИИССУ	18.03.11	13:23:56	Заказ на поставку мониторов	140
10	3456665	НИИССУ	01.04.11	18:00:23	Заказ на поставку мониторов	170

2. Создайте в базе данных **Контакты-ваша фамилия-№ группы** в режиме **Шаблоны таблиц** таблицу с именем **События-Ваша фамилия-№ группы**. Из списка **Шаблонов таблиц** используйте шаблон **События**. В создаваемой таблице переименуйте поля **Код** на **Код события**, **Место** на **Организация**, **Описание** на **Количество (шт)**, удалите поле **Вложение** . Измените тип поля **Количество (шт)** на **Числовой**. Используйте для заполнения создаваемой таблицы данные из ниже приведенной таблицы 7. Сохраните эту таблицу в вашей базе данных под именем **События -ваша фамилия-№ группы**.

Таблица 7 – События -ваша фамилия-№ группы

Код события	Название	Время начала	Время окончания	Организация	Количество (шт)
1	Отгрузка компьютеров	05.05.11	07.05.11	НИИССУ	50
2	Отгрузка компьютеров	08.05.11	10.05.11	МИФИ	40
3	Отгрузка мониторов	11.05.11	12.05.11	МИФИ	40
4	Отгрузка комплектующих	13.05.11	15.05.11	Завод «Молния»	50
5	Отгрузка компьютеров	16.05.11	18.05.11	НИИССУ	50
6	Отгрузка мониторов	18.05.11	21.05.11	НИИССУ	100
7	Отгрузка телевизоров	22.05.11	22.05.11	ОКБ «Стрела»	20
8	Отгрузка мониторов	23.05.11	23.05.11	МИФИ	20
9	Отгрузка мониторов	24.05.11	26.05.11	НИИТ	50
10	Отгрузка компьютеров	28.05.11	28.05.11	НИИВТ	20

### Контрольные вопросы

1. Какие типы данных таблиц из базы данных вы знаете?
2. Как переключится в режим **Конструктор таблиц** из **Режима таблицы** и наоборот?
3. Как удалить поле в макете таблицы в режиме **Конструктор таблиц**?
4. Как вставить новое поле в макете таблицы в режиме **Конструктор таблиц**?
5. Какие вы знаете шаблоны таблиц?

## Лабораторная работа №4. Поиск, сортировка и фильтрация записей в таблице базы данных

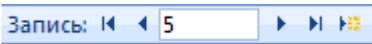
В таблице БД, содержащей сотни и тысячи записей, часто бывает очень трудно быстро найти необходимую информацию. Для упрощения этой процедуры Access предлагает средства для поиска и просмотра данных в базе данных. К ним можно отнести:

- переход к необходимой записи;
- поиск записи по тексту, содержащемуся в любом из полей;
- сортировка записей по возрастанию или убыванию значений конкретного поля;
- изменение количества отображаемых записей с помощью фильтра.

### Поиск записи по ее номеру в Режиме таблицы или Формы

Для поиска записи необходимо выполнить следующие действия.

1. Открыть таблицу в **Режиме таблицы** или **Форму** в **Режиме формы**.
2. Установить указатель на номер в поле номера записи, расположенному в нижней части окна таблицы базы данных, и щелкнуть по нему мышью.

3. Ввести номер нужной записи, например, 5  и нажать клавишу **ENTER**. Результатом выполнения этого пункта станет выделение записи с указанным номером.

### Поиск записи по тексту, содержащемуся в любом из полей

Поиск одной или нескольких записей наиболее просто можно выполнить с помощью команды **Найти**  в группе **Найти** из вкладки **Главная**. Данная операция выполняется следующим образом.

1. Отобразить таблицу в **Режиме таблицы**.
2. Выделить столбец (щелкнуть по его заголовку мышью), фразу или слово, в котором предполагается выполнить поиск.
3. Запустить операцию поиска, нажав кнопку **Найти**.
4. В появившемся диалоговом окне **Поиск и замена** в поле **Образец** ввести фразу для поиска, если она не была введена раньше в пункте 2. В поле **Совпадение** задать тип совпадения образца поиска с данными записей поля. Если требуется расширить поиск, то в поле **Просмотр** указать направление поиска (опции — *вверх, вниз, все*).
5. Для начала поиска нажать кнопку **Найти далее**.

6. Для перехода к следующей записи, содержащей заданный текст, активизировать кнопку **Найти далее**.

7. Для окончания поиска нажать кнопку **Заккрыть**.

Нетрудно заметить, что система позволяет выполнять поиск, как целых слов, фраз, так и частей слова.

### Сортировка записей в таблице или форме

Сортировка записей сводится к изменению порядка следования записей в соответствии с заданным критерием. В **Access** применяется сортировка *по возрастанию* или *убыванию* записей. При этом в **текстовом поле** записи располагаются в *алфавитном* или *обратном алфавитном* порядке, в **поле даты** — от самой *давней даты* к *самой последней* или, наоборот, в **числовом поле** — от *самого маленького* к *самому большому* или наоборот.

Для сортировки записей необходимо выполнить следующие действия.

1. В **Режиме формы** или в **Режиме таблицы** выбрать поле (столбец), по которому будет выполняться сортировка.

2. Выполнить одно из следующих действий:

- для сортировки по возрастанию нажать кнопку сортировка **По возрастанию**  в группе **Сортировка и фильтр** из вкладки **Главная**;

- для сортировки по убыванию нажать кнопку сортировка **По убыванию** .

Кроме этого, в **Access** ещё один удобный инструмент для сортировки записей. В режиме ввода данных рядом с именем каждого поля присутствует стрелочка. Щелкнув на ней кнопкой мыши, вы сможете установить порядок сортировки данных по этому полю. Допускается упорядочение по возрастанию (пункт Сортировка от А до Я) и по убыванию (пункт Сортировка от Я до А) (рис. 9).

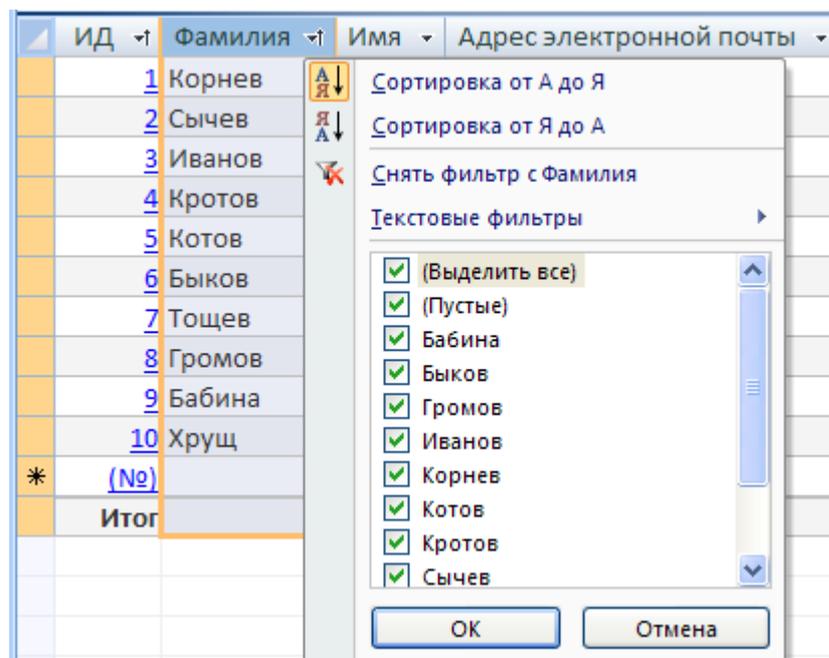


Рисунок 9 – Сортировка записей

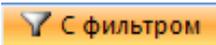
Чтобы вернуть предыдущий порядок размещения записей, воспользуйтесь кнопкой **Очистить все сортировки**  в группе **Сортировка и фильтр**.

### Применение фильтров для отбора записей

Еще одним средством Access, позволяющим сортировать (и не только) записи, является применение фильтров.

**Фильтр** — набор условий, применяемых для отбора подмножества записей или для сортировки записей.

**Фильтр с выбранным значением поля.** С помощью фильтров можно настроить отображение записей в таблицах. Самый простой фильтр позволяет отобразить в таблице записи с выбранным значением поля. Щелкните кнопкой мыши на стрелочке рядом с именем поля и отметьте значения нужных полей, например, **Отгрузка мониторов**, установив соответствующие флажки (см. рис. 10). После применения фильтра в нижней части таблицы в строке прокрутки появится кнопка

**С фильтром** , выделенная другим цветом. Чтобы отменить фильтр и отобразить все записи, нажмите эту кнопку еще раз.

**Текстовые фильтры** и **Числовые фильтры.** В Access предусмотрен фильтр, позволяющий отобразить значения полей, удовлетворяющие определенным условиям. Чтобы применить его к текстовым полям, щелкните кнопкой мыши на стрелочке рядом с именем поля и воспользуйтесь пунктом

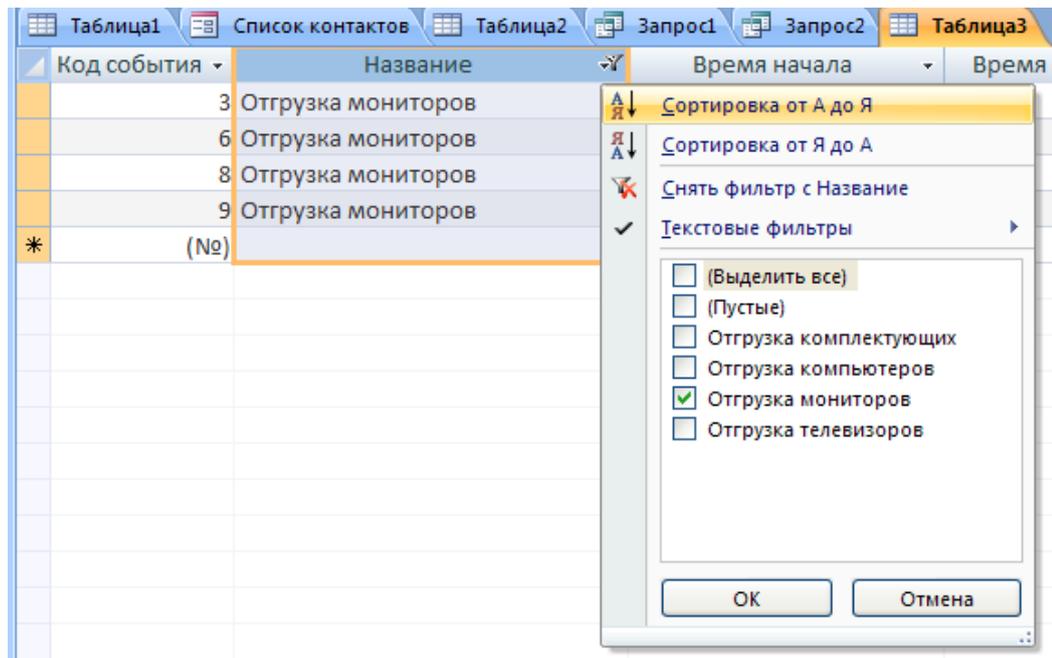


Рисунок 10 – Фильтр с выбранным значением поля Отгрузка мониторов

### Текстовые фильтры.

При этом вы можете выбрать логические операции для построения условий фильтрации. Можно отобрать записи, которые будут равны или не равны определенному значению, начинаться или не начинаться с указанного значения, содержать или не содержать его и т. д. После выбора логической операции вы увидите окно для ввода значения, с которым нужно сравнивать данные. Введите туда необходимое значение и щелкните кнопку **ОК** (рис. 3).

При построении фильтра отбора для числовых полей воспользуйтесь пунктом **Числовые фильтры**, аналогично как с **Текстовыми фильтрами**.

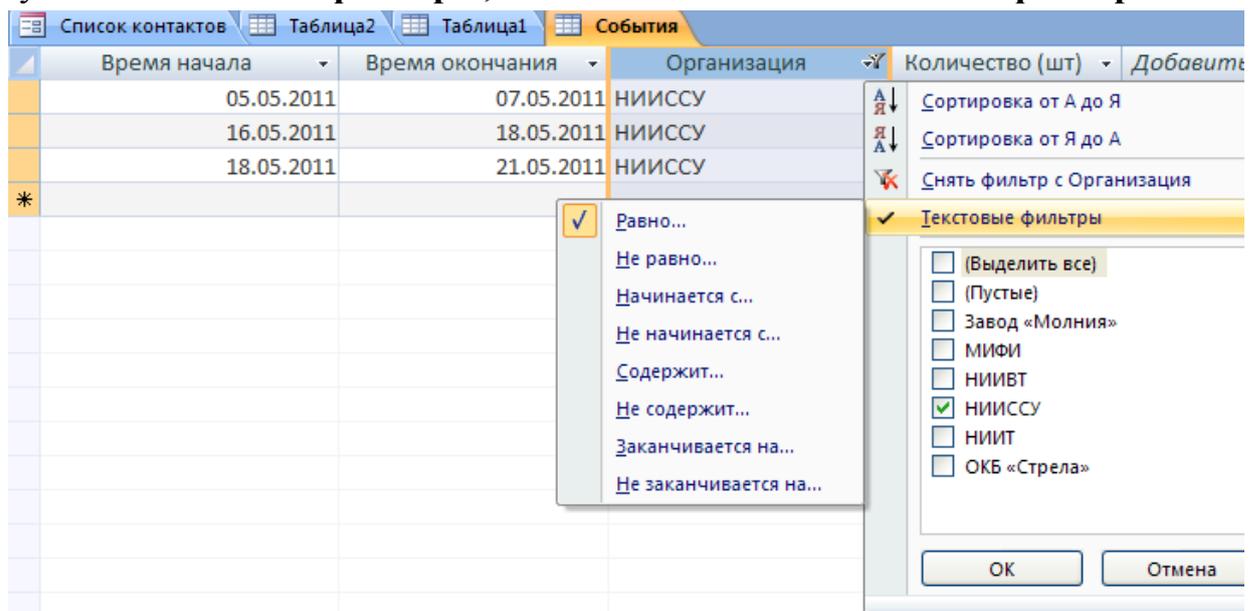
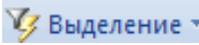
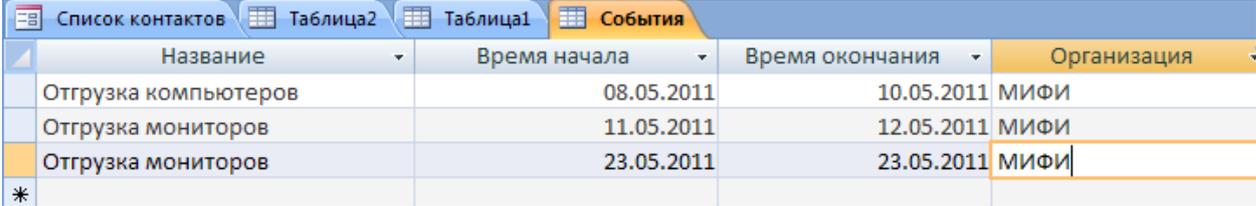


Рисунок 11 – Текстовый фильтр с логическим значением равно НИИССУ

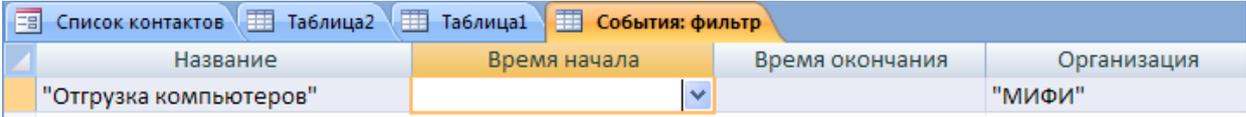
**Фильтр по выделенному значению поля.** Еще один фильтр, доступный при работе с таблицами **Access**, позволяет отобразить только те записи, в которых есть выделенное значение поля. Для применения такого фильтра сначала выделите значение, а потом нажмите кнопку **Выделение**  в группе **Сортировка и фильтр** (рис. 12). В появившемся окне укажите нужное условие. По умолчанию используется условие **Равно** и в таблице отображаются записи, значения которых равны выделенному значению поля.



Название	Время начала	Время окончания	Организация
Отгрузка компьютеров	08.05.2011	10.05.2011	МИФИ
Отгрузка мониторов	11.05.2011	12.05.2011	МИФИ
Отгрузка мониторов	23.05.2011	23.05.2011	МИФИ

Рисунок 12 – Применение фильтра по выделенному значению поля МИФИ

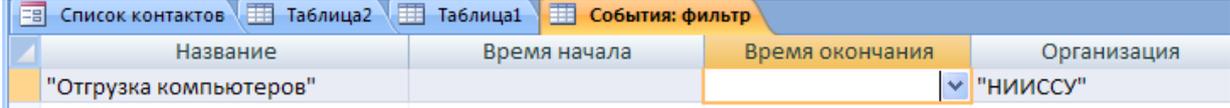
**Фильтр с заданием сложных условий отбора данных.** В программе **Access** также существует фильтр, который удобно применять для задания сложных условий отбора данных, расположенных в разных полях таблицы. В группе **Сортировка и фильтр** нажмите кнопку **Дополнительно** и в появившемся меню выберите пункт **Изменить фильтр**. В открывшемся окне задайте условия отбора.



Название	Время начала	Время окончания	Организация
"Отгрузка компьютеров"			"МИФИ"

Рисунок 13 – Условия отбора Отгрузка компьютеров и МИФИ

При создании такого фильтра в нижней части окна появится вкладка **Или**  **Найти**  **Или** , с помощью которой можно настроить дополнительные условия отбора, объединенные между собой логическим условием **Или** (рис. 14).



Название	Время начала	Время окончания	Организация
"Отгрузка компьютеров"			"НИИССУ"

Рисунок 14 – Условия отбора Отгрузка компьютеров и НИИССУ, вводимое после условия Или

Для отображения результата фильтрации нажмите кнопку **Применить фильтр** (рис. 15). Повторное нажатие этой кнопки отменит использование фильтра.

Код события ▾	Название ▾	Время начала ▾	Время окончания ▾	Место
1	Отгрузка компьютеров	05.05.2011	07.05.2011	НИИССУ
2	Отгрузка компьютеров	08.05.2011	10.05.2011	МИФИ
5	Отгрузка компьютеров	16.05.2011	18.05.2011	НИИССУ
(№)				

### Рисунок 15 – Фильтр с заданием сложных условий отбора данных: Отгрузка компьютеров и НИИССУ Или Отгрузка компьютеров и МИФИ

При создании условия отбора для числовых полей можно применять операции сравнения.

Удалить все созданные фильтры можно, нажав кнопку **Дополнительно** и выбрав пункт **Очистить все фильтры**.

#### Практическое задание

1. Проведите поиск записи по номеру 5 в таблице **Таблица 1-ваша фамилия-№ группы**.

2. Проведите поиск записей по имени организации (компании) **НИИССУ** в **Таблица 2-ваша фамилия-№ группы**.

3. В соответствии с пунктом 3.3. методических указаний проведите сортировку записей по возрастанию в столбце (поле) **Фамилия** формы **Список контактов** (рис. 9).

4. С помощью функции **Фильтр с выбранным значением поля** проведите поиск записей со значением **Отгрузка мониторов** в поле **Название** в таблице **События-ваша фамилия-№ группы** (рис. 10).

5. С помощью функции **Текстовые фильтры** проведите поиск записей, содержащих название организации **НИИССУ** в таблице **События-ваша фамилия-№ группы** (рис.11).

6. С помощью функции **Фильтр по выделенному значению поля** проведите поиск записей, содержащих название организации **МИФИ** в таблице **События-ваша фамилия-№ группы** (рис.12).

7. С помощью функции **Фильтр с заданием сложных условий отбора данных** проведите поиск записей с условиями отбора **Отгрузка компьютеров и НИИССУ Или Отгрузка компьютеров и МИФИ** в таблице **События-ваша фамилия-№ группы** (рис.15).

#### Контрольные вопросы

1. Какой вы знаете наиболее удобный инструмент для сортировки записей?
2. Как провести поиск записи с номером 123 в таблице базы данных?
3. Что такое фильтр?
4. Какие вы знаете разновидности фильтров?

## Лабораторная работа №5. Применение запросов для выборки записей из базы данных

Для извлечения записей из БД и составления отчетов применяются запросы.

**Запрос** — инструкция на отбор данных, хранящихся в таблицах, или требование на выполнение определенных действий с данными.

С помощью запроса извлекается из базы данных информация, отвечающая на конкретный вопрос, например, "Сколько в фирме ОАО "Трест" работает главных менеджеров?". *Задаваемые в запросе условия — значения конкретных полей — называются критериями отбора.* Запрос позволяет создать общий набор записей из данных, находящихся в разных таблицах, и использовать этот набор как источник данных для формы или отчета.

**Microsoft Access** позволяет при работе с запросами в отличие от простого просмотра записей, как это было в предыдущей лабораторной работе:

- сохранить запрос с записями;
- использовать в запросе формулы и выражения на основе значений полей таблицы;
- сохранить записи, полученные в результате выполнения запроса, в виде новой таблицы.

### Создание запросов в Access

Как и другие объекты базы данных, запросы можно синтезировать несколькими способами. Так, запрос можно создать самостоятельно или применить **Мастер запросов**, чтобы разработать простой запрос стандартными средствами. Индивидуальный запрос можно синтезировать в **Режиме конструктора**. В запрос можно преобразовать фильтр, который применяется для отбора записей. Все созданные запросы сохраняются как **Запрос1**, **Запрос2** и т.д. и отображаются своими именами в области переходов.

#### Создание запросов на выборку с помощью Конструктора запросов

Рассмотрим процедуру создания запросов с помощью **Конструктора запросов**. Она сводится к следующему.

1. Щелкнуть кнопку **Конструктор запросов** в группе **Другие** на вкладке **Создание**.

2. В открывшемся окне **Добавление таблицы** выбрать таблицу (например, **События**) или запрос и нажать кнопку **Добавить**.
3. Повторить шаг 2 для каждой таблицы и запроса.
4. Для отображения запроса в режиме **Конструктора** нажать кнопку **Заккрыть**.
5. На экране появится диалоговое окно **Запрос1**, состоящее из двух частей (рис. 16).

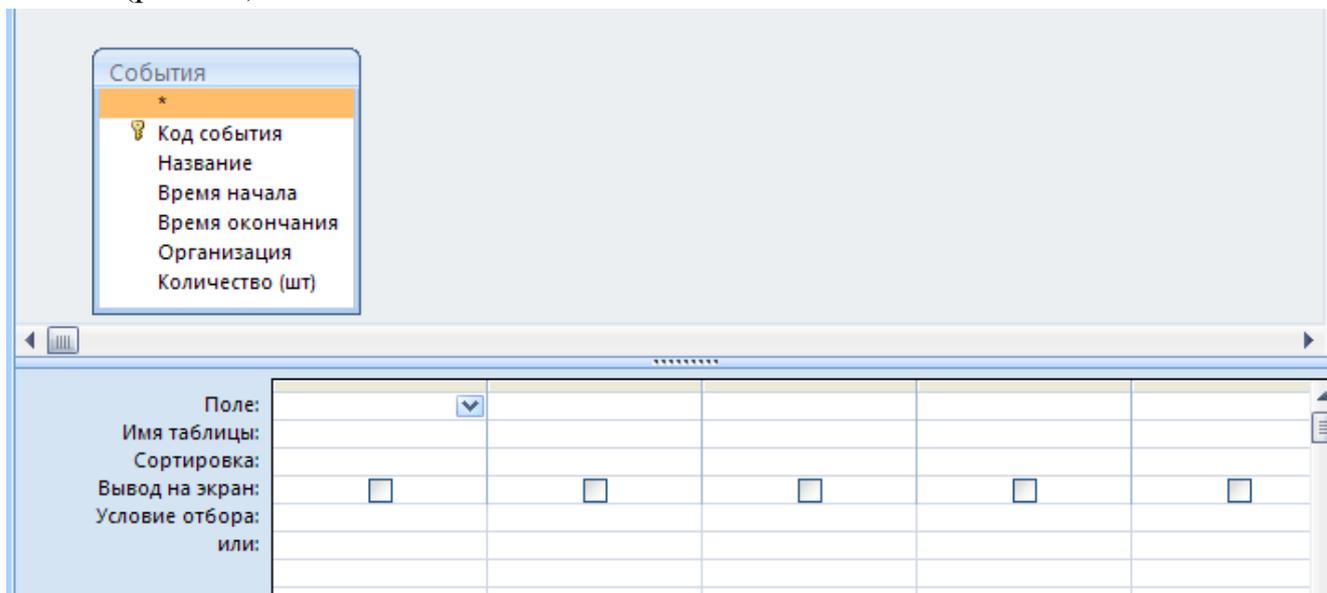


Рисунок 16 – Шаг 6 - Диалоговое окно **Запрос1**

В верхней части содержится окно со списком полей таблицы **События**. Для включения поля в запрос необходимо мышью перетащить имя этого поля из таблицы **События** в строку **Поле** из нижней части диалогового окна или дважды щелкнуть по его имени в этом списке. Это поле появится в нижней части диалогового окна и по умолчанию установится флажок в опции **Вывод на экран**.

6. В ячейке **Сортировка** можно задать способ упорядочения отобранных записей.

7. Аналогичные действия выполнить для других полей таблиц.

8. Ввести критерий отбора в ячейку **Условие отбора**. Для этого щелкнуть по ячейке **Условие отбора** в поле **Название**. Ввести требуемое значение критерия отбора (например, **Отгрузка компьютеров**). Для усложнения критерия запроса необходимо щелкнуть по ячейке **Условие отбора** в каком-либо другом поле, например в поле **Организация** и ввести в него требуемое значение критерия отбора (например, **МИФИ**). Такие же действия можно в случае необходимости провести для других полей. Для создания запроса с применением логики **ИЛИ** необходимо поочередно щелкнуть по ячейке **или** того столбца, где вводилось **Условие отбора**,

например, столбца **Название** и **Организация** и ввести поочерёдно в ячейку этого же столбца значение критерия отбора (например, **Отгрузка компьютеров** и **НИИССУ**) (рис. 17).

9. Чтобы сохранить таблицу надо щелкнуть кнопку **Сохранить** . В окне **Сохранение** набрать имя запроса, например, **Запрос1-ваша фамилия-№ группы** и щелкнуть кнопку **ОК**.

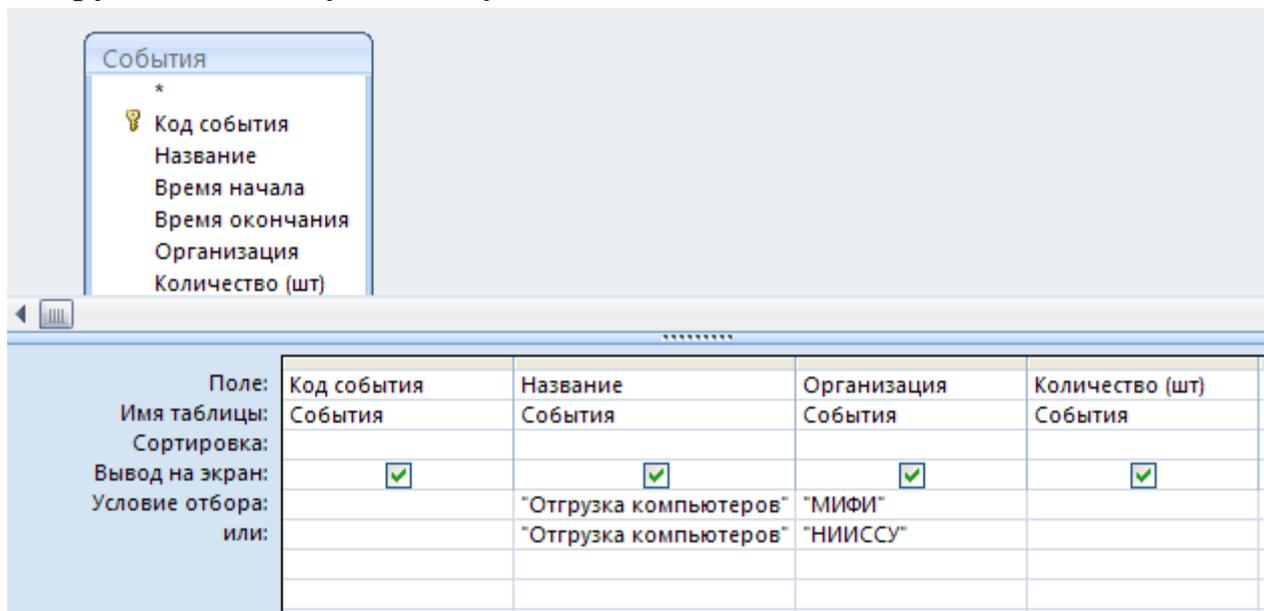


Рисунок 17 – Ввод полей и условий отбора

10. Для просмотра результатов выборки с помощью созданного запроса нажать кнопку **Выполнить**  в группе **Результаты** из вкладки **Конструктор** (рис. 18).

Код события	Название	Организация	Количество (шт)
1	Отгрузка компьютеров	НИИССУ	50
2	Отгрузка компьютеров	МИФИ	40
5	Отгрузка компьютеров	НИИССУ	50
* (№)			

Рисунок 18 – Просмотр результатов выборки.

Чтобы снова просмотреть запрос, дважды щёлкните кнопкой мыши на значке, расположенном в области переходов.

Запрос может отображаться в двух режимах: просмотра данных и конструктора. Переключится в режим конструктора можно, нажав кнопку **Режим** в группе **Режимы** из вкладки **Главная**.

Чтобы вернуться в режим просмотра данных запроса, выберите в меню кнопки **Режим** пункт **Режим таблицы**.

### **Изменение комбинации полей в запросе в режиме Конструктора запросов**

В существующий запрос можно вносить ряд изменений: добавлять или удалять поля из таблицы запроса, включать в запрос поля из других таблиц. Можно также изменять порядок следования полей в запросе (эта операция выполняется аналогично перемещению столбца в таблице данных). Все эти действия реализуются в режиме **Конструктора запросов**.

#### **Добавление поля в запрос**

11. Чтобы просмотреть запрос, дважды щёлкните кнопкой мыши на значке,

расположенном в области переходов. Переключится в режим конструктора, можно нажав кнопку **Режим** в группе **Режимы** из вкладки **Главная**.

12. На экране появится диалоговое окно выбранного запроса (рис. 16).

13. В списке полей, расположенном в верхней части конструктора запросов, дважды щелкнуть по имени поля, которое надо добавить в запрос, или просто перетащить его в нижнюю часть окна — бланк запроса.

#### **Удаление поля из запроса**

14. Выполнять шаги 11, 12.

15. Выделить поле в бланке запроса (в нижней части окна запроса) и нажать клавишу **DEL** на клавиатуре.

### **Практическое задание**

1. Из таблицы базы данных **События-ваша фамилия-№ группы** создайте запрос на выборку записей, содержащих следующие столбцы (поля): **Код события**, **Название**, **Организация** и **Количество (шт)**. В этом запросе укажите сортировку записей по возрастанию в поле **Организация**. Введите в строке **Условие отбора** критерии отбора **Отгрузка компьютеров** для поля **Название** и **МИФИ** для поля **Организация**, а также для создания запроса с применением логики **ИЛИ** в ячейку или столбца **Название** введите критерий отбора **Отгрузка компьютеров** и **Организация** введите критерий отбора **НИИССУ**.

2. Сохраните этот запрос под именем **Запрос1- ваша фамилия-№ группы**.

3. Просмотрите этот запрос с помощью кнопки **Выполнить**.

4. Из таблицы базы данных **События-ваша фамилия-№ группы**

добавьте в этот запрос столбец **Время начала** и **Время окончания**. Просмотрите этот скорректированный запрос с помощью кнопки **Выполнить**.

5. Из таблицы базы данных **Таблица2-ваша фамилия-№ группы** создайте запрос на выборку записей, содержащих следующие столбцы (поля): **№ звонка**, **Код звонка**, **Организация**, **Дата звонка**, **Тема** и **Количество (шт)**. В этом запросе введите в строке **Условие отбора** критерии отбора **НИИССУ** для поля **Организация** и **>05.02.2003** для поля **Дата звонка**, а также для создания запроса с применением логики **ИЛИ** в ячейку **или** для столбца **Организация** введите критерий отбора **МИФИ** и для столбца **Дата звонка** введите критерий отбора **>05.02.2003**.

6. Сохраните этот запрос под именем **Запрос2- ваша фамилия-№ группы**.

7. Просмотрите этот запрос с помощью кнопки **Выполнить**.

### **Контрольные вопросы**

1. Что такое запрос базы данных?
2. Что такое критерии отбора в запросе базы данных?
3. Как удалить поле из запроса?
4. Как добавить поле в запрос?
5. Какое назначение кнопки **Выполнить** в группе **Результаты** из вкладки **Конструктор**?

## Лабораторная работа №6. Работа с запросами с использованием построителя выражений

В предыдущем разделе были рассмотрены вопросы создания запросов и синтез критериев отбора. Основной же целью применения запросов является выборка требуемых записей. Поэтому в этой лабораторной работе основное внимание будет уделено запросам на изменение и применению построителей выражений, позволяющих производить выборку записей, расположенных внутри заданного диапазона, а также исключать определенные записи.

### Работа с построителем выражений

Построитель выражений применяется для задания условий в бланке запроса в виде операторов сравнения. Он состоит из следующих трех частей (рис. 19):

- в верхней части окна **Построителя** расположено поле конструирования выражений. Ниже находится область, предназначенная для создания элементов выражения и их последующей вставки в поле выражения. Допускается непосредственный ввод части выражения в поле выражения;

- в средней части окна находятся кнопки с часто используемыми операторами. При нажатии на одну из этих кнопок **Построитель** вставит соответствующий оператор в текущую позицию поля выражения. Чтобы вывести полный список операторов, нужно выбрать папку **Операторы** в нижнем левом поле и нужный тип в среднем поле. В правом поле будут выведены все операторы выбранного типа;

- в нижней части окна построителя находятся три поля:
  - в левом поле выводятся папки, содержащие таблицы, запросы, формы, объекты базы данных, встроенные и определенные пользователем функции, константы, операторы и общие выражения;

- в среднем поле задаются определенные элементы или типы элементов для папки, заданной в левом поле. Например, если выбрать в левом поле **Встроенные функции**, то в среднем поле появится список всех типов функции **Microsoft Access**;

- в правом поле выводится список значений (если они существуют) для элементов, заданных левым и средним полями. Например, если выбрать в левом поле **Встроенные функции** и тип функции в среднем, то в правом поле будет выведен список всех встроенных функций выбранного типа.

Обычно выражения задают диапазон изменения какого-либо поля. Для построения выражений применяются логические, арифметические операторы, а также операторы сравнения.

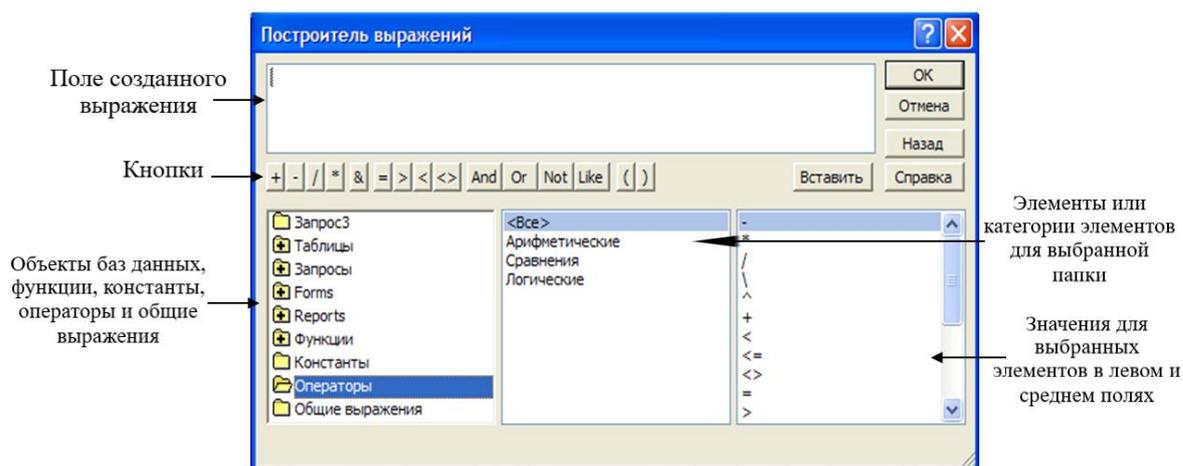


Рисунок 19 – Основные блоки Построителя выражений

Рассмотрим как создаются и применяются запросы на примере выборки данных из таблицы **Таблица 1-ваша фамилия-№ группы**, представленной в вашей базе данных **Контакты- ваша фамилия-№ группы**. Отберем из таблицы базы данных записи, содержащие звонки, сделанные после 7.02.11 и имеющие длительность не менее 10 мин.

Кроме этого рассчитаем стоимость звонков.

### Пример использования Построителя выражений

Процесс создания запроса с формированием критерия на основе применения операций сравнения сводится к следующему.

1. Щелкнуть кнопку **Конструктор запросов** в группе **Другие** на вкладке **Создание**.
2. В диалоговом окне **Добавление таблицы** щелкнуть по таблице **Таблица 1-Ваша фамилия-№ группы** и нажать кнопку **Добавить**, чтобы на ее основе создать запрос и далее нажать кнопку **Заккрыть**.
3. Перейти к таблице создания запроса. Включить в отбор информацию, расположенную в следующих полях " **Код звонка, Дата звонка, Продолжительность звонка (мин) и Стоимость минуты звонка (руб)** " (как это делается, описано в предыдущей лабораторной работе).
4. Установить курсор в ячейку **Условие отбора** поля **Дата звонка**.
5. Щелкнуть по кнопке **Построитель**  в группе **Настройка запроса** из вкладки **Конструктор**.

6. В открывшемся окне **Построитель выражений** нажать кнопку  или клавишу ">" на клавиатуре. Затем ввести значение 7.02.11 и нажать кнопку **ОК**. Сформированное выражение заносится в ячейку.

7. Установить курсор в ячейку **Условие отбора** поля **Продолжительность звонка (мин)** и ввести значение >10, аналогично как в пункте 6.

8. Для расчета стоимости выбранных звонков установить курсор в ячейке свободного столбца справа в строке **Поле**. Щелкнуть кнопку **Построитель**. В открывшемся окне построителя выражений дважды щелкнуть в левом поле нижнего окна кнопку **Таблицы** и далее в открывшемся подменю щелкнуть **Таблица 1-Ваша фамилия-№ группы**.

В открывшемся среднем поле нижней части окна построителя выражений дважды щелкнуть **Продолжительность звонка (мин)**, затем щелкнуть кнопку «\*» в средней части окна и наконец-то дважды щелкнуть в том же среднем поле нижней части окна **Стоимость минуты звонка (руб)** (рис. 20).

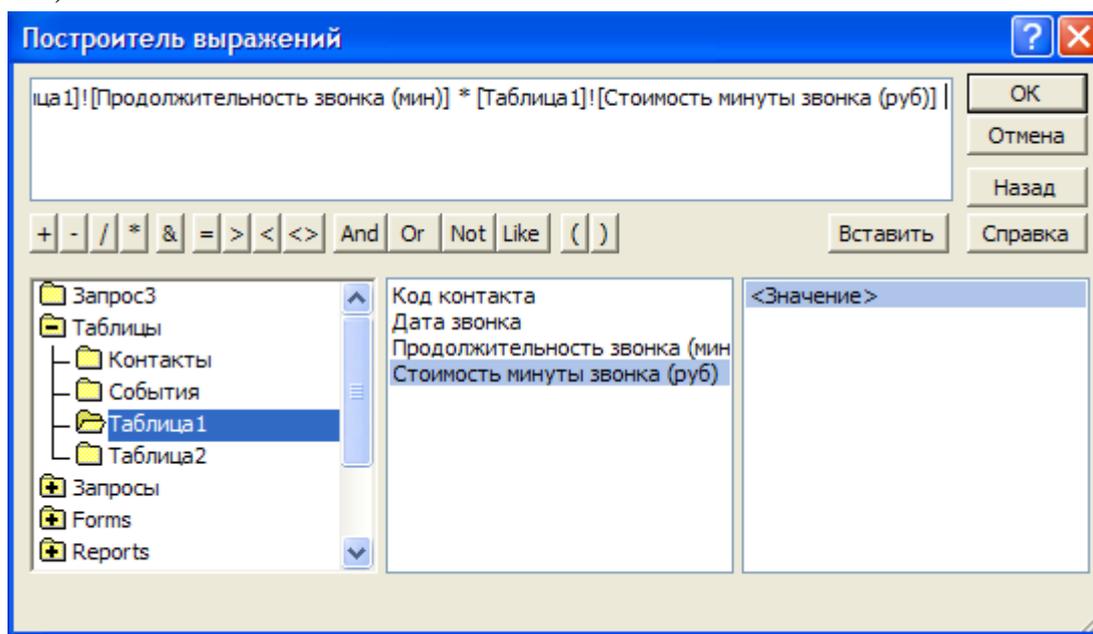


Рисунок 20 – Окно построителя выражений

В верхней части построителя выражений отобразится полученное выражение. Щелкнуть кнопку **ОК**.

В результате, выполненных шагов 1-8, получим бланк запроса, показанный на рис. 21.

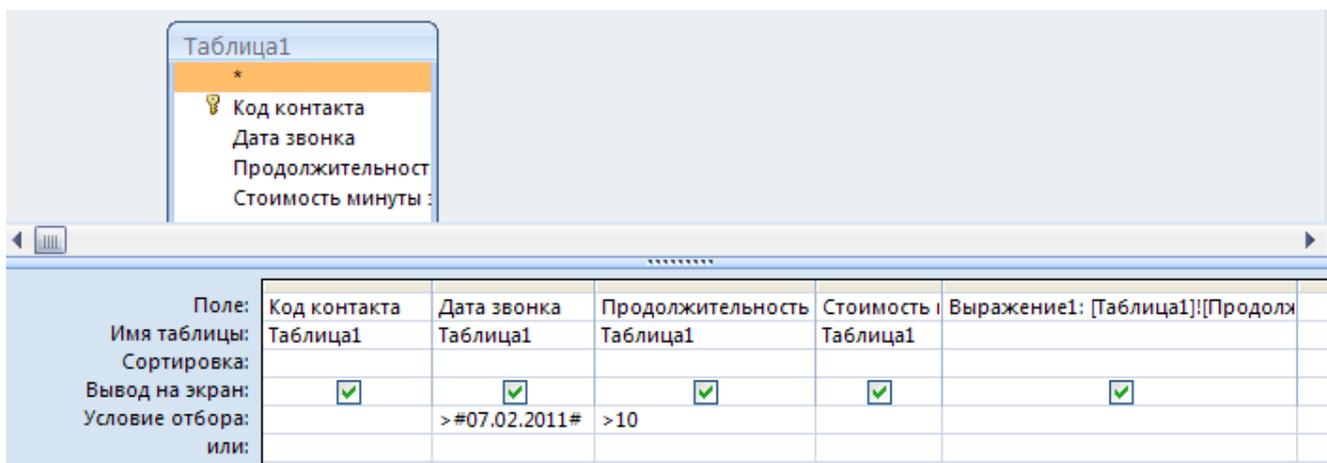


Рисунок 21 – Бланк сформированного запроса в режиме Конструктор

9. Созданное выражение построителем **Выражение1: [Таблица1]...** можно выделить двойным щелчком мыши и переименовать только одно слово **Выражение1** на выражение **Стоимость звонков**.

10. Сохранить таблицу с отобранными записями и рассчитанной стоимостью звонков под именем **Запрос 3-ваша фамилия-№ группы**, используя кнопку **Сохранить** .

11. Для применения созданного запроса и просмотра отобранных записей нажать кнопку **Выполнить**  в группе **Результаты** из вкладки **Конструктор** (рис. 22).

Код контакта	Дата звонка	Продолжительность	Стоимость минут	Стоимость звонков
8	05.03.2011	14	7,00р.	98,00р.
9	07.03.2011	23	5,10р.	117,30р.
10	07.03.2011	21	10,55р.	221,55р.
*	(№)		0,00р.	

Рисунок 22 – Бланк сформированного запроса в режиме таблицы

### Практическое задание

1. Создайте запрос для выборки данных из таблицы **Таблица 1-ваша фамилия-№ группы**, представленной в вашей базе данных **Контакты-ваша фамилия-№ группы**. Отберите из таблицы базы данных записи, содержащие звонки, сделанные после 7.02.11 и имеющие длительность не менее 10 мин. Кроме этого рассчитайте стоимость отобранных звонков в запросе с помощью построителя выражений.

2. Сохраните этот запрос под именем **Запрос3- ваша фамилия-№ группы**.

3. Просмотрите этот запрос с помощью кнопки **Выполнить**.

### **Контрольные вопросы**

1. Из каких трех частей состоит построитель выражений?
2. Какое назначение трех полей нижней части построителя выражений?
3. Какая кнопка во вкладке **Настройка запроса** используется для запуска построителя выражений?

## Лабораторная работа №7. Создание отчетов, сводных таблиц и диаграмм

### Способы создания отчетов

После создания базы данных и отбора из нее информации необходимо распечатать полученные результаты. Самое простое — это скопировать данные в отдельный документ и вывести их на печать. Но такой подход является малоэффективным, так как потребует внесения как редакционных, так и структурных изменений в документ. Тем более это актуально, если в окончательный документ необходимо включить различные диаграммы, результирующие вычисления и т.п. Чтобы сделать эффективной работу по подготовке такого документа, применяют процедуру подготовки отчета. Учитывая все выше сказанное, дадим понятие отчета в **Microsoft Access**.

**Отчет** — объект базы данных **Microsoft Access**, предназначенный для отображения данных, организованных и отформатированных в соответствии со спецификациями пользователя. С помощью отчетов составляются коммерческие сводки, списки телефонов или списки рассылки.

Больше всего сведений в отчете берется из базовой таблицы, запроса или формы, являющихся источниками данных для отчета. Другие данные вводятся в процессе разработки отчета. Рассмотрим способы создания отчетов.

### Создание простого отчета

Данный способ, в основном, применяется для создания простых отчетов.

Процесс создания такого отчета сводится к следующим шагам.

1. В области переходов выделите необходимый объект: таблицу, запрос или форму (например, **Запрос1** или **Запрос2**).

2. Щелкните кнопку **Отчёт** в группе **Отчёты** из вкладки **Создание**. На экране отобразится отчёт в **Режиме макета**, под именем ранее выделенного объекта.

3. Чтобы сохранить отчёт щелкните кнопку **Сохранить** . В окне **Сохранение** наберите имя отчёта и щелкните кнопку **ОК**.

В **Режиме макета** в отчёте можно вводить, используя вкладку **Формат**:

- рисунок (кнопка **Эмблема** );
- дату и время (кнопка **Дата и время** );

- номер страницы (кнопка **Номер страницы** ), а также изменять формат отчёта, воспользовавшись кнопкой **Автоформат** .

Кроме этого во вкладке **Параметры страницы** можно выбрать ориентацию отчёта: **Книжная** или **Альбомная**.

### **Создания отчетов с помощью Мастера отчетов**

Мастер отчетов обеспечивает отбор информации, включаемой в отчет, и задание стиля форматирования, определяющего внешний вид отчета. Данный метод позволяет выбрать отдельные поля, находящиеся в разных таблицах и запросах, и включить их в отчет. Мастер запоминает предыдущие стили оформления, что позволяет пропускать диалоговые окна с указанными параметрами.

Процедура создания отчета состоит из следующих шагов.

1. В области переходов выделите необходимый объект: таблицу, запрос или отчёт (например, **Запрос3**).

2. Щелкните кнопку **Мастер отчётов** в группе **Отчёты** из вкладки **Создание**.

3. В окне **Создание отчётов** из списка **Доступные поля** поочередно выделите поля, включаемые в отчет, и путем двойного щелчка мышью добавьте их в список **Выбранные поля**, например, поля: **Код звонка**, **Дата звонка**, **Продолжительность звонка (мин)** и **Стоимость звонков**.

4. Нажмите кнопку **Далее** в этом окне.

5. Определите способ группировки записей в зависимости от включенных полей, например, **Дата Звонка - по месяцам**. Это обобщающее поле появляется после щелчка мышью по кнопке .

6. Нажмите кнопку **Далее** в этом окне.

7. Определите порядок сортировки данных для каждой группы. Здесь никаких изменений вносить не будем. В этом же окне можно подсчитать итоги и внести их в отчете. Для этого нажмите кнопку **Итоги** и в открывшемся окне задайте итоговые параметры, например, **Sum**, для поля **Продолжительность звонка (мин)** и **Стоимость звонков**. После активизации кнопки **ОК** произойдёт возврат в окно сортировки данных. Нажмите кнопку **Далее**.

8. На экране появится окно задания макета для отчета. Выберите макет отчёта (ступенчатый, блок или структура) и ориентацию – книжную или альбомную и нажмите кнопку **Далее**.

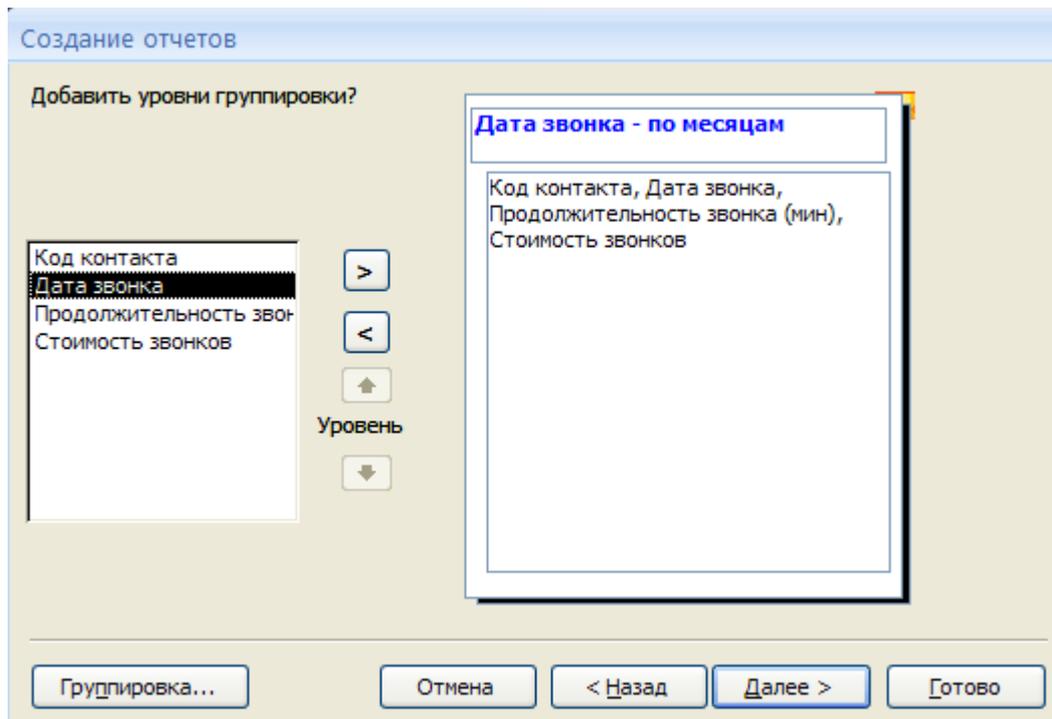


Рисунок 23 – Задание способа группировки данных в отчете

9. Задайте стиль отчета (например, **Поток**) и перейдите к окну, в котором надо указать имя отчета. После этого щелкните по кнопке **Готово**.

Отчет, полученный с помощью мастера отчетов, представляет собой промежуточный документ, который подлежит корректировке и редактированию с целью придания ему окончательного вида (рис.24).

**Запрос3**

Дата звонка - по месяцам	Код контакта	Дата звонка	Продолжительность звонка (мин)	Стоимость звонков
<b>Март 2011</b>				
	8	05.03.2011	14	98,00р.
	10	07.03.2011	21	221,55р.
	9	07.03.2011	23	117,30р.
Итоги для 'Дата звонка' = 07.03.2011 (3 записей)				
Sum			58	436,85
<b>ИТОГО</b>			58	436,85

4 мая 2011 г. Стр. 1 из 1

Рисунок 24 – Отчет, созданный с помощью мастера отчетов (стиль Поток)

### Внесение изменений в отчет

Для внесения изменений и редактирования отчета могут применяться различные операции. Рассмотрим некоторые из них.

**Компоновка отчета.** Отчет должен выглядеть привлекательно и быть легко читаемым. Для улучшения его внешнего вида можно переставлять элементы управления, изменять размер столбцов и расстояние между ними, а также менять их местоположение, удалять отдельные объекты и т.п. Все изменения в отчет вносятся в режиме **Конструктора**. Отчет в режиме **Конструктора** представляет собой набор элементов управления, параметры которых можно изменять вручную. Под элементами управления в **Access** понимают графический элемент, размещаемый в отчете. К ним относят кнопки, поля, флажки, разграничительные линии.

**Изменение расстояния между столбцами отчета.** Для этого необходимо:

1. В режиме **Конструктора** щелкнуть по области данных элемента управления, содержащего имя поля.
2. Поместить на выделенный элемент управления указатель мыши и, дождавшись, когда он превратится в перекрестие со стрелками, перетащить элемент управления в новую позицию.

**Для изменения ширины столбца** необходимо:

1. В режиме **Конструктора** щелкнуть по области данных элемента управления, содержащего имя поля.
2. Поместить указатель мыши на край выделенного объекта, чтобы он принял форму двухсторонней стрелки и перетащить маркер в нужном направлении.

**Изменение формы отчета.** Самый простой способ изменения формы отчета основан на использовании стандартных стилей оформления, предлагаемых **Access**. Для этого применяется команда **Автоформат**  из вкладки **Упорядочить** (режим **Конструктор**).

## Создание сводных таблиц

Общие принципы использования сводных таблиц и диаграмм напоминают работу с этими видами отчетов в **Excel**.

Для создания сводной таблицы сначала откройте таблицу, из которой будете создавать сводную таблицу, например, **Таблица 2-ваша фамилия-№ группы**, а затем воспользуйтесь кнопкой **Режим** в группе **Режимы** из вкладки **Главная**. В меню выберите пункт **Сводная таблица**. Вы увидите окно макета сводной таблицы и список полей таблицы базы данных.

На ленту добавится контекстный инструмент **Работа со сводными таблицами** и его вкладка **Конструктор**.

Структура сводных таблиц в **Access** такая же, как и в **Excel**, — здесь есть области заголовков строк и столбцов, область итогов и область фильтра.

Для создания сводной таблицы переместите поля из списка в соответствующие области. Это можно сделать, перетянув имена полей с помощью мыши (рис. 25). Так, например, в создаваемой сводной таблице необходимо выполнить следующее перетаскивание:

- в поле строк - **Организация**;
- в поле итогов или деталей – **Количество (шт)**;
- поле фильтра – **Тема**.

Для применения фильтра отображения данных воспользуйтесь кнопкой с изображением стрелочки, расположенной рядом с названием полей, размещенных в области строк, столбцов, итогов и фильтра. Например, в названии поля фильтра **Тема** в раскрывавшемся списке установите флажок около опции **Заказ на поставку компьютеров** (рис. 26).

Чтобы превратить базу данных в сводную диаграмму, нажмите кнопку **Режим** и в появившемся меню выберите пункт **Сводная диаграмма**.

Сводная диаграмма состоит из тех же элементов, что и сводная таблица (рис. 28).

Данные, размещенные в сводной диаграмме, соответствуют полям из области данных сводной таблицы.

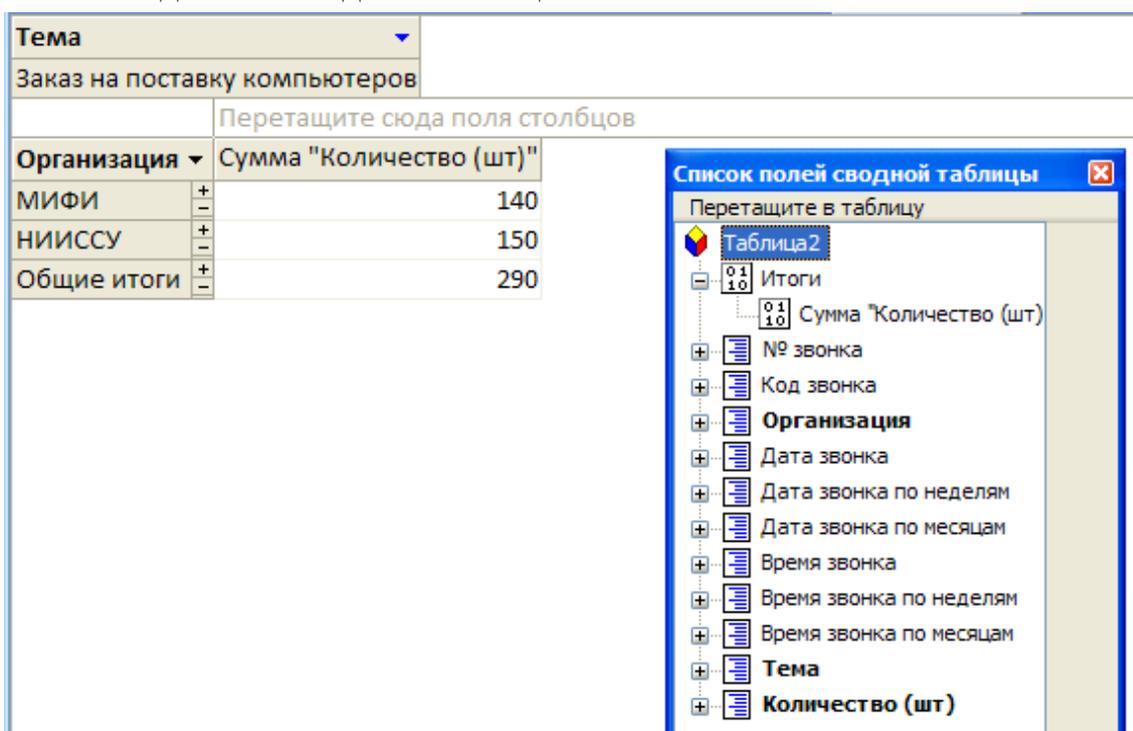


Рисунок 25 – Окно макета сводной таблицы

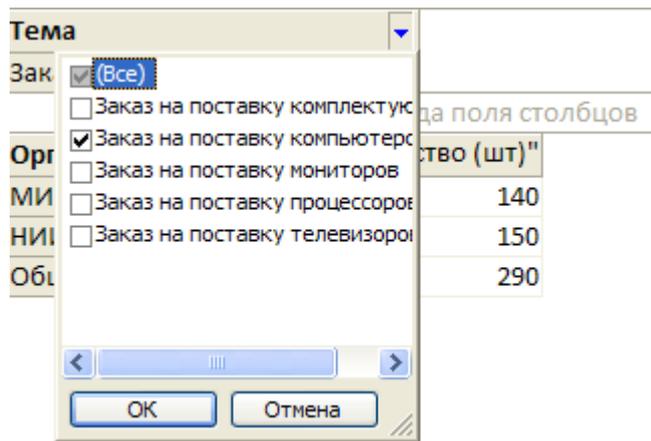


Рисунок 26 – Установка флажка около опции Заказ на поставку компьютеров

### Практические задание

1. Из запросов базы данных **Запрос1- ваша фамилия-№ группы** и **Запрос2- ваша фамилия-№ группы** создайте простые отчеты. В отчете **Запрос2- ваша фамилия-№ группы** введите альбомную ориентацию текста и заголовок отчета введите понравившийся вам рисунок. Сохраните отчеты в базе данных.

2. Из запроса базы данных **Запрос3- ваша фамилия-№ группы** создайте отчет с помощью мастера отчетов. В качестве доступных полей используйте поля: **Код звонка, Дата звонка, Продолжительность звонка (мин)** и **Стоимость звонков**.

Для способа группировки записей в зависимости от включенных полей используйте поле **Дата Звонка - по месяцам**.

Задайте итоговые параметры **Sum** для поля **Продолжительность звонка (мин)** и **Стоимость звонков**. Выберите макет отчёта ступенчатый и ориентацию – книжную.

Используйте стиль отчета **Поток**.

Сохраните отчет в базе данных.

3. Отчет, созданный из запроса базы данных **Запрос3- ваша фамилия-№ группы**, сделайте более привлекательным и читаемым, а также сохраните скорректированный отчет.

Для этого, по вашему усмотрению, выполните следующие действия:

- измените расстояние между столбцами;
- измените ширину столбцов и ширину строк;
- измените форму отчета на **Солнцестояние**.

В итоге вы должны получить отчет, по внешнему виду похожий на отчет, представленный на рис. 27.

## Запрос3

Дата звонка - по месяцам	Код контакта	Дата звонка	Продолжительность звонка (мин)	Стоимость звонков
Март 2011				
	8	05.03.2011	14	98,00р.
	10	07.03.2011	21	221,55р.
	9	07.03.2011	23	117,30р.
Итого для 'Дата звонка' = 07.03.2011 (3 записей)				
Sum			58	436,85
<b>ИТОГО</b>			58	436,85

28 апреля 2011 г. Стр. 1 из 1

Рисунок 27 – Скорректированный отчет Запрос3 (стиль Солнцестояние)

4. Из Таблица 2-ваша фамилия-№ группы создайте макет сводной таблицы, который показан на рис. 25.

Проведите фильтрацию данных по опции **Заказ на поставку компьютеров** (рис.26). Используя данные сводной таблицы постройте сводную диаграмму (тип **Гистограмма**), аналогично той, что представлена на рис. 28.

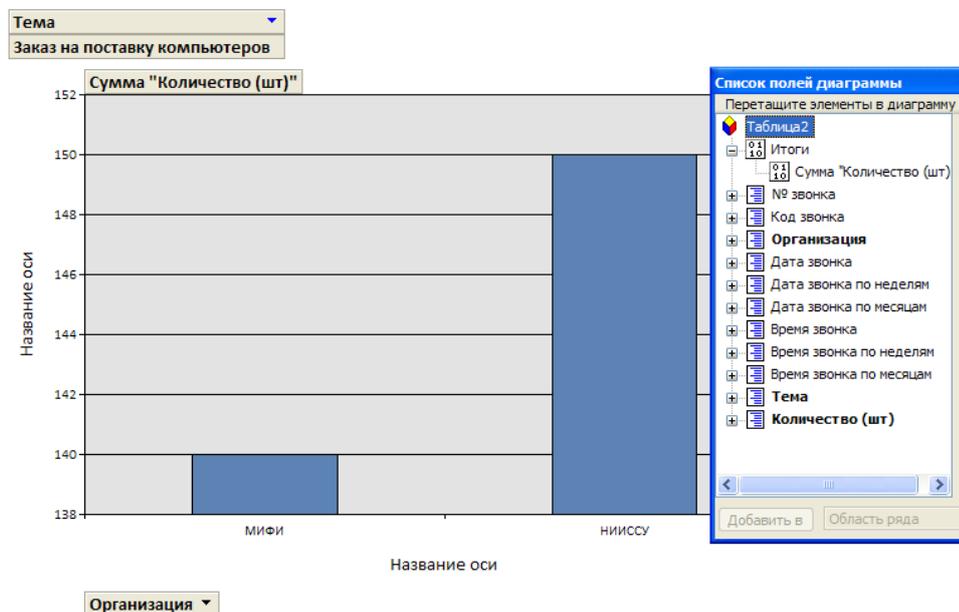


Рисунок 28 – Сводная Гистограмма

### Контрольные вопросы

1. Что представляет собой отчет базы данных **Microsoft Access**?
2. Как изменить расстояние между столбцами отчета?
3. Как изменить ширину столбцов отчета?
4. Как выполнить нумерацию страниц и дату создания отчета?
5. Как изменить форму отчета?

## **Лабораторная работа №8. Создание таблиц, форм и элементов управления в режиме Конструктор для автоматизированной информационной системы**

База данных **Microsoft Access** часто используется для создания специализированных автоматизированных информационных систем (АИС). Для этого в базе данных создаются объекты базы данных, такие как формы, таблицы, запросы, формы и т.д., а затем разрабатывается главная форма АИС, из которой будут вызываться, используя **Макрос**, второстепенные формы, а также вышеприведенные таблицы, запросы, формы и другие объекты. Рассмотрим разработку объектов АИС на примере создания части объектов АИС «Газозаправочная станция» с тем, чтобы получить навыки строить таблицы, формы и элементы управления главной формы в режиме **Конструктор** в базе данных **Microsoft Access**.

### **Разработка таблиц и форм АИС «Газозаправочная станция»**

Для запуска базы данных **Microsoft Access** выполним следующие действия **Пуск – Все программы - Microsoft Office - Microsoft Office Access 2007– Новая база данных**. Далее введём имя создаваемой пустой базы данных: **ГЗС\_ваша фамилия\_№ группы** и щелкнем кнопку **Создать**.

На экране отобразится окно вашей базы данных, для которой необходимо создать объекты этой БД.

Создадим для АИС «Газозаправочная станция» следующие объекты БД:

- 1)таблицы о продаже газа и поставке газа;
- 5)второстепенные формы о продавце ГЗС, поставке газа и сдаче денег инкассатору;
- 4)главную форму ГЗС №1, из которой будут вызываться, используя **Макрос**, второстепенные формы и таблицы.

### **Создание таблиц АИС в режиме Конструктор**

Для создания макета таблиц в режиме **Конструктора** необходимо:

1. Щелкнуть **Конструктор таблиц** в группе **Таблицы** из вкладки **Создание**;

2. В появившейся на экране таблице (таблице 2) описать каждое поле (его имя и тип данных). В качестве типов данных для данных таблиц будут использоваться типы: **Текстовый, Числовой, Дата/время, Денежный** или **Счетчик**.

3. Установить ключевое поле, что сводится к:

- выделению уникального имени поля (например, номер), которое предполагается сделать ключевым;

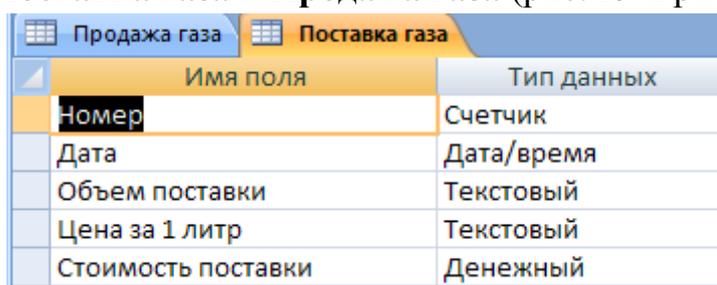
- нажатие кнопки **Ключевое поле**  в группе **Сервис** из вкладки **Конструктор**;

- в результате слева от имени поля появится знак ключа.

4. Для автоматизации процессов ввода/вывода можно организовать выпадающее меню. Для этого нужно выделить соответствующее поле, например, **Стоимость за 1л**, имеющее тип **Текстовый**, и во вкладке **Подстановка** в строке **Источник строк** перечислить данные выпадающего меню о стоимости 1 литра бензина, например, "16";"17";"18" (рис. 30).

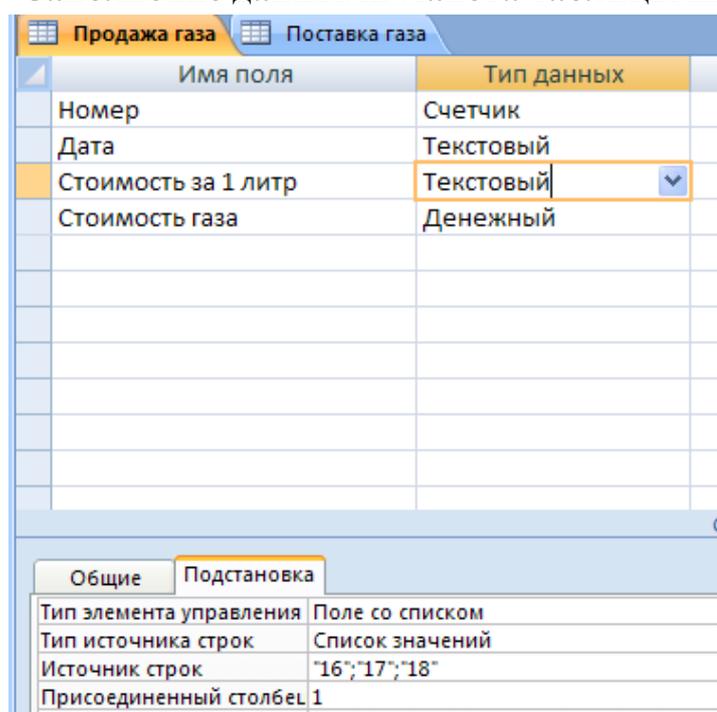
5. Для сохранения таблицы нажать кнопку **Сохранить**  на панели быстрого доступа, а затем в диалоговом окне **Сохранение** ввести имя таблицы, следуя соглашениям об именах объектов **Microsoft Access**, например, **Поставка газа** или **Продажа газа**. Затем щелкнуть кнопку «**ОК**».

В результате вышеприведенных действий поочередно можно создать макеты таблиц **Поставка газа** и **Продажа газа** (рис. 29 и рис. 30).



Имя поля	Тип данных
Номер	Счетчик
Дата	Дата/время
Объем поставки	Текстовый
Цена за 1 литр	Текстовый
Стоимость поставки	Денежный

Рисунок 29 – Заполнение данными макета таблицы **Поставка газа**



Имя поля	Тип данных
Номер	Счетчик
Дата	Текстовый
Стоимость за 1 литр	Текстовый
Стоимость газа	Денежный

Подстановка	
Тип элемента управления	Поле со списком
Тип источника строк	Список значений
Источник строк	"16";"17";"18"
Присоединенный столбец	1

Рисунок 30 – Заполнение данными макета таблицы **Продажа газа**



6. После этого, если щелкнуть кнопку **Режим** в группе **Режимы** из вкладки **Главная** и выбрать **Режим таблицы**, то можно перевести макеты таблиц, например, **Поставка газа** или **Продажа газа**, из режима **Конструктор** в режим таблицы, которые затем можно заполнить данными. Эти данные автоматически занесутся в файл таблицы при их вводе с клавиатуры.

Продажа газа		Поставка газа			
№	Номер	Дата	Объем поставки	Цена за 1 литр	Стоимость поставки
	1	01.03.2007	2000	13	26 000,00р.
	2	01.03.2007	2000	13	26 000,00р.
	3	01.03.2007	1000	13	13 000,00р.
	4	02.03.2007	1000	14	14 000,00р.
	5	02.03.2007	1000	14	14 000,00р.
	6	02.03.2007	1000	14	14 000,00р.
*	(№)				

Рисунок 31 – Заполнение данными таблицы **Поставка газа**

Продажа газа		Поставка газа		
№	Номер	Дата	Стоимость за 1 литр	Стоимость газа
	1	01.03.2007	16	300,00р.
	2	01.03.2007	16	400,00р.
	3	01.03.2007	17	500,00р.
	4	02.03.2007	18	200,00р.
	5	02.03.2007		200,00р.
	6	02.03.2007		100,00р.
*	(№)			

Рисунок 32 – Заполнение данными таблицы **Продажа газа** с выпадающим меню

Для сохранения данных таблиц необходимо нажать кнопку **Сохранить**  на панели быстрого доступа.

Если необходимо снова отредактировать макет таблицы, для этого надо перейти в режим **Конструктор**, используя кнопку **Режим**.

### Создание форм АИС в режиме **Конструктор**

Для вызова формы и панели элементов в режиме конструктора необходимо щелкнуть **Конструктор форм** в группе **Формы** из вкладки **Создание**.

Чтобы создать заголовок на форме щелкнем правой мышкой по форме, выберем команду **Заголовок/примечание формы**, щелкнем по **Заголовку**

**формы** и занесем в заголовок из панели элементов управления требуемые элементы **Надпись** и **Рисунок** (рис. 33), заполним **Надпись** текстом, например **Продавец ГЗС**, а **Рисунок** соответствующим рисунком, как это показано на рис. 34. Изменим в заголовке размер введенного текста и рисунка.

Чтобы занести элемент управления (**Надпись, Поле, Кнопка, Линия** и т. д.) в область данных на форму выполним следующие команды:

1) в режиме **Конструктор** выберем мышью элемент формы на панели элементов управления (рис. 33);

2) разместим его на форме (если размещаем на форме элемент **Кнопка**, то в

диалоговом окне **Создание кнопок**, щелкнем кнопку **Отмена**);

3) изменим размер элемента;

4) заполним элемент данными или названием, если это необходимо.



Рисунок 33 – Управляющие элементы формы

Для сохранения формы нажать кнопку **Сохранить**  на панели быстрого доступа, а затем в диалоговом окне **Сохранение** ввести имя формы, следуя соглашениям об именах объектов **Microsoft Access**, например, **Продавец ГЗС, Поставка газа, Сдача денег** или главная форма **ГЗС №1**. Затем на окне формы щелкнем кнопку «**Закрывать окно**». Чтобы отобразить созданную форму в **Режиме формы**, а не в режиме **Конструктор**,

необходимо щелкнуть кнопку **Режим**  в группе **Режимы** из вкладки **Главная** и выбрать **Режим формы**. Если необходимо форму отредактировать снова, для этого надо перейти в режим **Конструктор**, используя кнопку **Режим**. С помощью этой кнопки возможен и обратный переход.

В результате вышеприведенных действий можно поочередно создать формы (рис.34 – 37).

Продажа газа    Поставка газа    **Продавец ГЗС**    Поставка газа

# Продавец ГЗС



Дата работы       Дата передачи смены

Время начала работы       Время передачи смены

Время окончания работы

Количество газа в начале смены       Количество газа при передаче смены

**Сведения о продавце**      **Сведения о сменщике**

Фамилия       Фамилия

Имя       Имя

Отчество       Отчество

\_\_\_\_\_      \_\_\_\_\_  
подпись      подпись

Рисунок 34 – Форма Продавец ГЗС

Продажа газа    Поставка газа    **Продавец ГЗС**    Поставка газа

# Поставка газа

Дата поставки

Время поставки

Объем поставки (л)

Цена за 1 л

Стоимость поставки

**Сведения о поставщике**

Фамилия

Имя       \_\_\_\_\_

Отчество       подпись

**Сведения о продавце**

Фамилия

Имя       \_\_\_\_\_

Отчество       подпись

Рисунок 35 – Форма Поставка газа

Рисунок 36 – Форма Сдача денег инкассатору

Рисунок 37 – Главная форма ГЗС №1

Теперь остается доработать для главной формы ГЗС №1 элементы управления – кнопки (организовать переходы), по которым будут вызываться второстепенные формы и таблицы, если это потребуется.

Чтобы создать переход для элемента управления - кнопки, используя **Макрос**, выполним следующие действия:

- 1) щелкнем правой мышкой по форме кнопки;
- 2) выберем **Обработка событий – Макросы – ОК**;
- 3) введем имя **Макроса**, если это необходимо, и нажмем **ОК** или без введения имени нажмем кнопку **ОК**;

4) в поле **Макрокоманда** выберем **ОткрытьОбъект**, например, **Форму, Таблицу, Запрос** или **Отчет** (в нашем случае **Форму** или **Таблицу**);

5) в нижней части окна **Макрос** из всплывающего меню выберем имя уже созданного **Объекта** (в нашем случае **Формы** или **Таблицы**);

6) сохраним **Макрос**.

Используя вышеприведенные действия, создадим для формы: **ГЗС №1** элементы управления со всеми кнопками и переходы на соответствующие формы и таблицы (рис. 37). Чтобы отобразить созданную форму в **Режиме формы**, необходимо щелкнуть кнопку **Режим** в группе **Режимы** из вкладки **Главная** и выбрать **Режим формы**. При нажатии кнопок формы **ГЗС №1** будут вызываться соответствующие формы или таблицы (рис. 32-33, 34-36).

В результате будет создана база данных **ГЗС**.

### Практическое задание

1.Создайте пустую базу данных с именем **ГЗС\_Ваша фамилия\_№ группы**.

2. Для этой БД в соответствии с п. 3.2.1. методических указаний создайте макеты таблиц **Поставка газа** и **Продажа газа** в режиме **Конструктор** (рис. 29 и 30), а для столбца **Стоимость** или **Цена за 1л** введите выпадающие меню значений этой стоимости:13, 14, 15 или 16, 17, 18 и заполните эти таблицы данными, как это показано рис. 31 и 32.

3. Создайте второстепенные формы **Продавец ГЗС, Поставка газа** и **Сдача денег инкассатору** (рис. 34-36).

4. Создайте главную форму **ГЗС №1**. Заполните эту форму данными, как это показано на рис. 37.

5. Создайте для элементов **Кнопка** на рис. 37 элемент управления (переход), используя **Макрос**, так чтобы при нажатии кнопок этой формы вызывались соответствующие формы или таблицы (рис. 31-32, 34-36).

6. Откройте главную форму **ГЗС №1** в режиме **Форма** и проверьте правильность работы **АИС «Газозаправочная станция»**.

### Контрольные вопросы

1. Для каких целей используется кнопка **Режим**?
2. Как создать выпадающее меню для автоматизации процессов ввода/вывода данных в ячейку таблицы?
3. Как создать заголовок на форме?
4. Как занести элемент **Надпись, Поле, Кнопка** или **Линия** в область данных на форму?
5. Как создать элемент управления, используя **Макрос**?

# ИНДИВИДУАЛЬНЫЕ ВАРИАНТЫ К ЛАБОРАТОРНЫМ ЗАДАНИЯМ

## Общее задание

1. Создать базу данных в заданной предметной области (по вариантам).
2. Создать формы для заполнения каждой таблицы БД.
3. Создать 2 запроса произвольного содержания.
4. Создать 2 отчета по базе данных.

## Вариант 1

Разработать БД «Абитуриент» для автоматизации работы приемной комиссии ВУЗа. БД должна содержать четыре таблицы: анкеты абитуриентов, данные о специальностях, данные о дисциплинах и результаты экзаменов.

**Анкета** включает следующие данные об абитуриенте:

- регистрационный номер (ключевое поле);
- фамилия, имя, отчество;
- дата рождения;
- оконченное среднее учебное заведение (название, номер, населенный пункт);
- дата окончания учебного заведения;
- наличие красного диплома или золотой/серебряной медали;
- адрес (город, улица, номер дома, телефон);
- шифр специальности.

**Данные о специальностях** содержат:

- шифр специальности (ключевое поле).
- название специальности;

**Данные о дисциплинах** содержат:

- шифр дисциплины (ключевое поле).
- название дисциплины;

**Результаты экзаменов** содержат:

- регистрационный номер абитуриента;
- шифр дисциплины;
- экзаменационная оценка.

## **Вариант 2**

Разработать БД «Зарплата» для автоматизации начисления почасовой заработной платы в бухгалтерии. База данных состоит из трех таблиц, содержащих сведения о работниках, ставки почасовой оплаты и табель отработанных часов.

На каждого работника хранятся следующие данные:

- личный номер (ключевое поле);
- фамилия, имя, отчество;
- отдел;
- должность;
- разряд;

Тарифная сетка для почасовой оплаты:

- должность (ключевое поле вместе с разрядом);
- разряд (от 7 до 15);
- ставка (руб/час).

Табель содержит:

- личный номер;
- месяц;
- количество часов, отработанных за месяц.

## **Вариант 3**

Разработать БД «Оптовая база». На оптовой базе хранятся товары, получаемые от поставщиков. Потребителями являются организации, предварительно направляющие на базу заявки на товар. Отдел снабжения базы ведет учет движения товаров.

БД должна состоять из четырех таблиц: «Склад», «Товары», «Заявки» и «Отпуск товаров». Таблицы имеют следующую структуру:

«Склад»:

- код товара;
- количество;
- дата поступления.

«Товары»:

- код товара (ключевое поле);
- название товара;
- единица измерения;

«Заявки»:

- код заявки (ключевое поле);
- название организации;

- код товара;
  - требуемое количество;
- «Отпуск товаров»:
- код заявки (ключевое поле);
  - код товара;
  - отпущенное количество;
  - дата отпуска товара.

#### **Вариант 4**

Разработать БД «Потребительская корзина» для анализа уровня жизни в семье. Уровень жизни зависит от соотношения доходов семьи и цен на потребляемые продукты.

БД содержит 3 таблицы: «Продукты», «Доходы» и «Потребление». Таблицы имеют следующую структуру:

«Продукты»:

- код продукта (ключевое поле);
- наименование;
- ед. измерения.

«Доходы»:

- год, месяц (ключевое поле);
- совокупный доход за месяц.

«Потребление»:

- год, месяц;
- код продукта;
- количество;
- цена.

#### **Вариант 5**

Разработать БД «Библиотека» для учета хранимой и выданной читателям литературы. БД состоит из трех таблиц со следующей структурой:

«Книги»:

- шифр книги (ключевое поле);
- автор;
- название;
- год издания;
- количество экземпляров.

«Читатели»:

- читательский билет (ключевое поле),

- фамилия и инициалы,
  - отдел (адрес);
- «Выдача»:
- шифр книги;
  - читательский билет;
  - количество экземпляров;
  - дата выдачи;
  - дата возвращения;
  - дата фактического возвращения.

### **Вариант 6**

Разработать БД «ГАИ» для учета нарушений правил дорожного движения водителями. БД состоит из четырех таблиц: «Автомобили», «Водители», «Нарушения» и «Сведения о нарушителях» со следующей структурой:

«Автомобили»:

- марка автомобиля;
- серия и номер технического паспорта (ключевое поле);
- государственный номер;
- номер двигателя;
- номер кузова;
- владелец;
- адрес владельца.

«Водители»:

- фамилия, имя и отчество водителя;
- адрес;
- серия и номер водительского удостоверения (ключевое поле).

«Нарушения»:

- название нарушения;
- шифр нарушения (ключевое поле);
- «Сведения о нарушителях»:
- серия и номер водительского удостоверения;
- гос. номер автомобиля;
- шифр нарушения;
- дата нарушения.

### **Вариант 7**

Разработать БД «Старт» для подсчета результатов соревнований. База данных состоит из трех таблиц:

«Участники»:

- фамилия и инициалы;
- стартовый номер (ключевое поле);
- шифр группы (учитывающий пол и возраст);
- спортивная организация.

«Протокол старта»:

- стартовый номер;
- время старта;
- отметка о не выходе на старт.

«Протокол финиша»:

- стартовый номер;
- время финиша;
- отметка о сходе с дистанции.

### **Вариант 8**

Разработать БД «Перевозки», для транспортного предприятия. На предприятии имеется пакет заявок от других организаций на перевозку различных грузов. БД состоит из трех таблиц: «Транспорт», «Заявки», «Доставка», имеющих следующую структуру:

«Транспорт»:

- марка автомобиля;
- государственный номер (ключевое поле);
- расход топлива (литров на 100 км.).

«Заявки»:

- код заявки (ключевое поле);
- дата;
- пункт отправления;
- пункт назначения;
- название груза;
- единица измерения;
- количество груза.

«Доставка»:

- дата и время отправления,
- дата и время возвращения,
- гос. номер автомобиля,

- код заявки,
- ед. измерения,
- количество фактически перевезенного груза,
- пройденное расстояние.

### **Вариант 9**

Разработать БД «Сессия» для анализа успеваемости на факультете по конкретной специальности. БД состоит из четырех таблиц: «Студенты», «Экзамены», «Зачеты» и «Дисциплины» со следующей структурой:

«Студенты»:

- шифр студента (ключевое поле);
- фамилия, имя, отчество;
- курс;
- группа.

«Экзамены»:

- шифр студента;
- дата;
- шифр дисциплины;
- оценка.

«Зачеты»:

- шифр студента;
- дата;
- шифр дисциплины;
- зачет.

«Дисциплины»:

- шифр дисциплины (ключевое поле);
- название дисциплины.

### **Вариант 10**

Разработать БД «Учебная нагрузка» для учета нагрузки преподавателя ВУЗа и автоматизации отчета о выполнении нагрузки. БД состоит из 3 таблиц со следующей структурой.

Таблица «Дисциплины»:

- код дисциплины (ключевое поле);
- название дисциплины;
- специальность
- курс;

Таблица «Виды нагрузки» (лекции, лабораторные работы, семинары, индивидуальная работа, зачеты, экзамены, прочее):

- тип занятия (ключевое поле);
- название нагрузки.

Таблица «Выполнение нагрузки»:

- дата;
- номер пары;
- номер группы (подгруппы);
- код дисциплины;
- название темы;
- тип занятия (код);
- количество часов.

### **Вариант 11**

Разработать БД «Программное обеспечение» для хранения и обработки сведений о файлах на CD-ROM. БД должна состоять из файлов «CD-ROM», «Владельцы», «Файлы» со следующей структурой:

«CD-ROM»:

- шифр диска (ключевое поле);
- название диска;
- дата выпуска;
- шифр владельца.

«Владельцы»:

- шифр владельца (ключевое поле);
- фамилия, имя, отчество;
- адрес;
- телефон.

«Файлы»:

- название файла (пакета);
- объем в Кбайтах;
- шифр диска;
- пояснения о назначении и свойствах.

## **Вариант 12**

Разработать БД «Классный руководитель». БД содержит три таблицы: «Ученики», «Успеваемость» и «Дисциплины» со следующей структурой.

«Ученики»:

- фамилия, имя;
- шифр ученика (ключевое поле);
- домашний адрес;
- телефон;
- фамилия, имя, отчество родителей.

«Успеваемость»:

- шифр дисциплины;
- шифр ученика;
- оценка;
- пропуск занятия;
- дата.

«Дисциплины»:

- шифр дисциплины (ключевое поле);
- название дисциплины.

## **Вариант 13**

Разработать БД «Администратор гостиницы». БД содержит четыре таблицы: «Гостиничные номера», «Проживание», «Заявки на бронь» и «Информация о местах».

«Гостиничные номера»:

- номер комнаты (ключевое поле);
- количество мест.

«Проживание»:

- фамилия, имя, отчество;
- серия и номер паспорта;
- дата начала проживания;
- дата окончания проживания;
- номер комнаты;
- номер места.

«Заявки на бронь»:

- номер заявки;
- организация или лицо, подавшее заявку;
- заявляемое начало проживания;

- заявляемое окончание проживания;
  - количество заявляемых мест.
- «Информация о местах»:
- номер комнаты (ключевое поле вместе с номером места);
  - номер места;
  - состояние (занято, свободно или забронировано);
  - номер заявки (если забронировано).

#### **Вариант 14**

Разработать БД «Товарооборот» для протомарного магазина. БД состоит из четырех таблиц: «Товары», «Поступление товаров», «Продажа товаров» и «Поставщики» со следующей структурой.

«Товары»:

- код товара (ключевое поле);
- наименование товара;
- единица измерения количества товара.

«Поступление товаров»:

- код товара;
- дата поступления;
- цена приобретения товара за единицу измерения;
- код поставщика.

«Продажа товаров»:

- код товара;
- месяц продажи;
- проданное количество за месяц;
- цена продажи товара.

«Поставщики»:

- код поставщика (ключевое поле);
- название поставщика;
- адрес поставщика;
- телефон поставщика.

## **Вариант 15**

Разработать БД «Промышленность региона». БД состоит из трех таблиц: «Промышленные предприятия», «Уплата налогов», «Прибыль», «Виды налогов» со следующей структурой.

«Промышленные предприятия»:

- код предприятия (ключевое поле);
- название предприятия;
- адрес предприятия;
- форма собственности;
- выпускаемая продукция.

«Уплата налогов»:

- код предприятия;
- код налога;
- сумма налога;
- фактическая дата уплаты;
- планируемая дата уплаты.

«Прибыль»:

- код предприятия;
- сумма прибыли за год.

«Виды налогов» (предусмотреть 3 вида налогов):

- код налога (ключевое поле);
- название налога.

## РАБОТА С ИНФОРМАЦИОННОЙ ПРАВОВОЙ СИСТЕМОЙ КОНСУЛЬТАНТ ПЛЮС. ЗНАКОМСТВО С СИСТЕМОЙ И ИЗУЧЕНИЕ ВОЗМОЖНОСТЕЙ КАРТОЧКИ ПОИСКА

**Цель работы:** приобретение практических навыков работы с информационной правовой системой «Консультант Плюс».

Информационная правовая система (ИПС) Консультант Плюс включает все законодательство РФ: от основополагающих документов до узкоотраслевых актов. Для удобства поиска информации все документы содержатся в Едином информационном массиве. Поскольку документы каждого типа имеют свои специфические особенности, они включаются в соответствующие Разделы информационного массива (рис. 38).

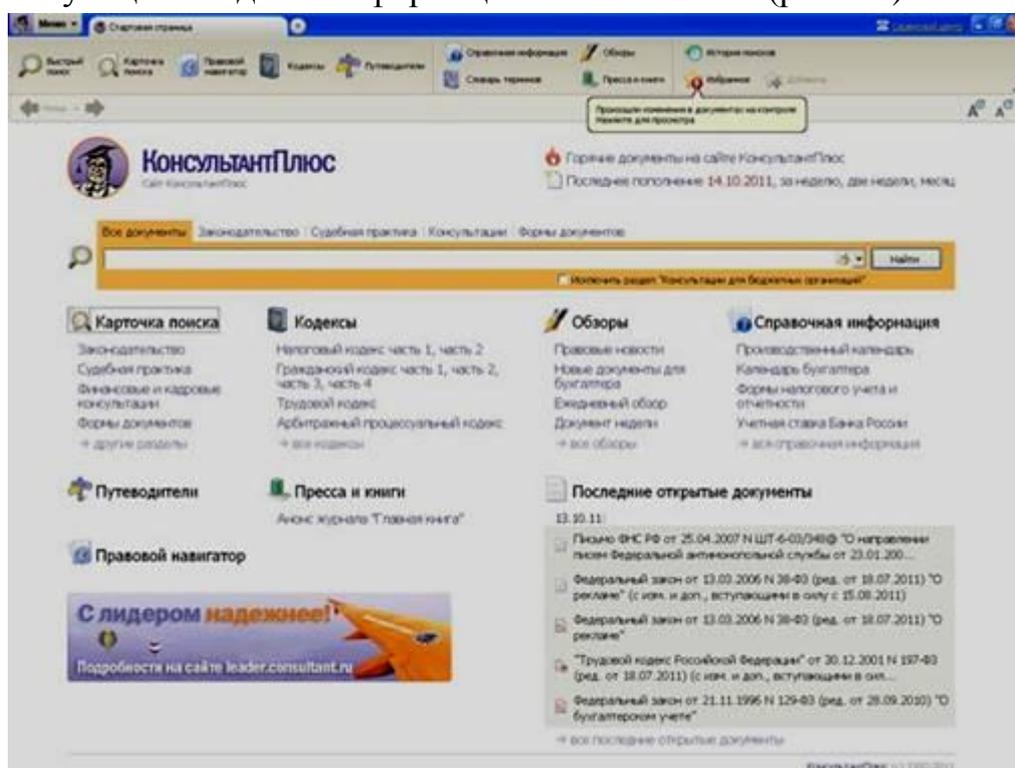


Рисунок 38 – Стартовое окно СПС «Консультант Плюс»

Названия разделов сформулированы таким образом, чтобы можно было легко ориентироваться, какие документы в каком разделе находятся (таблица 8).

Каждый раздел Единого информационного массива, в свою очередь, состоит из Информационных банков. Информационные банки различаются полнотой информации и характером содержащихся в них документов. Это дает возможность при установке «Консультант Плюс» гибко подходить к выбору объема информации, заказывая необходимые информационные банки.

Таблица 8 – Список разделов информационного массива

<b>Законодательство</b>	Нормативные и иные официальные акты федеральных и региональных органов государственной власти.
<b>Судебная практика</b>	Судебные акты. Решения высших судов, Апелляционные суды.
<b>Финансовые и кадровые консультации</b>	Консультационные материалы по бухгалтерскому учету, налогообложению, банковской, инвестиционной, внешнеэкономической деятельности, вопросам валютного регулирования. Путеводитель по кадровым вопросам. Схемы отражения в бухучете финансово-хозяйственных операций (проводки), а также материалы бухгалтерских печатных изданий.
<b>Консультации для бюджетных организаций</b>	"Путеводитель по бюджетному учету и налогам"; "Вопросы-ответы" (бюджетные организации); "Корреспонденция счетов" (бюджетные организации); "Пресса и книги (бюджетные организации)".
<b>Комментарии законодательства</b>	Комментарии к нормативным актам федерального законодательства. Путеводитель по судебной практике (ГК РФ); Путеводитель по корпоративным спорам, Юридическая пресса.
<b>Формы документов</b>	Типовые формы, бланки, образцы деловой документации.
<b>Законопроекты</b>	Проекты федеральных законов, находящиеся на рассмотрении в Федеральном Собрании РФ.
<b>Международные правовые акты</b>	Многосторонние и двусторонние международные договоры Российской Федерации, документы международных организаций, документы о ратификации.
<b>Правовые акты по здравоохранению</b>	Нормативные документы по медицине и фармацевтике, консультации по медицинской и фармацевтической деятельности.
<b>Технические нормы и правила</b>	Строительство

После запуска программы появляется диалоговое окно «КонсультантПлюс», где необходимо сделать выбор средства поиска информации:

1) Карточка поиска, где можно осуществить поиск документов по реквизитам и контексту;

2) Быстрый поиск, который обеспечивает эффективный поиск как отдельных документов, так и подборок документов для решения конкретной проблемы;

3) Правовой навигатор, где можно осуществить поиск информации по необходимой тематике;

4) Кодексы, где размещены кодексы Российской Федерации;

5) Путеводители – актуальные материалы по важным темам;

Структурной единицей Информационного банка системы является документ. Любой документ, кроме непосредственно текста, имеет определенные идентификационные характеристики (реквизиты), которые отличают данный документ от других. Поэтому, чтобы найти необходимые документы из системы, нужно заполнить Карточку поиска.

Карточка поиска – основное средство поиска документов в Информационном банке системы. Она представляет собой таблицу с некоторым количеством поисковых полей. Система ищет документы, одновременно удовлетворяющие всем заполненным полям Карточки поиска. Однако не обязательно заполнять все поисковые поля. Для поиска любого документа достаточно правильно заполнить лишь два-три поля.

При заполнении полей следует обращать внимание на информационную строку внизу Карточки поиска (рис. 39). В ней содержится информация о количестве документов, удовлетворяющих запросу. Если сформированный таким образом список документов будет слишком большим, следует уточнить запрос. Желательно, чтобы количество найденных документов не превышало 30 – 50.

Если же при поиске документа реквизиты его неизвестны или известны приблизительно, то основным средством поиска по конкретному правовому вопросу является поле «Текст документа», где следует задать слова или фразы, которые должны встречаться в тексте этого документа. Если запросу с использованием только данного поля удовлетворяет много документов, то следует его уточнить, используя, в зависимости от имеющейся информации, другие поля Карточки поиска. В системе КонсультантПлюс предусмотрена возможность уточнять полученные списки несколько раз по разным полям.

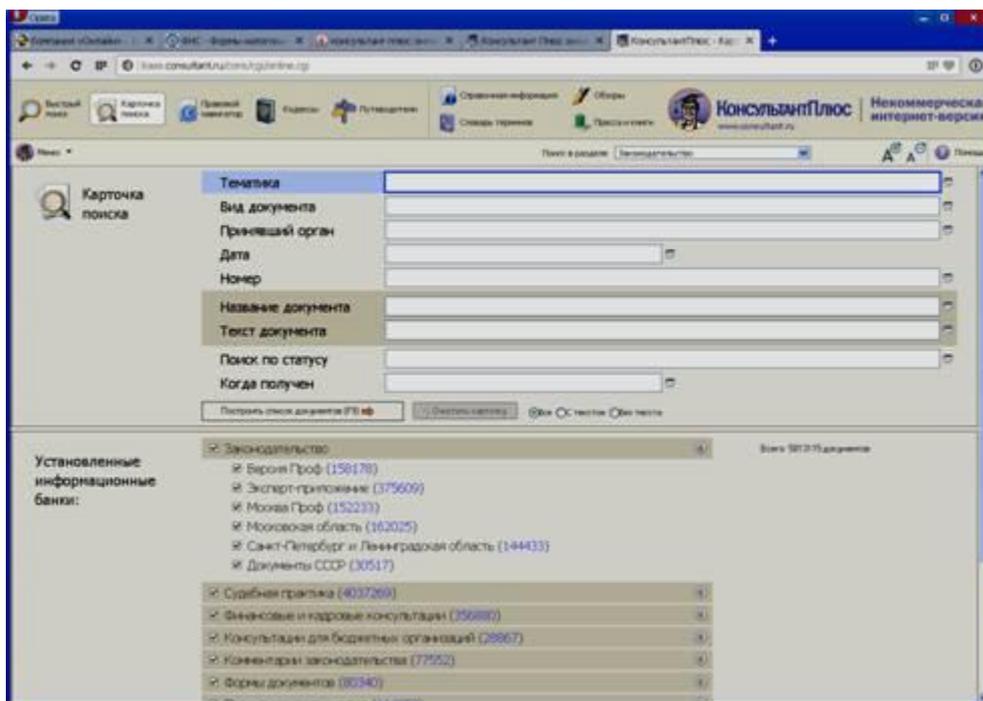


Рисунок 39 – Информационная строка Карточки поиска

### Порядок выполнения работы

1. Изучить теоретическую часть по приведенным выше данным и дополнительной литературе.
2. Запустить с рабочего браузера для доступа в интернет.
3. Набрать адрес [http://www. consultant. ru/online/](http://www.consultant.ru/online/) информационной правовой системы «Консультант Плюс».
4. Ознакомиться со структурой и возможностями Стартового окна информационной правовой системы «Консультант Плюс».
5. Из Стартового окна перейти в раздел «Законодательство». Ознакомиться с общим построением справочно-информационной правовой системы «Консультант Плюс».
6. Изучить поочередно все подпункты основного меню системы. Зайти в «Карточку поиска», рассмотреть все её элементы.
7. Зайти в режим Правового навигатора. Изучить: особенности поиска информации по конкретному правовому вопросу; двухуровневую структуру словаря; ключевые понятия и группы ключевых понятий; различные виды сортировки списка. Выйти из Правового навигатора.
8. Выполнить задание, указанное в табл. 9.
9. Выйти из системы «Консультант Плюс».

Таблица 9 – Задание по практической работе

Вид поиска	Задание
1	2
<p><b>Поиск по номеру и дате документа</b></p>	<p>Найдите Федеральный закон -ФЗ «Об информации, информационных технологиях и о защите информации». Найдите статью, посвященную ограниченному доступу к информации и сохраните её в MS Word.</p>
<p><b>Поиск по виду документа и его названию</b></p>	<p>Найдите Гражданский процессуальный кодекс. Найдите все изменения, внесенные в него в 2013 году. Сколько из них вступило в силу на настоящий момент.</p> <p>Найдите Постановление Правительства РФ «О государственном учете и регистрации баз и банков данных». Где первоначально был опубликован этот документ? Когда последний раз в него были внесены изменения, кем?</p>
<p><b>Поиск по тексту документа</b></p>	<p>Необходимо узнать порядок расчета оплаты труда адвокатов. Найдите последний документ по этому вопросу. Что означают значки   на полях документа?</p> <p>Найдите бланк грузовой таможенной декларации. Сколько вариантов присутствует в системе?. Когда были внесены последние изменения?</p> <p>Переведите действующий вариант в Excel.</p> <p>Организация совершила продажу товара за наличный расчет. В соответствии с законодательством она должна была воспользоваться контрольно-кассовой машиной либо бланками строгой отчетности. Найдите в Кодексе об административных правонарушениях РФ размер штрафа за нарушение данного требования. Как изменялась величина данного штрафа начиная с 2010 года?</p>
<p><b>Поиск правовому навигатору</b></p>	<p>Необходимо определить, чему равен минимальный размер оплаты труда (МРОТ). Найдите последний документ, который внес эти изменения. Определите точки входа в документ, полученный с использованием Правового навигатора.</p> <p>Найдите документы, в которых дается ответ на правовой вопрос: кому предоставляется отсрочка от призыва на военную службу. Какой Федеральный закон регулирует этот вопрос?</p>

1	2
<b>Поиск по принявшему органу</b>	ГТК РФ в 2003 году утвердил форму требования об уплате таможенных платежей. Найдите документ, содержащий необходимую информацию. Постройте дерево связей (ссылок). Сделайте переход из просматриваемого документа в связанные с ним другие документы. Сколько прямых и обратных ссылок существует в полученном документе? Найдите Приказ Генпрокуратуры РФ № 39 «О применении бланков процессуальных документов».
<b>Работа со списком документов</b>	Постройте список документов, которые касаются договора пожизненного содержания с иждивением, которые были написаны после 1 января 2001 года.
<b>Поиск по всем разделам справочной правовой системы</b>	Сформируйте список документов о возможности работы сотрудников в ночное время. Поиск информации проводите по всем разделам справочной правовой системы. Создайте папку «Работа сотрудников» и сохраните в нее только те документы, которые находятся в разделе «Комментарии законодательства». Сформируйте список документов по нормам возмещения командировочных расходов сотрудников и сохраните все документы в папку «Нормы расходов». Поиск проводите по разделу «Законодательство / ВерсияПроф»

### **Порядок оформления результатов**

**После номера и названия каждого задания дайте ответ в следующем порядке:**

- каким разделом справочно-правовой системы вы пользовались для выполнения данного задания;
- какие поля, в какой последовательности и с каким содержанием использовали или какие иные действия, связанные с формированием первичного запроса, предпринимали для выполнения данного задания;
- дальнейшие действия, в том числе уточняющие и иные, необходимые для выполнения данного задания;
- названия найденных документов, их реквизиты и, при необходимости, комментарии к ним;
- если выполнялись действия по переносу документов в другую программу, по установке закладок, копированию, печати и другое, то опишите их.

**Описание каждого своих действий сопровождайте скриншотами.**

## **Контрольные вопросы**

1. Какова структура Единого информационного массива системы Консультант Плюс?
2. Документ как единица ИБ. Что понимается под документом в системе КонсультантПлюс?
3. Как осуществить сортировку законов по дате принятия, названию, дате начала действия и номеру?
4. С какой целью нужна вкладка «Справочная информация», и какие справки она дает?
5. Для чего служит «Правовой навигатор», можно ли его заменить, если да, то чем?
6. Как найти, открыть, изучить и занести в созданную вами папку «Документы Консультант Плюс» документ, составьте алгоритм?
7. Каковы особенности поиска информации по конкретному правовому вопросу?
8. Как найти списки документов, включенных в систему за последний месяц?

## ОСНОВЫ СОЗДАНИЯ HTML-СТРАНИЦ

**Тема: Глобальная компьютерная сеть Интернет. Подключение к Интернету. Практическая работа № 18 "Создание подключения к Интернету. Подключение к Интернету и определение IP - адреса"**

Компьютерные сети принято разделять на локальные (LAN — Local Area Network), региональные (MAN — Metropolitan Area Network) и глобальные (WAN — Wide Area Network).

Глобальная сеть (WAN — Wide Area Network) — это сеть, которая объединяет компьютеры, находящиеся на большом расстоянии.

В физическом смысле такую сеть можно рассматривать как несколько миллионов компьютеров, связанных друг с другом всевозможными линиями связи, однако такой «физический» взгляд на глобальную сеть не отражает современный уровень развития компьютерных коммуникаций. Лучше рассматривать такую сеть как некое информационное пространство, которое мы привыкли называть Интернетом.

В глобальных сетях часто используются уже существующие линии связи, изначально предназначенные совсем для других целей. Например, сети строятся на основе низкоскоростных телефонных кабелей и телеграфных каналов общего назначения. Однако на современном этапе используются радиоканалы, оптоволоконные кабели и другие технологические решения. Надежность функционирования глобальной сети обеспечивает большое количество линий связи между региональными сегментами сети. Например, российский региональный сегмент Интернета имеет несколько магистральных линий связи, соединяющих его с североамериканским, европейским и японским сегментами. В качестве таких «магистральных» линий связи обычно используются оптоволоконные линии с пропускной способностью до 20 Гбит/с и более.

Основу, «каркас» Интернета составляют огромное количество серверов, постоянно подключенных к сети. К серверам Интернета могут подключаться с помощью локальных сетей, по коммутируемым линиям или по беспроводным каналам сотни миллионов пользователей Интернета. Физически магистральные каналы связи проложены не только по поверхности земли, но и под водой.

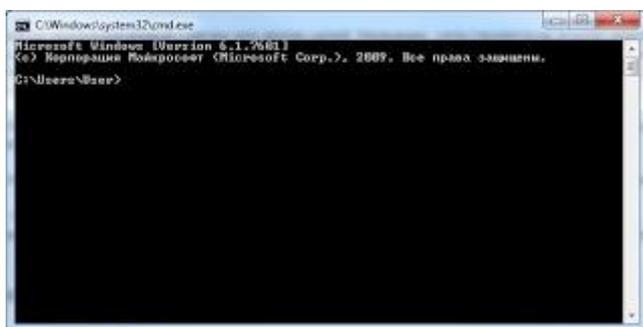
Каждый компьютер, подключённый в сеть, имеет свой уникальный адрес для того, чтобы в процессе обмена информацией компьютеры могли найти друг друга, где бы они ни располагались.

IP-адрес — уникальный числовой адрес, однозначно идентифицирующий узел, группу узлов или сеть.

IP-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел (так называемых «октетов»), разделенных точками — W.X.Y.Z , каждое из которых может принимать значения в диапазоне от 0 до 255, например, 213.128.193.154.

По формуле  $N = 2^k$  можно подсчитать, что общее количество различных IP-адресов составляет более 4 миллиардов.  
 $N = 2^{32} = 4294967296$

Задание 1: *Определить IP рабочего компьютера:*



1. Запустить Командную строку (Пуск - Все программы - Стандартные - Командная строка).
2. В окне сеанса MS-DOS введите команду ipconfig.
3. Узнайте и сохраните IP вашего компьютера.

Пространство адресов в Интернете имеет иерархическую организацию. Для удобства пользователей узлы сети помимо IP-адреса могут иметь мнемонические имена, (мнемоника— это совокупность правил и приемов, имеющих целью облегчить запоминание возможно большего числа сведений, фактов. по которым можно определить принадлежность или функциональное назначение узла). Например, узел, на котором работает Web-сервер, часто именуют «www».

Узлы, как правило, объединяются в логические группы по какому-либо признаку, например, по принадлежности к той или иной организации. Такие группы называются доменами.

Домены могут быть связаны различными физическими линиями. Разделение сети на домены может осуществляться по разным принципам: по функциональному назначению, по территориальному расположению и др. Домены, также как и узлы, имеют имена. Каждому домену присваивается индивидуальное имя. Общие правила построения доменных имен

следующие:

имя может состоять только из букв латинского алфавита, цифр и символа «-»; длина имени не может превышать 63 символов.

Домены могут входить в более крупные домены. Таким образом, формируется иерархическая структура

DNS (Domain Name System) – это система доменных имен. На самом верхнем уровне иерархии располагается корневой домен, который обычно обозначается точкой («.»).

Следующий уровень иерархии составляют три группы доменов верхнего, или первого, уровня (Top Level Domains).

[Домены общего назначения](#)

[Национальные домены](#)

[Домены с именем "агра"](#)

Каждый домен верхнего уровня, как правило, включает в себя домены второго уровня, имена которых выбираются произвольно, например, по имени организации, за которой зарегистрировано это имя. Так, например, в домене первого уровня «ru» имеется домен второго уровня «internet-school», закрепленный за Интернет-школой «Телешкола». Теоретически домен любого уровня может содержать в себе как узлы, так и домены более низкого уровня, однако, на практике домены, уровень которых ниже третьего, встречаются не часто.

Для адресации конкретного узла или домена используется так называемое полностью определенное [доменное имя \(FQDN\)](#).

Рассмотрим, как происходит передача данных между компьютерами в глобальной сети. За такую передачу отвечают, прежде всего, протоколы:

протоколы маршрутизации (*IP (Internet protocol), ISMP, RIP*) – обрабатывают адресацию данных, определяют пути и обеспечивают передачу данных на сетевом уровне;

транспортный протокол TCP (*Transmission control protocol*) – управляет передачей данных на транспортном уровне,

обеспечивающий гарантированную доставку данных.

IP-маршрутизация — процесс выбора пути для передачи пакета в сетях, представляющих собой объединение сетей, базирующихся на разных сетевых технологиях.

Под путем (маршрутом) понимается последовательность маршрутизаторов, через которые проходит пакет по пути к узлу-назначению.

Маршрутизация осуществляется на узле-отправителе в момент отправки IP-пакета, а затем на IP-маршрутизаторах.

Если за доставку информации от компьютера-отправителя к компьютеру-получателю отвечает протокол маршрутизации IP, то транспортный протокол TCP обеспечивает разбиение файлов на IP-пакеты в процессе передачи и сборку файлов в процессе получения.

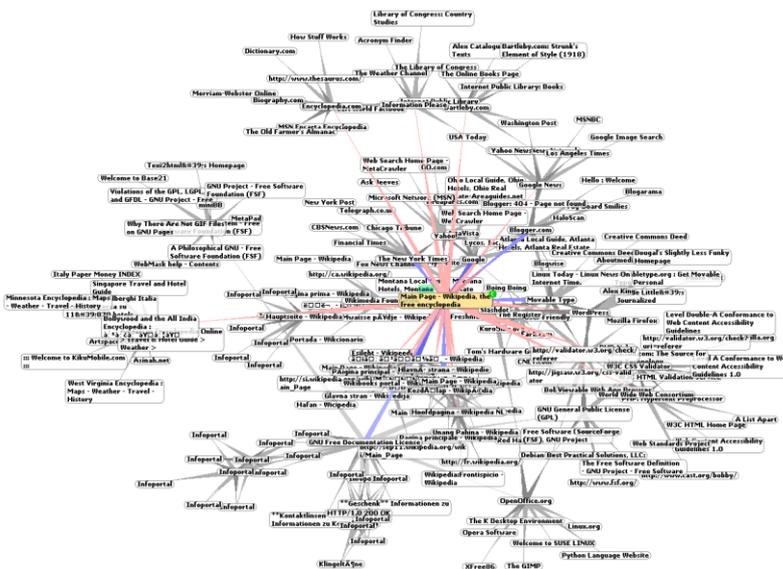
Пакет — обобщенное название блока данных, который передается каким-либо модулем сетевого аппаратно-программного обеспечения по сети как единое целое.

Пакеты формируются абонентской системой-отправителем и направляются через сеть. Пакет состоит из двух частей — заголовка, содержащего служебные данные, необходимые для управления доставкой пакета, и собственно данных, подлежащих передаче. В сети передаются отдельные, независимые пакеты (датаграммы).

### **Технология WWW**

WWW - всемирное хранилище информации, существующее на технической базе сети Интернет.

Технология WWW позволяет создавать ссылки (их называют гиперссылками), которые реализуют переходы не только внутри исходного документа, но и на другой документ, находящийся на данном компьютере, и что самое главное, на любой документ любого компьютера, подключенного в данный момент к сети Интернет. Web-страницы с помощью гиперссылок связаны между собой и образуют Всемирную паутину.



## WWW или Всемирная паутина:

- представляет собой множество информационных ресурсов, организованных в единое целое;
- объединяет многочисленные ресурсы, размещённые в компьютерах по всему миру;
- организована так, что в ней информационные ресурсы представлены не в линейной последовательности, а снабжены гиперссылками, явно указывающие возможные переходы, связи между ресурсами.

WWW содержит информацию самого разного характера:

- самые свежие новости - политические, экономические, культурные, спортивные;
- научную, техническую, образовательную и справочную информацию абсолютно любого рода;
- рекламу разнообразных товаров и услуг;
- ресурсы для досуга и развлечений - книги, музыку, фильмы, игры и многое другое.

Всемирную паутину образуют миллионы веб-серверов сети Интернет, расположенных по всему миру.

Веб-сервер — это компьютерная программа, запускаемая на подключённом к сети компьютере и использующая протокол HTTP для передачи данных.

Для просмотра информации, полученной от веб-сервера, на клиентском компьютере применяется специальная программа — веб-браузер. Основная функция веб-браузера — отображение гипертекста. Всемирная паутина

неразрывно связана с понятиями гипертекста и гиперссылки. Большая часть информации в WEB представляет собой именно гипертекст.

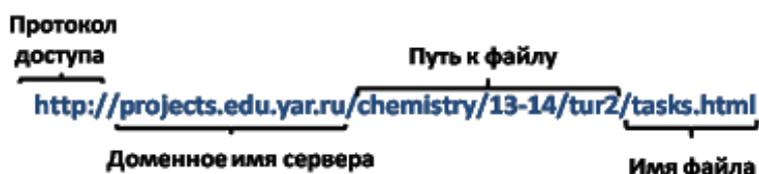
Гипертéкст — термин, введённый Тедом Нельсоном в 1963 году для обозначения текста «ветвящегося или выполняющего действия по запросу».

Гиперссылка состоит из двух частей: указателя ссылки и адресной части ссылки. Указатель ссылки - это то, что мы видим на Web-странице (текст или рисунок), обычно выделенный синим цветом и подчёркиванием. Активизация гиперссылки вызывает переход на другую страницу.

Адресная часть гиперссылки представляет собой URL-адрес (URL - Universal Resource Locator - универсальный указатель ресурсов) документа, на который указывает ссылка.

URL-адрес = протокол доступа + доменное имя ( IP-адрес сервера) + путь к файлу + имя файла Web-страницы

Пример: URL-адрес Web-страницы вопросов II тура Интернет-проекта по химии "Путешествие в мир химии"



### Язык разметки гипертекста

Создание Web-страниц осуществляется с помощью языка разметки гипертекста (Hyper Text Markup Language - HTML). Основа используемой в HTML технологии состоит в том, что в обычный текстовый документ вставляются управляющие символы (тэги), в результате мы получаем текстовый документ, который при просмотре в браузере мы видим в форме Web-страницы. С помощью тэгов можно изменять размер, начертание и цвет символов, фон, определять положение текста на странице, вставлять гиперссылки и т.д.

Web-страница может быть мультимедийной, т.е. содержать ссылки на различные мультимедийные объекты: графические изображения, анимацию, звук и видео.

Интерактивные страницы содержат формы, которые может заполнять посетитель. На форме могут быть различные элементы управления: текстовые поля, списки, переключатели, флажки и др.

Язык разметки гипертекста рассматривает документ как совокупность объектов, свойства которых можно изменять. Это позволяет создавать динамические Web-страницы, т.е. страницы, которые могут меняться уже

после загрузки в браузер. Например, текст можно менять цвет, когда к нему подводится курсор, заголовок - перемещаться и т.п.

#### Flash - технология

Технология Flash основана на использовании векторной графики в формате Shockwave Flash (SWF) разработанная компанией Macromedia®. Это далеко не первый векторный формат, но именно его создателям удалось найти наиболее удачное сочетание между изобразительными возможностями графики, инструментальными средствами для работы с ней, а также механизмом включения результата в Web-страницы. Основным преимуществом Flash технологии является его межплатформенность, то есть этот формат может использоваться на любой аппаратно-программной платформе (в частности, как на компьютерах Macintosh, работающих под управлением операционной системы MacOS, так и на компьютерах IBM с ОС Windows). И еще одна весомая особенность Flash технологии: созданные на его основе изображения могут быть не только анимированы, но еще и дополнены интерактивными элементами и звуковым сопровождением.

Межплатформенность и возможность создания интерактивных мультимедийных приложений, обусловили быстрый рост популярности Flash технологии среди Web-дизайнеров. Поэтому почти одновременно с появлением самого формата фирмой Macromedia® были созданы встраиваемые компоненты (Plug-In) для двух основных браузеров Сети: Internet Explorer и Netscape Communicator. Это и способствовало еще более широкому распространению Flash технологии на просторах всемирной сети Интернет. В результате разработчики этих браузеров, объявили о намерении включить поддержку Flash непосредственно в ядро своих продуктов.

#### Web - сайт

Веб-сайт (англ. Website, от web - паутина и site - «место») - в компьютерной сети объединённая под одним адресом (Доменным именем или IP-адресом) совокупность документов частного лица или организации. По умолчанию подразумевается что сайт располагается в сети Интернет. Все веб-сайты Интернета в совокупности составляют Всемирную паутину. Для прямого доступа клиентов к веб-сайтам на серверах был специально разработан протокол HTTP. Веб-сайты иначе называют Интернет-представительством человека или организации. Когда говорят «своя страничка в Интернет», то подразумевается целый веб-сайт или личная страница в составе чужого сайта. Кроме веб-сайтов в сети Интернет так же доступны WAP-сайты для мобильных телефонов.

Изначально веб-сайты представляли из себя совокупности статических документов. В настоящее время большинству из них свойственна

динамичность и интерактивность. Для таких случаев специалисты используют термин веб-приложение - готовый программный комплекс для решения задач веб-сайта. Веб-приложение входит в состав веб-сайта, но веб-приложение без данных сайтом является только технически.

В большинстве случаев в Интернете одному веб-сайту соответствует одно доменное имя. Именно по доменным именам сайты идентифицируются в глобальной сети. Возможны иные варианты: один сайт на нескольких доменах или несколько сайтов под одним доменом. Обычно несколько доменов используют крупные сайты (веб-порталы) чтобы логически отделить разные виды предоставляемых услуг (mail.google.com, news.google.com, maps.google.com). Нередки и случаи выделения отдельных доменов для разных стран или языков. Например, google.ru и google.fr логически являются сайтом Google на разных языках, но технически это разные сайты. Объединение нескольких сайтов под одним доменом характерно для бесплатных хостингов. Иногда для идентификации сайтов в адресе после указания хоста стоит тильда и имя сайта: example.com/~my-site-name/, иногда используется доменное имя третьего уровня: my-site-name.example.com.

Аппаратные сервера для хранения веб-сайтов называются веб-серверами. Сама услуга хранения называется веб-хостингом. Раньше каждый сайт хранился на своём собственном сервере, но с ростом Интернета технологическим улучшением серверов на одном компьютере стало возможно размещение множества сайтов (виртуальный хостинг). Сейчас сервера для хранения только одного сайта называются выделенными (англ. dedicated).

Один и тот же сайт может быть доступен по разным адресам и хранится на разных серверах. Копия оригинального сайта в таком случае называется зеркалом. Существует так же понятие оффлайновая версия сайта - это копия сайта, которая может быть просмотрена на любом компьютере без подключения к компьютерной сети и использования серверного ПО.

Интернет - портал

Что такое портал? В книге о Гарри Поттере фигурирует магическое устройство с таким названием – оно позволяет героям современной сказки мгновенно переместиться в нужное место.

Что такое интернет-портал? Примерно то же самое – это интернет-сайт, который содержит большое число ссылок на другие сайты Интернета. При помощи портала посетитель может направиться в любом интересующем его направлении. Это удобный интерфейс, который помогает сориентироваться в сети, найти нужную информацию по всему интернету. Помимо

навигационной части интернет-портал имеет оригинальный контент – новости, обзоры, финансовые сводки и сервисную часть, которая включает в себя различные услуги – почту, форумы, информацию о погоде, доски объявлений, голосования, развлечения, и т.п.

В различных интернет-порталах эти части развиты неодинаково. Одни порталы позиционируют себя в основном как [поисковые системы](#), другие – информационные либо сервисные. Но каждый развивает все три направления. Набор предоставляемых порталом услуг зависит от владельца сайта, его возможностей, желания и фантазии. Все это служит одной цели – удовлетворить потребности как можно большего числа потребителей.

Интернет-порталы принято подразделять на горизонтальные и вертикальные.

Горизонтальные порталы, их еще называют универсальные. Ориентированы на максимально широкую аудиторию, предлагают разноплановый контент и имеют большой набор разнообразных сервисов. Как правило, они выстраиваются вокруг поисковых систем.

Вертикальные порталы или порталы-ниши. Это порталы узко тематические. Они направлены на какую-то определенную тематику или сферу деятельности и представляют интерес для пользователей сети по определенным направлениям. Среди таких тематических порталов наиболее распространены финансовые, технологические, развлекательные и религиозные ресурсы, а также это могут быть региональные порталы – сайты какого-нибудь региона, города. Как правило, такие порталы образуют вокруг себя «сообщества» («community») – более-менее постоянную группу людей, систематически общающихся между собой в чате или форуме этого портала.

Существует разновидности интернет-порталов, которые в какой-то степени можно отнести к вертикальным.

Корпоративные порталы - это веб-сайты, которые предназначены для внутреннего пользования сотрудниками какой-либо компании. Они предоставляют доступ сотрудникам к корпоративной информации и к ограниченному количеству внешних веб-сайтов. В отличие от публичных, такие порталы доступны для ограниченного числа пользователей. Примером такого портала может служить сайт про [строительство коттеджей](#).

Государственные порталы - это сайты госструктур, которые постепенно набирают вес, обзаводятся каталогами ресурсов, форумами. Они публикуют новости, экономические или политические обзоры в рамках своей

специфики.

Информационные порталы - обеспечивают информационное обслуживание пользователей в определенном направлении (новости, законодательство, образование). Обновление информации на них происходит в реальном времени.

Можно выделить еще порталы общего назначения – они объединяют несколько тем и ориентированы на широкую аудиторию и смешанные порталы - они сочетают в себе функции электронной торговли и справочных сервисов.

В последнее время происходит некое размывание границы между понятиями портал и сайт. В самом деле, мы говорим, что «портал – это сайт, который...». Давайте попробуем разобраться, чем же отличается интернет-портал от интернет-сайта? Если сайт большой, у него разветвленная внутренняя структура и большое количество ссылок, может ли он называться порталом?

### Браузеры

Для путешествий по Всемирной паутине используются специальные программы - браузеры. Браузер загружает Web-страницу и отображает её в соответствии с тэгами языка разметки гипертекста (HTML). В настоящее время известно достаточно большое количество браузеров (Internet Explorer, Mozilla Firefox, Google Chrome, Chromium, Amigo и др.

После загрузки в браузер начальной (домашней) страницы для путешествия по Всемирной паутине можно приступить различными способами:

- воспользоваться ссылками загруженной в браузер Web-страницы;
- в строку Адрес ввести адрес (URL) Web-страницы;
- воспользоваться "закладками" Web-страниц.

### Сохранение Web - страниц

В процессе просмотра в браузере Web-страниц их копии и связанные с ними мультимедийные файлы автоматически сохраняются в кэш-памяти локального компьютера. Пользователь может установить объём кэш-памяти и место её размещения в определённой папке на жёстком диске. В случае нехватки места на жёстком диске кэш-память можно очистить.

Загрузка копий Web-страниц из кэш-памяти локального компьютера существенно ускоряет их просмотр, однако в этом случае может возникнуть

ситуация, когда Web-страница в Интернете изменилась, а мы будем просматривать её устаревшую копию.

Найти нужные и интересные Web-страницы в кэш-памяти довольно трудно, поэтому полезно специально сохранять их в определённой папке на локальном компьютере. Можно выбрать различные варианты сохранения Web-страниц:

- сохранение страницы в формате Web-страница (только HTML) приведёт к сохранению самой страницы, но при этом не сохраняются связанные с ней рисунки, звуковые и прочие файлы;
- сохранение в формате TXT приведёт к сохранению самой страницы в текстовом формате;
- сохранение страницы в формате Web-страница полностью приведёт к сохранению не только самой страницы, но и связанных с ней рисунков, звуковых и прочих файлов в отдельной папке.

Можно сохранить не только Web-страницу полностью, но и отдельные её части: текст, изображения или ссылки.

Подготовьтесь ответить на следующие вопросы:

1. Из каких двух частей состоит гиперссылка? Какую функцию выполняет каждая из них?
2. Какие части включает URL-адрес документа в Интернете?
3. В чём состоит различие между языком разметки гипертекста и языками программирования?
4. Всемирная паутина состоит из Web-страниц? Web-сайтов? Интернет-порталов?

## Лабораторная работа №9. Создание простейшего HTML-документа. Форматирование текста

Рассмотрим последовательность создания и редактирования простейшего HTML-документа.

### Задание 1

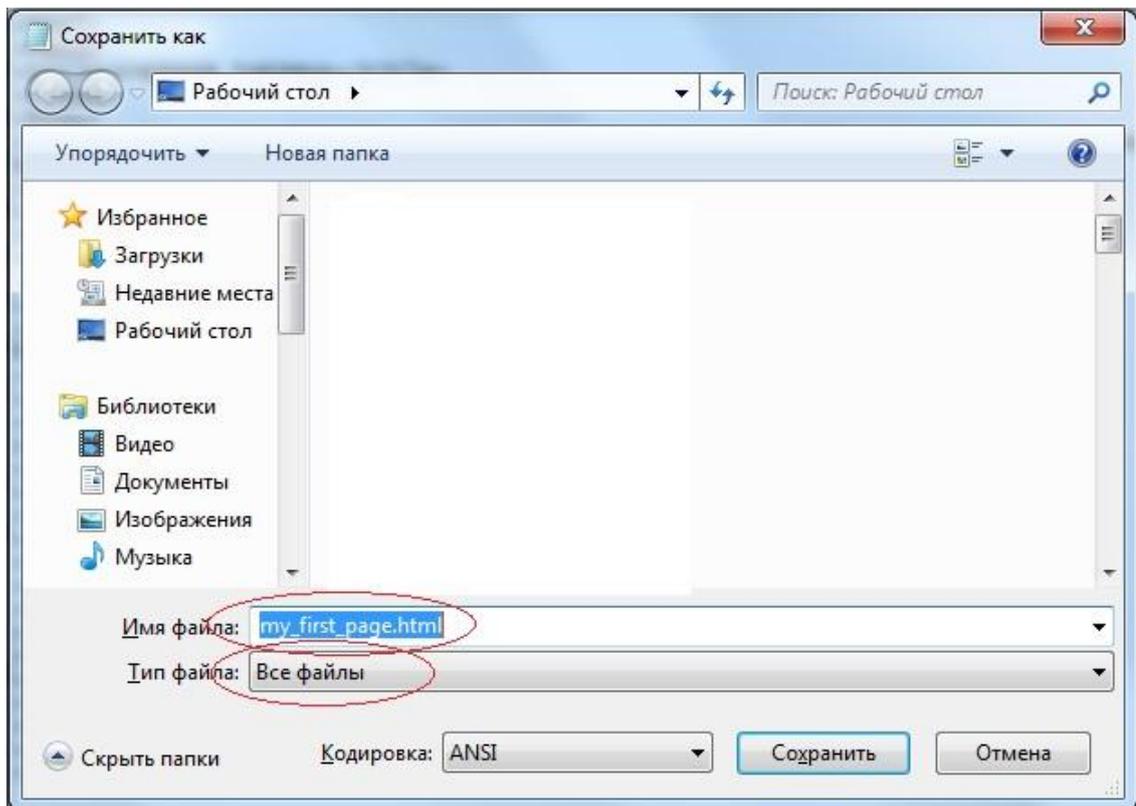
1. Откройте текстовый редактор Блокнот (меню Пуск→Все программы→Стандартные).
2. В окне текстового редактора введите следующий текст:

```
<html>
<head>
<title>Моя первая страница</title>
</head> <body>
<p>Привет!
</body>
</html>
```

Это пример простейшей структуры html-документа.

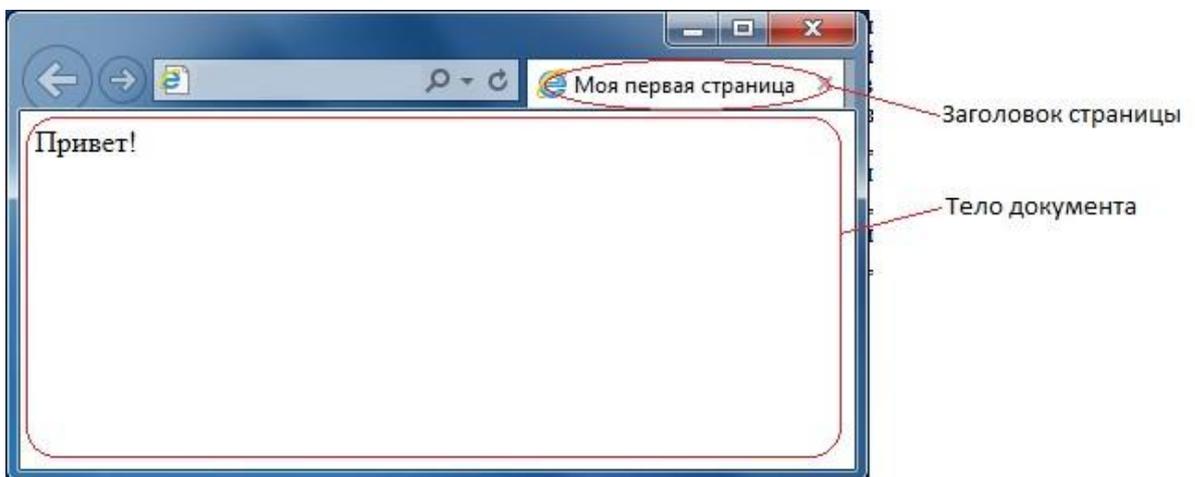
Первый тег в документе - **<HTML>**, сообщает браузеру о том, что он имеет дело именно с документом в формате HTML. Тег **<HTML>** и парный ему закрывающий тег **</HTML>** можно считать, как бы "конвертом", в который помещается весь документ. Любой HTML-документ состоит из заголовка, который задается при помощи тега **<TITLE>**, и тела документа, который определяется тегом **<BODY>**. В заголовке документа размещается служебная информация, комментарии автора и заголовок страницы, заключаемый в теги **<TITLE>**. Заголовок, вписанный между тегами **<TITLE>**, в основное текстовое поле браузеру не попадает, а, как правило, размещается в заголовке окна браузера. Тег **<P>** обозначает абзац в тексте.

3. Сохраните документ, используя пункт "Сохранить как..." в меню "Файл". Укажите тип файла – "Все файлы" и дайте ему название, например, `my_first_page.html`. Обратите внимание, что имя файла обязательно должно заканчиваться расширением `.html` или `.htm`.



4. Откройте полученный HTML-документ в любом интернет-браузере, например, Internet Explorer. Документ при этом будет иметь приблизительно следующий вид:

5.



6. Не закрывая браузер, отредактируйте документ в Блокноте, внося в нем следующие изменения:

- поменяйте заголовок страницы:

```
<title>Страница приветствия</title>
```

- после строки

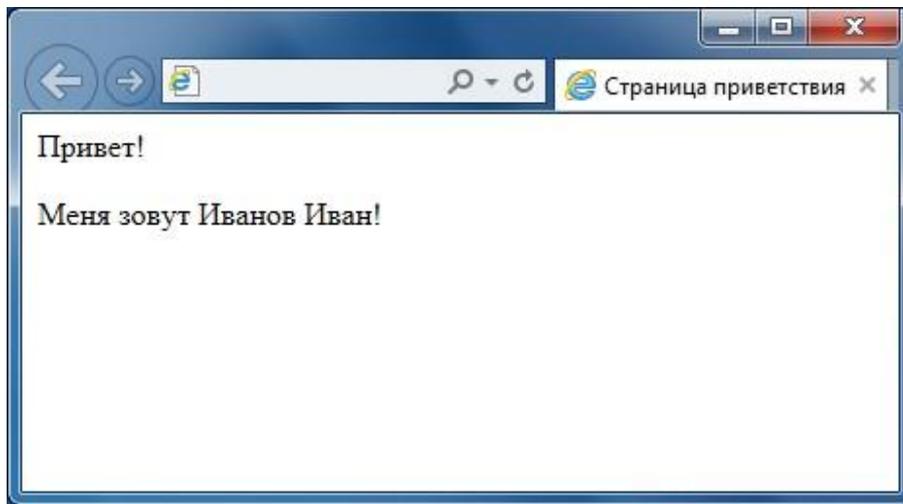
```
<p>Привет!
```

добавьте строку:

```
<p>Меня зовут Иванов Иван!
```

при заполнении используйте свои имя и фамилию.

6. Сохраните документ (Файл □ Сохранить).
7. Для отображения в браузере изменений документа, нажмите кнопку F5 на клавиатуре.



При необходимости можно просмотреть HTML-код в самом браузере.

Большинство элементов языка HTML описывает части содержания документа и помещается между тегами **<BODY>** и **</BODY>**, то есть, внутри структурного элемента BODY. Такие элементы делят на блочные и текстовые. Блочные элементы относятся к частям текста уровня абзаца. Текстовые элементы описывают свойства отдельных фраз и еще более мелких частей текста. Теперь можно сформулировать правила вложения элементов:

- элементы не должны пересекаться (т.е. если открывающий тег располагается внутри элемента, то и соответствующий закрывающий тег должен располагаться внутри этого же элемента);
- блочные элементы могут содержать вложенные блочные и текстовые элементы;
- текстовые элементы могут содержать вложенные текстовые элементы;

- текстовые элементы не могут содержать вложенные блочные элементы.

Строго говоря, все правила языка HTML можно рассматривать исключительно как "пожелания". Средство, используемое для отображения

Web-документа, сделает все возможное, чтобы истолковать разметку наиболее разумным образом. Тем не менее, гарантию правильного воспроизведения документа дает только неукоснительное следование требованиям спецификации языка.

### **Функциональные блочные элементы**

В большинстве документов основными функциональными элементами являются заголовки и абзацы. Язык HTML поддерживает шесть уровней заголовков. Они задаются при помощи парных тегов от `<H1>` до `<H6>`. При отображении Web-документа на экране компьютера эти элементы показываются при помощи шрифтов разного размера.

Обычные абзацы задаются с помощью парного тега `<P>`. Язык HTML не содержит средств для создания абзацного отступа ("красной строки"), поэтому при отображении на экране компьютера абзацы разделяются пустой строкой.

Закрывающий тег `</P>` рассматривается как необязательный.

Подразумевается, что он стоит перед тегом, который задает начало очередного абзаца документа. Например:

```
<h1>Заголовок</h1>  
<p>Первый абзац  
<p>Второй абзац  
<h2>Заголовок второго уровня</h2>
```

## **Заголовок**

Первый абзац

Второй абзац

## **Заголовок второго уровня**

Следствием наличия специального тега, определяющего абзац, является тот факт, что обычного символа конца строки, вводимого по нажатию клавиши ENTER, для создания абзацного отступа недостаточно. Язык HTML рассматривает символы конца строки и пробелы особым образом. Любая последовательность, состоящая только из пробелов и символов конца строки, при отображении документа рассматривается как

одиночный пробел. Это, в частности, означает, что символ конца строки даже не осуществляет перехода на новую строку (для этой цели используется текстовый элемент, задаваемый непарным тегом **<BR>**).

### Текстовое оформление страниц

Для изменения параметров шрифта можно использовать тег **<font>**. Для тега используются следующие атрибуты: **face**, **size** и **color**.

Атрибут **Face** служит для задания гарнитуры шрифтов, использующихся для текста. Названий шрифтов можно указать несколько, через запятую. В этом случае, если первый указанный шрифт не будет найден, будет использоваться следующий по списку.

#### Пример 1. Использование атрибута **face**

```
<font face="Arial, Helvetica, sans-serif">Текст будет написан  
шрифтом  
Arial.</font>
```

**Size** задает размер шрифта в условных единицах от 1 до 7. Средний размер, используемый по умолчанию принят 3. Размер шрифта можно указывать как абсолютной величиной (например, `size=4`), так и относительной (например, `size=+1`, `size=-1`). В последнем случае размер изменяется относительно базового.

#### Пример 2. Задание размера шрифта

```
<font size=1>Шрифт размера 1</font><br>  
<font size=2>Шрифт размера 2</font><br>  
<font size=3>Шрифт размера 3</font><br>  
<font size=4>Шрифт размера 4</font><br>  
<font size=5>Шрифт размера 5</font><br>  
<font size=6>Шрифт размера 6</font><br> <font size=7>Шрифт  
размера 7</font><br>
```

Шрифт размера 1

Шрифт размера 2

Шрифт размера 3

Шрифт размера 4

Шрифт размера 5

Шрифт размера 6

Шрифт размера 7

**Color** определяет цвет текста, который можно задавать с помощью названий цветов или цифровыми значениями в шестнадцатеричном формате. Существует 16 основных цветов, представленных в таблице.

Цвет	Символическое название	Шестнадцатеричное значение
Морской волны	Aqua	#00FFFF
Черный	Black	#000000
Голубой	Blue	#0000FF
Малиновый	Fuchsia	#FF00FF
Серый	Gray	#808080
Зеленый	Green	#008000
Ярко-зеленый	Lime	#00FF00
Темно-красный	Maroon	#800000
Темно-синий	Navy	#000080
Оливковый	Olive	#808000
Пурпурный	Purple	#800080
Красный	Red	#FF0000
Серебряный	Silver	#C0C0C0
Темной морской волны	Teal	#008080
Белый	White	#FFFFFF
Желтый	Yellow	#FFFF00

Кроме того, на сайте [http://www.puzzleweb.ru/html/colors\\_html.php](http://www.puzzleweb.ru/html/colors_html.php) (и других подобных) имеется расширенная таблица из 140 цветов и удобное средство для их подбора в шестнадцатеричном формате.

**Пример 3. Изменение цвета текста**

```
<font size=5 color=red face=Arial>П</font>ервая буква этого предложения будет написана шрифтом Arial, красным цветом и увеличенной.</font>
```

**П**ервая буква этого предложения будет написана шрифтом Arial, красным цветом и увеличенной.

Видоизменение текста - средства его форматирования, такие как выбор начертания шрифта и использование эффектов, позволяющих менять вид текста. В таблице перечислены основные теги, которые применяются для изменения оформления текста.

КодHTML	Описание	Пример
<b>Текст</b>	Жирный текст	<b>Текст</b>
<i>Текст</i>	Курсивное начертание текста	<i>Текст</i>
<u>Текст</u>	Подчеркнутый текст	<u>Текст</u>
<sup>Текст</sup>	Верхний индекс	$e=mc^2$
<sub>Текст</sub>	Нижний индекс	H <sub>2</sub> O
<strike>Текст</strike>	Зачеркнутый текст	<del>Текст</del>

<pre>Т е к с т</pre>	Текст пишется как есть, включая все пробелы	Т е к с т
<em>Текст</em>	Курсивный текст	<i>Текст</i>
<strong>Текст</strong>	Жирный текст	<b>Текст</b>

Обычно для создания верхнего или нижнего индекса используется тег **small**, делающий индекс меньше по размеру основного шрифта.

#### Пример 4. Создание нижнего индекса

```
<b>Формула серной кислоты:</b>
<i>H<sub><small>2</small></sub>SO<sub><small>4</small></sub></i>
>
```

**Формула серной кислоты:**



#### Выравнивание текста

Выравнивание текста определяет его внешний вид и ориентацию краев абзаца и может выполняться по левому, правому краю, по центру или по ширине.

КодHTML	Описание	Пример
<p>Текст</p>	Добавляет новый параграф, по умолчанию выровненный по левому краю. Перед параграфом автоматически добавляется пустая строка.	Текст
<p align=left>Текст</p>	Выравнивание по левому краю.	Текст
<p align=right>Текст</p>	Выравнивание по правому краю.	Текст
<p align=center>Текст</p>	Выравнивание по центру.	Текст
<p align=justify>Текст</p>	Выравнивание по ширине.	Текст по ширине

## Задание 2

1. Поместите на HTML-страницу следующий код:

```

<html>
<body>
<font size="3" face="Times, Palatino, serif" color="#FF0000">Это
пример
№ 1</font> <br>
<font size="2" face="Arial, Helvetica, nonserif" color="#008000">Это
пример № 1</font> <br>
<font size="5" face="Verdana, Arial, Courier, Times"
color="#FF8000">Это пример№ 1</font>
</body>

</html>

```

2. Загрузите страницу в браузер и посмотрите, что получилось.

Это пример № 1  
Это пример № 1  
Это пример № 1

3. Как правило, шрифты типа serif используются для текстов, а шрифты типа nonserif используются для заголовков. Однако, если размер шрифта небольшой тексты serif могут оказаться неразборчивыми. Поэтому старайтесь не использовать минимальные размеры шрифтов. Для

упражнения поместите на страницу текст с разными размерами шрифта serif и оцените возможность их легкого прочтения.

4. Иногда для заглавных букв используют размер шрифта, отличный от размера шрифта для остального текста. Поместите на страницу следующую конструкцию и посмотрите в браузере, что получилось:

```
<font size="5">Д</font>обро пожаловать на мою страницу!
```

Добро пожаловать на мою страницу!

5. Для заголовков часто используют специальные тэги заголовков: <h1>, <h2>, ...<h6>.

Используйте следующие примеры для отображения заголовков:

```
<h1>Заголовок 1</h1>  
<h2> Заголовок 2</h2>  
<h6> Заголовок 6</h6>
```

# Заголовок 1

## Заголовок 2

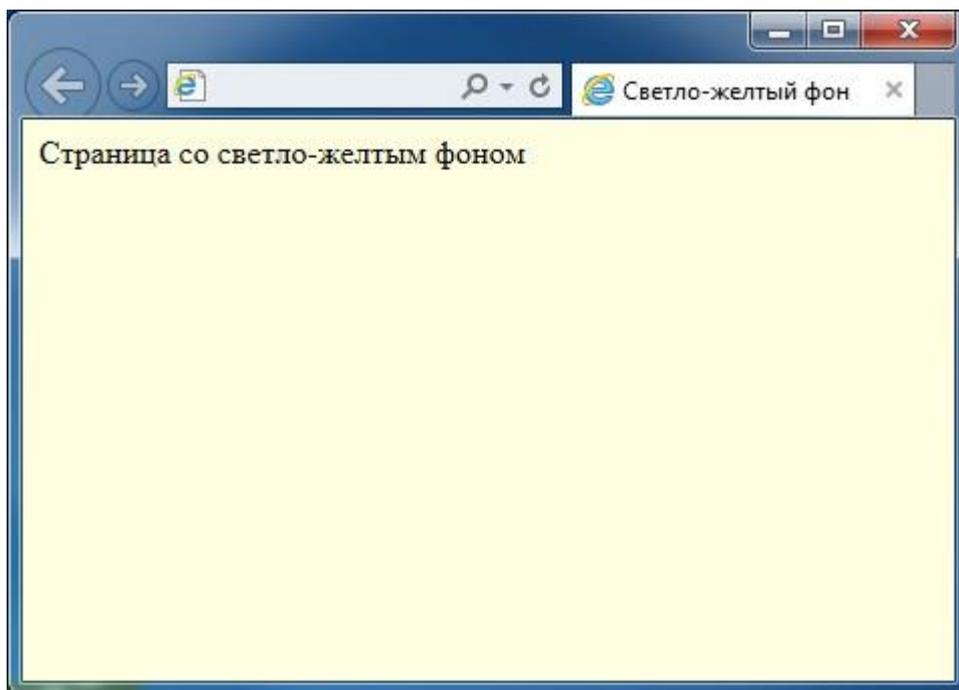
Заголовок 6

### Изменения фонового оформления страницы

У тега <BODY> есть несколько атрибутов, позволяющих поменять фон на странице, цвет текста, а также задать отступы от границ окна. Цвет фона страницы можно задать с помощью атрибута **bgcolor**.

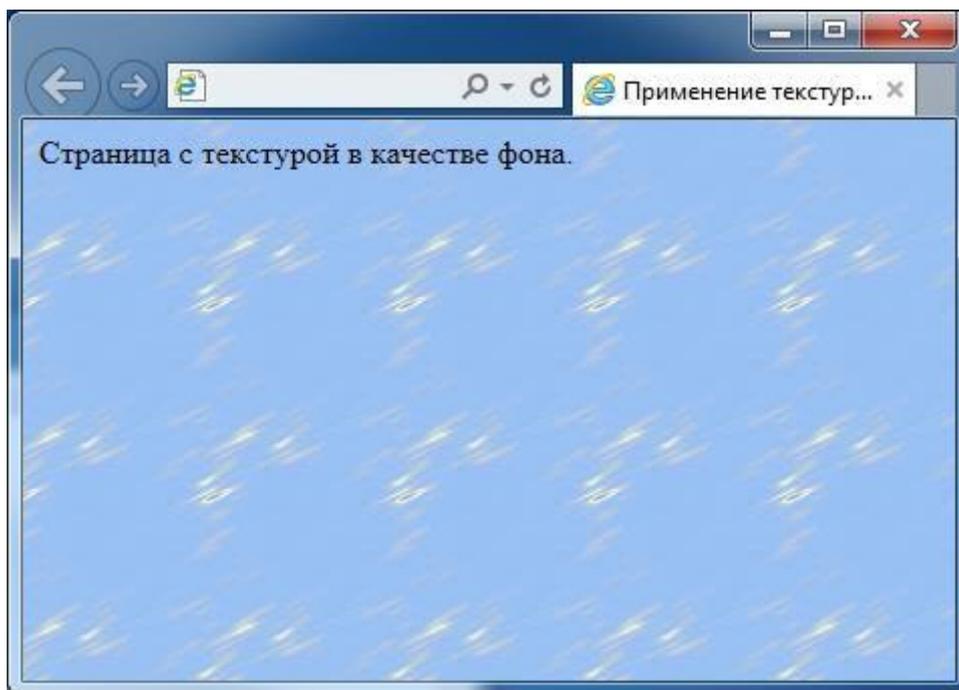
#### Пример 5. Страница со светло-желтым фоном

```
<html>  
<head>  
  <title>Светло-желтый фон</title>  
</head>  
<body bgcolor="lightyellow">  Страница со светло-желтым фоном  
</body>  
</html>
```



Атрибут **b bgcolor** устанавливает одноцветный фон на странице. Но в качестве фона может быть использована и повторяющаяся картинка, или так называемая текстура. Для этого служит следующий атрибут **background**, позволяющий установить текстурное изображение в качестве фона Webстраницы.

```
Пример 6. Страница с изображением в качестве текстуры <html>  
<head> <title>Применение текстурного  
фона</title>  
</head>  
<body background="texture.jpg">  
    Страница с текстурой в качестве фона.  
</body>  
</html>
```



В данном примере в качестве текстуры фона использовалось изображение в файле `texture.jpg`. Чтобы браузер смог найти изображениеисточник, оно должно быть расположено в одной папке с HTML-файлом, для которого устанавливается текстура. В противном случае необходимо прописывать полный адрес изображения источника.

### Задания для самостоятельной работы

1. Создайте HTML-документ и оформите в нем текст в соответствии с образцом вашего варианта.
2. С использованием таблицы цветов подберите удобный для восприятия цвет фона страницы.
3. Создайте копию полученного HTML-документа. В качестве фона этого документа используйте текстурное изображение, выбранное вами из папки с текстурами.

#### *Вариант 1*

**Пусть вечно спорят**

*дверь с ключом*

**нервенетве**

О и .

**важности**

Главнее нету в споре том

**Глазка замочной скважины.**

### Вариант 2

Есть в близости  
людей заветная черта,  
Ее не перейти влюбленности<sup>и</sup> страсти, —  
*Пусть в жуткой*  
ТИШИНЕ сливаются уста,  
И сердце рвется от любви на части.

### Вариант 3

Все, что в сердце  
твоим туманится,  
Станет ясно в моей тишине,  
*И когда он с тобой* расстанется,  
Ты признаешься только мне.

### Вариант 4

Любовь глубокой  
нежности полна,  
В соблазнах, *горестях* закалена,  
Крепка в разлуке, вдалеке — горда,  
Все та же — чудо, долгие года.

### Вариант 5

Жизнь не состязание в борьбе.  
С юмором живи, *но без усмешки.*  
живи *обдуманно*, без спешки.  
учись  
Силы пробуждать в себе И

### Вариант 6

Октябрь уж наступил,  
Уж роща отряхает последние листы С нагих своих ветвей.  
Дохнул осенний хлад,  
Дорога промерзает,

Журча, <sup>еще</sup> бежит

## **За мельницу ручей**

### **Вариант 7**

И в ночи январской,

Сам дивясь судьбе, небывалой

беззвездной,

Возвращенный из смертной бездны,

Ленинград салютует себе.

### **Вариант 8**

И ты хочешь

А тебе еще мало по-русски ,

на всех языках **З**нать, как круты подъемы и спуски

И ночем у нас

совесть и страх.

### **Вариант 9**

Что войны, что чума? - конец им виден <sup>скорый</sup> ,  
Им приговор почти произнесен.

**Но кто нас защитит** от ужаса, который

**Был бегом** времени **когда-то наречен?**

### **Вариант 10**

*Дорогою ценой*

и неожиданной

помнишь

Я узнала, что и .

ждешь

А быть может, **и место найдешь Ты - могилы моей** *безымянной* .

## Лабораторная работа №10. Создание списков

### Нумерованные списки

Нумерованные списки представляют собой набор элементов с их порядковыми номерами. Вид и тип нумерации зависит от атрибутов тега **OL**, который и используется для создания списка. В качестве маркеров могут быть следующие значения: арабские цифры заглавные латинские буквы прописные латинские буквы заглавные римские цифры прописные римские цифры

Ниже, в таблице приведены различные атрибуты тега **OL** и результат их применения.

Код HTML	Пример
<code>&lt;ol&gt; &lt;li&gt;текст&lt;/li&gt; &lt;li&gt;текст&lt;/li&gt; &lt;li&gt;текст&lt;/li&gt; &lt;/ol&gt;</code>	Нумерованный список с параметрами по умолчанию: 1. текст 2. текст 3. текст
<code>&lt;ol start="5"&gt;</code>	Нумерованный список, начинающийся с пяти: 5. текст 6. текст 7. текст
<code>&lt;ol type="A"&gt;</code>	Нумерованный список с заглавными буквами латинского алфавита: А. текст В. текст С. текст
<code>&lt;ol type="a"&gt;</code>	Нумерованный список с прописными буквами латинского алфавита: а. текст б. текст с. текст
<code>&lt;ol type="I"&gt;</code>	Нумерованный список с прописными римскими буквами: I. текст II. текст III. текст
<code>&lt;ol type="i"&gt;</code>	Нумерованный список со строчными римскими буквами: i. текст ii. текст iii. текст
<code>&lt;ol type="1"&gt;</code>	Нумерованный список с арабскими цифрами: 1. текст 2. текст 3. текст
<code>&lt;ol type="I" start="7"&gt;</code>	Список с римскими цифрами начинающийся с семи: IV. текст V. текст VI. текст

## Маркированные списки

Маркированные списки позволяют разбить большой текст на отдельные блоки. Тем самым привлекается внимание читателя к тексту и повышается его читабельность. С учетом худшего восприятия текста с экрана монитора, чем печатного варианта, это является весьма полезным приемом.

Для установки маркированного списка используется тег **UL** и **LI** (Пример 1).

### Пример 1. Создание маркированного списка

```
<html>
<body>
Что следует учитывать при тестировании сайта:
<ul>
<li>работоспособность всех ссылок</li>
<li>поддержку разных браузеров</li>
<li>читабельность текста</li>
</ul>
</body>
</html>
```

Ниже показан результат примера 1.

Что следует учитывать при тестировании сайта:

- работоспособность всех ссылок
- поддержку разных браузеров
- читабельность текста

Обратите внимание, что у маркированного текста появляются отступы сверху и снизу. Чтобы от них избавиться, список можно делать без тега **UL**. При этом исчезнут и отступы текста перед маркерами.

### Пример 2. Создание маркированного списка без отступов

```
<html>
<body>
Что следует учитывать при тестировании сайта:
<li>работоспособность всех ссылок</li>
<li>поддержку разных браузеров</li>
<li>читабельность текста</li>
</body>
</html>
```

Ниже показан результат примера 2.

Что следует учитывать при тестировании сайта:

- работоспособность всех ссылок
- поддержку разных браузеров
- читабельность текста

Маркеры могут принимать один из трех видов: круг (по умолчанию), окружность и квадрат. Для выбора типа маркера используется атрибут **type="..."** тега **UL**. Вместо многоточия подставляется одно из трех значений, указанных в таблице.

Код HTML	Пример
<code>&lt;ul type="disc"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"><li>• работоспособность всех ссылок</li><li>• поддержку разных браузеров</li><li>• читабельность текста</li></ul>
<code>&lt;ul type="circle"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"><li>○ работоспособность всех ссылок</li><li>○ поддержку разных браузеров</li><li>○ читабельность текста</li></ul>
<code>&lt;ul type="square"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"><li>■ работоспособность всех ссылок</li><li>■ поддержку разных браузеров</li><li>■ читабельность текста</li></ul>

С помощью CSS этот список можно расширить и вместо встроенных символов использовать в качестве маркера рисунки.

### Многоуровневые списки

Многоуровневые списки применяются на веб-страницах для создания сложной структуры текста либо для использования у многоуровневых меню. Создать многоуровневый список HTML совсем не сложно, достаточно в один из элементов списка `<LI>` `</LI>` вложить тег `<UL>` или `<OL>` с новыми элементами списка вот так выглядит код подобной конструкции:

### Пример 3. Создание многоуровневого списка

```
<html>
<body>
Основные устройства компьютера:
<ol>
<li>монитор</li> <li>системный
блок
<ul>
<li>материнская плата</li>
<li>процессор</li>
</ul>
</li>
<li>клавиатура</li> <li>мышь</li>
</ol>
</body>
</html>
```

Результат выглядит следующим образом:

- Основные устройства компьютера:
1. монитор
  2. системный блок
    - материнская плата
    - процессор
  3. клавиатура
  4. мышь

Как видно из примера вложенный список получается, путем вставки одного списка в другой. То есть один список вставляется в элемент «li» другого списка. Единственное неудобство, что такой вид организации списков легко может привести к путанице.

#### Графические маркеры списка

В качестве маркеров списка можно использовать графические изображения, что широко применяется для создания привлекательных, красиво оформленных HTML-документов. На самом деле такая возможность не предоставляется непосредственно языком HTML, а реализуется несколько искусственно.

Чтобы понять идею, необходимо разобраться в механизме реализации списков на HTML-страницах. Оказывается, что тег списка **<UL>** (как, впрочем, и теги списков других типов, рассматриваемых ниже) выполняет единственную задачу — указывает браузеру, что вся информация, располагаемая после данного тега должна отображаться со сдвигом вправо

(отступом) на некоторую величину. Теги **<LI>**, указывающие на отдельные элементы списка, обеспечивают вывод стандартных маркеров элементов списка.

Если же нам требуется построить список с графическими маркерами, то можно вообще обойтись без тегов **<LI>**. Достаточно будет перед каждым элементом списка вставить желаемое графическое изображение. Единственной задачей, которую нужно при этом решить, будет отделение элементов списка друг от друга. Для этого можно использовать теги абзаца **<P>** или принудительного перевода строки **<BR>**.

**Пример 4. Создание списка с графическими маркерами <html>**

```
<head>
<title>Маркированный список</title>
</head>
<body>
<ul>
<b>Знаки зодиака:</b><br>
 Овен<BR>
 Телец<BR>
 Близнецы<BR>
 Рак<BR>
 Лев<BR>
 Дева<BR>
 Весы<BR>
 Скорпион<BR>
 Стрелец<BR>
 Козерог<BR>
 Водолей<BR>
 Рыбы
</ul>
</body>
</html>
```

В окне браузера список выглядит следующим образом:

### Знаки зодиака:

- ★ Овен
- ★ Телец
- ★ Близнецы
- ★ Рак
- ★ Лев
- ★ Дева
- ★ Весы
- ★ Скорпион
- ★ Стрелец
- ★ Козерог
- ★ Водолей
- ★ Рыбы

В приведенном примере в качестве маркера элементов списка используется графический файл gold\_star.jpg. Заметим, что использование графики на HTML-страницах может значительно увеличить объем передаваемой информации. Однако в данном случае это увеличение крайне незначительно. Здесь для всех маркеров используется один и тот же файл, который будет передан только один раз. Размеры файла, содержащего маленькое изображение, также крайне незначительны.

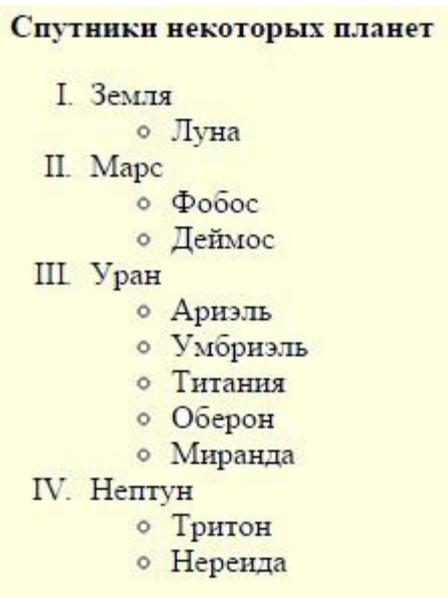
### Задания для самостоятельной работы

1. Создайте HTML-страницу с двухуровневым списком спутников некоторых планет и оформите согласно образцу используя типы маркера или нумерации, взятые из таблицы.

Таблица – Варианты оформления списков

№ варианта	1-ый уровень	2-ой уровень
1	Арабские цифры	Маркер "square"
2	Римские цифры строчные	Маркер "circle"
3	Римские цифры прописные	Арабские цифры
4	Прописные буквы	Римские цифры строчные
5	Маркер "disc"	Римские цифры прописные
6	Маркер "square"	Арабские цифры
7	Маркер "circle"	Римские цифры строчные
8	Арабские цифры	Римские цифры прописные
9	Римские цифры строчные	Прописные буквы
10	Римские цифры прописные	Маркер "disc"

Пример двухуровневого списка, в котором для первого уровня используются римские прописные цифры, а для второго - маркер "circle":



2. Оформите страницу используя цветной фон или фоновую текстуру.

3. Создайте рисунок небольшого размера в виде какого-либо значка или символа в графическом редакторе MS Paint (или любом другом). Измените созданный в задании 1 двухуровневый список используйте в качестве маркера для первого уровня собственный рисунок.

## Лабораторная работа № 11. Создание гиперссылок

Важнейшим свойством языка HTML является возможность размещения на странице ссылок на другие документы. Возможны ссылки:

- на удаленный HTML файл;
- на некоторую точку в текущем HTML-документе;
- на любой файл, не являющийся HTML-документом.

В качестве ссылки можно использовать текст или графику.

Для создания ссылки необходимо сообщить браузеру, что является ссылкой, а также указать адрес документа, на который следует сделать ссылку. Оба действия выполняются с помощью тега **A**, который имеет единственный атрибут **href**. В качестве значения используется адрес документа: для документа в сети интернет - это его **URL**, для документа на компьютере пользователя - это имя и путь к нему.

Адрес ссылки может быть абсолютным и относительным. Абсолютные адреса работают везде и всюду независимо от имени сайта или расположения html-страницы, где прописана ссылка.

### Пример 1. Использование абсолютных ссылок

```
<html>
<body>
<a href=http:\\www.yandex.ru>Абсолютная ссылка на сайт</a> <a
href="c:\\docs\\user\\main.html">Абсолютная ссылка на другой локальный
html-документ</a>
</body>
</html>
```

Относительные ссылки, как следует из их названия, построены относительно текущего документа или адреса. Примеры таких адресов:

1. /
2. /demo/
3. /images/pic.gif
4. ../help/me.html

5. manual/info.html Первые две ссылки называются неполные и указывают веб-серверу загружать файл index.html (или default.html) находящемуся в корне сайта (пример 1) или папке demo (пример 2). Если файл index.html отсутствует, браузер, как правило, показывает список файлов, находящихся в данном каталоге. Слэш перед адресом говорит о том, что адресация начинается от корня сайта (пример 3), двоеточие - перейти на уровень выше в списке каталогов сайта (пример 4).

## Пример 2. Использование относительных ссылок

```
<html>
<body>
<a href=images/photo.jpg>Относительная ссылка на рисунок</a><br> <a
href=main.html> Относительная ссылка на другой html-документ или
страницу</a>
</body>
</html>
```

## Ссылки внутри страницы

Большие документы читаются лучше, если они имеют оглавление со ссылками на соответствующие разделы. Для создания ссылки следует вначале сделать закладку в соответствующем месте и дать ей имя при помощи атрибута name тега **A**.

## Пример 3. Создание внутренней ссылки

```
<html>
<body>
<a name=top></a>Друг уронил утюг в унитаз. И разбил его. Причем так
разбил, что по назначению унитаз и использовать никак нельзя.
Мгновением назад только что вот все было хорошо и вот уже дыра, да
такая, что можно забыть, что есть такой предмет в доме. Махнул рукой
нечаянно, а потом мучайся...
<a href=#top>Наверх</a>
</body>
</html>
```

Между тегам **<a name=top>** и **</a>** отсутствует текст, так как требуется лишь указать местоположение перехода по ссылке, находящейся внизу страницы. Имя ссылки на закладку начинается символом #, после чего идет название закладки. Название выбирается любое, соответствующее тематике.

Можно, также, делать ссылку на закладку, находящуюся в другой вебстранице и даже другом сайте. Для этого в адресе ссылки надлежит указать ее адрес и в конце добавить символ решетки # и имя закладки.

## Пример 4. Ссылка на закладку из другой веб-страницы

```
<html>
<body>
<a href=text.html#bottom>Перейти к нижней части текста</a>
</body>
</html>
```

### Ссылка на новое окно

Если требуется сделать ссылку на документ, который открывается в новом окне браузера, используется атрибут **target=\_blank** тега **A**.

Создание нового окна обычно требуется в случаях, когда делается ссылка на другой сайт, в остальном лучше открывать документы в текущем окне, поскольку обилие окон может сбить читателя с толку.

Так как ссылки на текущее или новое окно ничем не отличаются друг от друга, на некоторых сайтах рядом со ссылкой ставят специальную иконку, показывающую, что документ открывается в новом окне.

#### Пример 5. Создание ссылки на новое окно

```
<html>
<body>
<a href=http:\\www.igsha.ru>Обычная ссылка на сайт
www.igsha.ru</a><br><a href=http:\\www.igsha.ru target=_blank>Ссылка
открывает новое окно на сайт www.igsha.ru</a>
</body>
</html>
```

[Обычная ссылка на сайт www.igsha.ru](http://www.igsha.ru)

[Ссылка открывает новое окно на сайт www.igsha.ru](http://www.igsha.ru)

С помощью HTML можно задавать цвета ссылок на странице.

Для этого используются следующие атрибуты тега **<BODY>**.

1. **link** — определяет цвет ссылок на веб-странице. Цвет по умолчанию синий, #0000FF.
2. **alink** — цвет активной ссылки. Цвет ссылки меняется при нажатии на ней кнопки мыши. Цвет по умолчанию красный, #FF0000.
3. **vlink** — цвет уже посещенных ссылок. Цвет по умолчанию фиолетовый, #800080.

#### Пример 6. Изменение цвета ссылок

```
<html>
<body link="red" vlink="grey" alink="blue" bgcolor="lightyellow">
<p><a href="content.html">Содержание сайта</a></p>
</body>
</html>
```

В данном примере цвет ссылок установлен красным, а цвет уже посещенных ссылок - серым. Ссылка перед нажатием и после него:

## Задания для самостоятельной работы

1. Создайте в html-документе словарь терминов, который содержит внутренние ссылки для перехода по заглавным буквам терминов и на начало документа согласно образцу. Тематику словаря выбрать из таблицы 1.

## Информатика - словарь терминов

[А](#) [Б](#) [В](#) [Г](#) [Д](#) [Е](#) [Ж](#) [З](#) [И](#) [К](#)

### А

#### АРХИТЕКТУРА ФОН НЕЙМАНА

архитектура компьютера, имеющего одно арифметико-логическое устройство, через которое проходит поток данных, и одно устройство управления, через которое проходит поток команд.

#### АСИНХРОННАЯ ПЕРЕДАЧА ДАННЫХ

способ передачи и метод извлечения данных из непрерывного потока сообщений, при которых передающая сторона в каждое данное вводит стартовый и стоповый биты, указывающие, где данное начинается и где кончается.

[В начало](#)

### Б

#### БАЙТ

машинное слово минимальной размерности, адресуемое в процессе обработки данных. Размерность байта - 8 бит - принята не только для представления данных в большинстве компьютеров, но и в качестве стандарта для хранения данных на внешних носителях, для передачи данных по каналам связи, для представления текстовой информации.

#### БОД

единица измерения скорости передачи данных.

[В начало](#)

### В

Таблица 1 – Варианты тематик словаря терминов

№ варианта	Тематика словаря терминов
1	Информатика
2	Информационные технологии
3	Физика
4	География
5	Биология
6	Экономика
7	Управление

8	Право
9	Социология
10	Общество

**Образец** (фрагмент словаря терминов)

В качестве источника информации для словаря терминов можно использовать сайт <http://www.glossary.ru> или другие. Словарь должен содержать 10 групп терминов с одинаковыми заглавными буквами (начиная от буквы А) по 2-3 термина в каждом (если имеются).

2. Оформите html-документ используя цветной фон и разные типы шрифтов для названий терминов и их определений. Добавьте горизонтальные разделительные линии между группами терминов по заглавным буквам используя одиночный тег **<HR>**. Сохраните полученный документ под именем glossary.html.

3. Создайте словарь терминов состоящий из нескольких html-документов: главной страницы и страниц групп терминов, переход между которыми осуществляется с использованием внешних ссылок. Для этого выполните следующее.

3.1. Для каждой группы терминов создайте собственные html-документы, например, group\_a.html, group\_b.html и т.д.

**Образец** (содержание group\_a.html - группа терминов на букву А)

## Информатика - словарь терминов

---

### А

**АРХИТЕКТУРА ФОН НЕЙМАНА**  
архитектура компьютера, имеющего одно арифметико-логическое устройство, через которое проходит поток данных, и одно устройство управления, через которое проходит поток команд.

**АСИНХРОННАЯ ПЕРЕДАЧА ДАННЫХ**  
способ передачи и метод извлечения данных из непрерывного потока сообщений, при которых передающая сторона в каждое данное вводит стартовый и стоповый биты, указывающие, где данное начинается и где кончается.

[На главную](#)

3.2. Создайте документ main.html содержащий только название словаря терминов и внешние относительные ссылки на html-документы групп терминов (group\_1.html, group\_2.html и т.д.).

**Образец главной страницы словаря**

# Информатика - словарь терминов

---

[А](#)[Б](#)[В](#)[Г](#)[Д](#)[Е](#)[Ж](#)[З](#)[И](#)[К](#)

Для переходов между страницами используйте относительные ссылки.

## Лабораторная работа № 12. Вставка изображений в html-страницы

Для вставки **изображений в HTML** применяются два основных формата GIF и JPEG. Формат GIF может хранить внутри себя простейшую анимацию (динамические баннеры), JPEG отлично подходит для изображений с большим количеством цветов, например, фотографий. Третьим форматом для вебграфики является формат PNG, но он не получил широкого применения в вебдизайне. Любое изображение в форматах GIF или JPEG вставляется на вебстраницу при помощи тега **<IMG>**, закрывающего тега нет.

Посредством атрибута **src** задается адрес (URL) файла с изображением, т.е. браузер находит нужное изображение в каталоге сайта по пути (адресу URL), прописанному в этом атрибуте. Для указания адреса изображения можно задавать как абсолютный, так и относительный адрес. Для удобства все изображения сайта находятся в отдельной папке, обычно с именем **image**.

Для примера возьмите любое изображение, лучше небольшого формата, и сохраните в созданной папке image. Пусть это будет эмблема нашего вуза с именем **irgau.jpg**. Далее мы будем обращаться к ней для обучения.

### Пример 1. Вставка изображения

```
<html>
<body>

</body>
</html>
```

Браузер отобразит:



Теперь попробуем вставить изображение с текстом:

### Пример 2. Вставка изображения с текстом

```
<html>
<body>
<p>Текст в который просто вставлена картинка.Обтекание не задано.</p>
</body>
</html>
```



Текст в который просто вставлена картинка.

Обтекание не задано.

По умолчанию, если не задано никаких параметров обтекания, рисунок располагается на странице как объект строки (слово, буква и т.д.). Высота строки при этом увеличивается до размеров рисунка, а текст не распределяется вокруг изображения. Обтекание задается с выравниванием изображения относительно правого или левого края страницы. После чего рисунок вставляется на нужное место, а текст располагается с другой стороны.

### Выравнивание изображений

Для изображений можно указывать их положение относительно текста или других изображений на веб-странице. Способ выравнивания изображений задается атрибутом **align** тега **IMG**. В таблице перечислены возможные значения этого атрибута и результат его использования.

Код HTML	Описание	Пример
<code>&lt;img src="smile.jpg" align=texttop&gt;</code>	Верхняя граница изображения выравнивается по самому высокому текстовому элементу текущей строки.	Lorem ipsum dolor sit amet,  consectetur adipiscing elit...
<code>&lt;img src="smile.jpg" align=top&gt;</code>	Верхняя граница изображения выравнивается по самому высокому элементу текущей строки.	ipsum dolor sit amet,  consectetur adipiscing elit...
<code>&lt;img src="smile.jpg" align=middle&gt;</code>	Выравнивание середины изображения по базовой линии текущей строки.	amet,  consectetur adipiscing elit...
<code>&lt;img src="smile.jpg" align=absmiddle&gt;</code>	Выравнивание середины изображения посередине текущей строки.	amet,  consectetur adipiscing elit...

baseline</b> >	Выравнивание изображения по базовой линии текущей строки.	Lorem ipsum dolor sit amet, consectetur adipiscing elit... 
bottom</b> >	Выравнивание нижней границы изображения по окружающему тексту.	Lorem ipsum dolor sit amet, consectetur adipiscing elit.  ..
left</b> >	Выравнивает изображение по левому краю окна.	 Lorem ipsum dolor sit amet, consectetur adipiscing elit...
right</b> >	Выравнивает изображение по правому краю окна.	Lorem ipsum dolor sit amet,  consectetur adipiscing elit...

Наиболее популярные атрибуты – **left** и **right**, создающие обтекание текста вокруг изображения.

### Пример 3. Вставка изображения с обтеканием

```
<html>
<body>
<p>Текст, в который просто
вставлена картинка. Задано расположение изображения у левого края
страницы, текст обтекает справа.</p>
</body>
</html>
```

Браузер отобразит код так:



Текст, в который просто вставлена картинка.  
Задано расположение изображения у левого края  
страницы, текст обтекает справа.

Аналогичным образом можно задать обтекание рисунка другими способами, представленными в таблице.

### Создание отступов при обтекании текста

Кроме способа выравнивания и обтекания текстом, для изображения можно задать поля отступов, которые не будут заняты текстом при

обтекании. Задаются отступы двумя атрибутами: *vspace* - верхний и нижний, *hspace* - левый и правый. Значения задаются в пикселях (px). Рассмотрим пример с выравниванием изображения по левому краю и отступами слева и справа - 35px, сверху и снизу - 25px.

#### Пример 4. Задание отступов

```
<html>
<body>
<p>Текст в который просто вставлена картинка.  Задано расположение изображения у левого края страницы, текст
обтекает справа.</p>
</body>
</html>
```

Результат выглядит следующим образом:

Текст в который просто вставлена картинка.



Задано расположение  
изображения у левого  
края страницы, текст  
обтекает справа.

#### Создание рамки вокруг текста

В HTML можно задать рамку, определенной толщины, вокруг изображения. Делается это с помощью атрибута *border*, значение которого задается в пикселях. Единственный минус в том, что цвет рамки будет черным, а изменить его средствами HTML никак нельзя. Но можно заранее на изображении, в графическом редакторе, выполнить рамку произвольного цвета. Рассмотрим пример написания кода вставки изображения с рамкой.

#### Пример 5. Создание рамки вокруг текста

```
<html>
<body>
<p>Текст, в который вставлена картинка с
рамкой.Задано расположение по центру.</p>
</body>
</html>
```

В результате в браузере отобразится следующее:

Текст, в который вставлена  
картинка с рамкой.



Задано

расположение по центру.

### Изображения - ссылки

Изображения могут быть не только приемом web-дизайна, но и [гиперссылками](#) на другие web-страницы. Делается это точно так же, как с текстом, при помощи того самого тега `<A>`. В тег `<A>` просто включается тег нужного изображения. Лучше всего к такой ссылке (как и к любой гиперссылке) добавить тег *title*, так как это дает дополнительную информацию.

#### Пример 6. Вставка изображения-ссылки

```
<html>  
<body>  
<a href="http://www.igsha.ru/" title="Логотип сайта igsha.ru"></a >  
</body>  
</html>
```

В браузере отобразится логотип ИрГАУ, по нажатию которого открывается сайт вуза:



При наведении курсора к изображению появляется всплывающая подсказка «Логотип сайта igsha.ru». По умолчанию изображения содержащие гиперссылку отображаются в рамке фиолетового цвета. Для того чтобы она не отображалась значение атрибута `border` необходимо установить равным нулю (`border="0"`).

## Альтернативный текст

В каждом браузере есть функция отключения изображений. Пользователь, использующий такую функцию, может видеть описание того что представляет собой изображение изображения. Это описание (или альтернативный текст) указывается в теге *alt*.

### Пример 7. Создание альтернативного текста

```
<html>
<body>

</body>
</html>
```

Увидеть альтернативный текст можно выключив отображение изображений в браузере.

## Изменение размера изображения

Размеры изображения задаются атрибутами *width* - ширина и *height* - высота, значения задаются как в пикселях, так и в процентах от ширины экрана.

### Пример 8. Изменение размера изображения

```
<html>
<body>

</body>
</html>
```

Некоторые рекомендации по вставке изображений на web-страницу.

1. Старайтесь не использовать слишком большие размеры файла изображения, так как это влияет на скорость загрузки страницы.
2. Атрибут **ALT** имеет очень важное значение, рекомендуется добавлять его для каждого тега **IMG**. Содержание текстового сообщения должно очень точно описывать изображение, причем кратко.

3. Изображения на web-странице должны соответствовать текстовому содержанию.

#### **Задание**

1. Предположим, у Вас есть файл изображения mypicture.gif. Включите в HTML-страницу это изображение:

```

```

Заметим, что в этом случае файл mypicture.gif должен находиться в том же самом каталоге (папке), где находится ваша страница.

2. Загрузите страницу в браузер и посмотрите, что получилось.

3. Включите в HTML-страницу изображение, которая находится в специальном каталоге (папке) для файлов изображений. Обычно, такой каталог называется images и он является подкаталогом основного каталога, где находятся файлы страниц.

Создайте такой каталог и включите в Вашу страницу следующий тэг:

```

```

4. Загрузите страницу в браузер и посмотрите, что получилось.

5. Создайте каталог images2, который находится на том же уровне, что и каталог с Вашей страницей. Включите в Вашу страницу следующий тэг:

```

```

6. Загрузите страницу в браузер и посмотрите, что получилось.

7. Измените размеры изображения, включенного в Вашу HTML-страницу:

```

```

где x – ширина изображения в пикселях, а y – высота изображения (также в пикселях).

8. Сделайте рамку вокруг изображения:

```

```

Заметьте, что значение атрибута **border** есть толщина рамки. 9. Загрузите страницу в браузер и посмотрите, что получилось.

### Задания для самостоятельной работы

1. Скопируйте любое изображение, расположенное в папке images в свою папку, внедрите его в текст из лабораторной работы №1 используя относительные ссылки и выровняйте его относительно текста согласно варианту.

№ варианта	Способ выравнивания
1	верхняя граница изображения выравнивается по самому высокому элементу текущей строки
2	выравнивание середины изображения по базовой линии текущей строки
3	выравнивание середины изображения посередине текущей строки
4	выравнивание изображения по базовой линии текущей строки
5	выравнивание нижней границы изображения по окружающему тексту
6	выравнивание изображения по левому краю окна
7	выравнивание изображения по правому краю окна
8	выравнивание верхней границы изображения по самому высокому текстовому элементу текущей строки
9	выравнивание нижней границы изображения по окружающему тексту
10	выравнивание середины изображения посередине текущей строки

2. Добавьте к рисунку альтернативный текст, с произвольным содержанием.

3. Добавьте в изображение ссылку на страницу с информацией о вашем факультете (используйте официальный сайт вуза). 4. Измените цвет и толщину рамки вокруг изображения.

## Лабораторная работа № 13. Создание таблиц

Таблица состоит из строк и столбцов ячеек, которые могут содержать текст и рисунки. Обычно таблицы используются для упорядочения и представления данных, однако возможности таблиц этим не ограничиваются. С помощью таблиц удобно верстать макеты страниц, расположив нужным образом фрагменты текста и изображений.

Для добавления таблицы на веб-страницу используется тег-контейнер **TABLE**. Таблица должна содержать хотя бы одну строку и колонку.

Для добавления строк используются теги **<tr>** и **</tr>**. Чтобы разделить строки на колонки применяются теги **<td>** и **</td>**. Теги **<th>** и **</th>** иногда заменяют **<td>** и **</td>** когда необходимо выровнять текст по центру и сделать его полужирным.

### Параметры таблицы

Для изменения вида и свойств таблицы используется множество атрибутов, которые добавляются в теге **TABLE**.

**<table атрибут1=... атрибут2=...>** Описание атрибутов таблицы и их свойств описано ниже.

Свойство	Значение	Описание	Пример
<b>align=</b>	Left Right Center	Выравнивание таблицы	align=center
<b>background=</b>	URL	Фоновый рисунок	background=pic.gif
<b>bgcolor=</b>	#rrggbb	Цвет фона таблицы	bgcolor=#FF9900
<b>border=</b>	n	Толщина рамки в пикселах	border=2
<b>bordercolor=</b>	#rrggbb	Цвет рамки	bordercolor=#333333
<b>bordercolordark=</b>	#rrggbb	Тень рамки	border-color-dark=#f0f0f0
<b>cellpadding=</b>	n	Расстояние между ячейкой и ее содержимым	cellpadding=7
<b>cellspacing=</b>	n	Дистанция между ячейками	cellspacing=3
<b>nowrap</b>		Запрещает переносы строк в тексте	<table nowrap>

<b>frame=</b>	Void Above Below Lhs Rhs Hsides Vsides Box	Задание типа рамки таблицы	frame=hsides
<b>valign=</b>	Top Bottom	Выравнивание по высоте	valign=top
<b>width=</b>	n n%	Минимальная ширина таблицы, можно задавать в пикселах или процентах	width=90%
<b>height</b>	n n%	Минимальная высота таблицы, можно задавать в пикселах или процентах	height=18

### Примечание

1. Таблица, если не указано особо, всегда выравнивается по левому краю.
2. Если ширина таблицы не указана, она подгоняется под содержание ячеек.

**Задание Создать 3 таблицы, используя приведенный ниже листинг**

```

1.1 Создание таблицы
<html>
<head>
<title>таблица</title>
</head>
<body>
<table border="2" align=center>
<tr>
<td colspan=2 align=center>
<b>Заголовок таблицы</b>
</td>
</tr>
<tr>
<td align="center">
Первая ячейка первой строки
</td>
<td align="center">
Вторая ячейка первой строки

```

```

</td>
</tr>
<tr>
<td align="center">
Первая ячейка второй строки
</td>
<td align="center">
Вторая ячейка второй строки
</td>
</tr>
</table>
</body>
</html>

```

Тег **<TABLE>** задает таблицу. Атрибуты **border="2"** и **align=center** задают, соответственно, размер границ таблицы и выравнивание ее по центру страницы. Тег **<TR>** задает строку таблицы. Тег **<TD>** задает ячейку таблицы.

Атрибут **colspan=n** объединяет n ячеек по горизонтали (по столбцам). В результате получится следующая таблица, состоящая из двух столбцов и двух строк:

Заголовок Таблицы.	
Первая ячейка первой строки	Вторая ячейка первой строки
Первая ячейка второй строки	Вторая ячейка второй строки

```

1.2. Сложная таблица <html>
<head>
<title>сложная таблица</title>
</head>
<body>
<table border="1" width="75%" align=center>
<tr>
<td width="66%" colspan="2">
<p align="center">Две ячейки, объединенные по горизонтали </td>
</tr>
<tr>
<td width="33%" rowspan="2" valign="middle">Две ячейки,
объединенные по вертикали
</td>
<td width="33%"> по
левому краю </td>
</tr>
<tr>
<td width="33%">
<p align="right">по правому краю
</td>
</tr>
</table>
</body>
</html>

```

Атрибут **colspan=n** объединяет ячейки по горизонтали (по столбцам)

Атрибут **rowspan=n** объединяет ячейки по вертикали (по строкам)

Атрибут **valign="middle"** выравнивает текст в ячейке по центру ячейки по вертикали

В результате получается следующая таблица:

Две ячейки, объединенные по горизонтали	
Две ячейки, объединенные по вертикали	по левому краю
	по правому краю

### 1.3. Более сложная таблица <html>

```
<head>
<title>УЧЕБНАЯ ТАБЛИЦА</title>
</head>
<body>
<table width="50%" border="6" cellspacing="6" cellpadding="20"
align="center" bordercolorlight="lime" bordercolordark="green"
bgcolor="#dffffd"><thead bgcolor="aqua">
<tr><th colspan="3">УЧЕБНАЯ ТАБЛИЦА</th></tr>
</thead>
<tbody>
<tr>
<td width="33%">Это первая ячейка</td>
<td width="33%">Это вторая ячейка</td>
<td rowspan="3">А это три ячейки третьего столбца объединились в
одну большую</td>
</tr>
<tr>
<td colspan="2">Это единственная ячейка второй строки,
объединяющая оба столбца</td>
</tr>
<tr>
<td> Это первая ячейка третьей строки</td>
<td width="33%">А это вторая ячейка третьей строки</td> </tr>
</tbody>
<tfoot bgcolor="yellow">
<tr>
<td colspan="3" align="center"> <small>конец</small></td></tr>
</tfoot>
</table>
</body>
</html>
```

В результате получается следующая таблица:

УЧЕБНАЯ ТАБЛИЦА		
Это первая ячейка	Это вторая ячейка	А это три ячейки третьего столбца объединились в одну большую
Это единственная ячейка второй строки, объединяющая оба столбца		
Это первая ячейка третьей строки	А это вторая ячейка третьей строки	
конец		

Атрибут **CELLSPACING="6"** задает свободное пространство между ячейками таблицы

Атрибут **CELLPADDING="20"** задает свободное пространство между данными в ячейке и ее границами

Атрибут **BORDER COLORLIGHT="Lime"** задает цвет левого и верхнего углов таблицы

Атрибут **BORDER COLORDARK="Green"** задает цвет правого и нижнего углов таблицы

Атрибут **BGCOLOR="#DFFFFD"** задает цвет фона таблицы В

### Задания для самостоятельной работы

Используя атрибуты объединения ячеек, изменения цвета фона и выравнивания текста создайте таблицу в HTML согласно образцу по варианту.

### Вариант 1

Таблица. Вариант 1		
Это первая ячейка	Это вторая ячейка	Это две объединенные ячейки
Это первая ячейка второй строки	Это две объединенные ячейки второго столбца	
Это первая ячейка третьей строки		Это последняя ячейка

### Вариант 2

Таблица. Вариант 2		
Это первая ячейка	Это вторая ячейка	Это третья ячейка
Это три объединенные ячейки		
Это первая ячейка третьей строки	Это две объединенные ячейки	

### Вариант 3

Таблица. Вариант 3		
Это первая ячейка	Это две объединенные ячейки	Это третья ячейка
Это две объединенные ячейки первого столбца		Это вторая ячейка третьей строки

### Вариант 4

Таблица. Вариант 4		
Это три объединенные ячейки первого столбца	Это вторая ячейка	Это третья ячейка
	Это две объединенные ячейки	
	Это вторая ячейка третьей строки	Это последняя ячейка

### Вариант 5

Таблица. Вариант 5		
Это две объединенные ячейки первого столбца	Это вторая ячейка	Это третья ячейка
	Это первая ячейка третьей строки	Это две объединенные ячейки второго столбца

### Вариант 6

Таблица. Вариант 6		
Это две объединенные ячейки	Это вторая ячейка	Это третья ячейка
	Это две объединенные ячеки второго столбца	Это третья ячейка второй строки
Это первая ячейка третьей строки		Это последняя ячейка

### Вариант 7

Таблица. Вариант 7		
Это две объединенные ячейки		Это третья ячейка
Это три объединенные ячейки		
Это первая ячейка третьей строки	Это вторая ячейка третьей строки	Это последняя ячейка

### Вариант 8

Таблица. Вариант 8		
Это две объединенные ячейки	Это вторая ячейка	Это две объединенные ячейки
	Это первая ячейка третьей строки	
		Это последняя ячейка

### Вариант 9

Таблица. Вариант 9		
Это первая ячейка	Это вторая ячейка	Это три объединенные ячейки
Это две объединенные ячейки		
Это первая ячейка третьей строки	Это вторая ячейка третьей строки	

### Вариант 10

Таблица. Вариант 10		
Это две объединенные ячейки	Это вторая ячейка	Это третья ячейка
	Это четыре объединенные ячейки	
Это первая ячейка третьей строки		

## Лабораторная работа № 14. Создание фреймов

Фреймы – это области окна браузера, в которые одновременно могут быть загружены разные страницы. Например, можно создать один фрейм для меню сайта, а другой фрейм будет использоваться для отображения информации, соответствующей каждому из пунктов меню.

Два вида тэгов используются для создания фреймов: **<frameset>** и **<frame>**. Первый из них определяет характеристики набора используемых фреймов, а второй – характеристики каждого фрейма. Синтаксис тэга **<frameset>** следующий:

```
<frameset rows|cols="a,b,c,..." border=n frameborder="1|0"  
framespacing=n>  
...  
</frameset>
```

Атрибуты **rows** и **cols** определяют, как должны располагаться фреймы в этом наборе фреймов: строками или столбцами. Если необходимо и то и другое, следует использовать вложенные наборы фреймов. Каждое из значений "a, b, c, ..." может быть следующим:

- **n** - высота/ширина строки/столбца, заданная в пикселях.
- **n%** - высота/ширина строки/столбца, заданная в процентах от высоты/ширины родительского окна или фрейма.
- **\*** - высота/ширина строки/столбца определяется доступным пространством.
- **n\*** - строка/столбец займет в n раз больше пространства, чем строка/столбец, определенные с \*.

Атрибут **border** определяет ширину рамки фреймов.

Атрибут **frameborder** определяет, должны ли фреймы иметь рамку. Если указано **frameborder="0"**, то рамки не будет.

Атрибуты **framespacing** и **border** (используются для разных браузеров) определяет ширину рамки фреймов. Полезно их использовать одновременно. Синтаксис тэга **<frame>** следующий:

```
<frame src="framesource" name="framename" scrolling="yes|no|auto"  
frameborder="1|0"  
noreferrer marginwidth="n" marginheight="n">  
...  
</frame>
```

Атрибут **src** определяет HTML-страницу, загружаемую во фрейм. Например, `<frame src="main.html">`.

Атрибут **name** задает имя фрейма.

Атрибут **scrolling** определяет, должен ли фрейм иметь полосы прокрутки содержания.

Атрибут **frameborder** определяет, должен ли фрейм иметь рамку. Если указано **frameborder="0"**, то рамки не будет.

Атрибут **noresize** определяет, что пользователь не сможет динамически менять размер фрейма.

Атрибуты **marginwidth** и **marginheight** определяют расстояние (в пикселях) между содержимым фрейма и рамкой.

Часто необходимо загрузить во фрейм страницу, а гиперссылка для выполнения этой операции расположена в другом фрейме. В этом случае Вы должны использовать в тэге `<a>` атрибут **target** и указать в качестве его значения имя фрейма. Например, для того, чтобы загрузить во фрейм **main** из фрейма **menu** страницу **services.htm**, разместите во фрейме **menu** следующую ссылку:

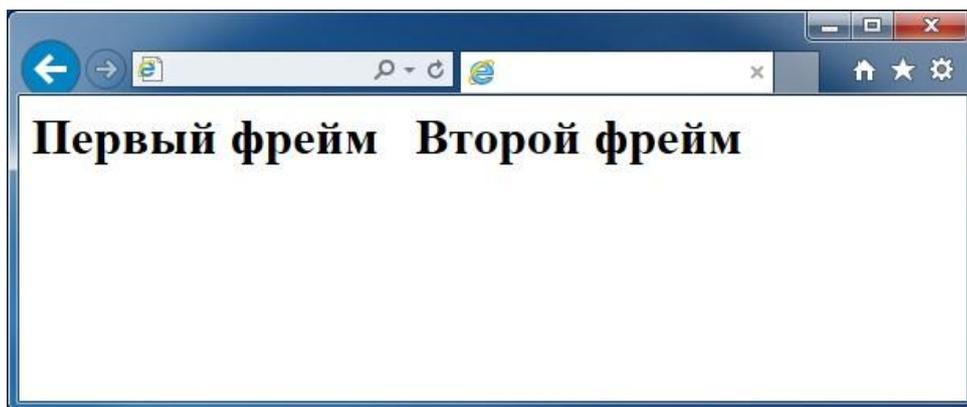
```
<a href="services.html" target="main">Наши услуги</a>
```

Ниже перечислены специальные значения, которые можно использовать для атрибута **target**:

- **\_parent** - страница будет загружена в текущий фрейм набора фреймов предыдущего уровня вложенности;
- **\_top** - страница будет загружена непосредственно в окно браузера;
- **\_blank** - страница будет загружена в новое окно браузера;
- **\_self** - страница будет загружена в тот же самый фрейм, где расположена ссылка на нее.

**Пример 1. Создание простого фрейма** `<html>`

```
<frameset cols="40%, 60%" border=0>  
<frame src="menu.html" name="menu">  
<frame src="main.html" name="main">  
</frameset>  
</html>
```



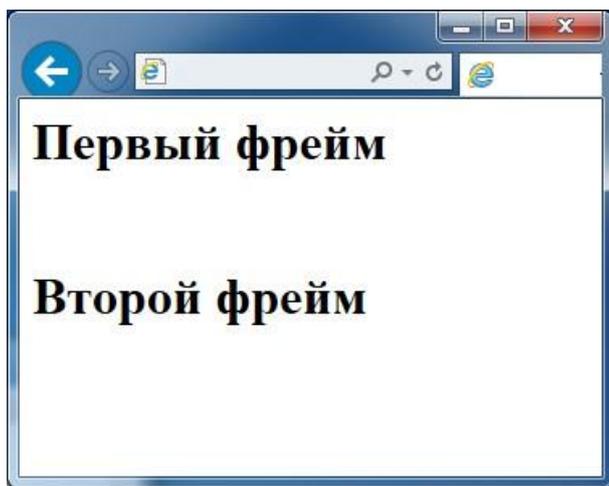
В приведенном примере присутствует объявление двух фреймов, которые будут располагаться вертикальными полосами и занимать соответственно 30 и 70 процентов рабочей области. Вертикальное расположение устанавливается атрибутом **rows="..."**, а для горизонтальных полос используют атрибут **cols="..."**. Параметр **border="..."** определяет границу между фреймами. Определение документа, изначально загружаемого при открытии этого файла (этот документ является HTML-страницей.), задается атрибутом тега **<FRAME> src="..."**, при этом необходимо указать атрибут **name="..."**, позволяющий задать "имя" созданной области в виде последовательности латинских букв и цифр, использованной как значение этого атрибута. Это имя можно использовать, чтобы загружать новые документы в ранее созданную область. Для этого в тег **<A>**, определяющий гиперссылку, необходимо добавить атрибут **target="..."**, значение которого совпадает с ранее определенным именем области. При переходе по данной гиперссылке новый документ загрузится в указанный фрейм. Например, предположим, что начальная страница Web-узла состоит из двух фреймов: слева располагается навигационная панель, а справа - текущая страница. Если правой области присвоено имя, используемое во всех ссылках, имеющих в левой области, то щелчок на любой ссылке навигационной панели приведет к обновлению информации в соседней области, оставляя навигационную панель без изменений. В данном документе находятся только ссылки на другие (существующие) HTML-документы, которые будут загружены браузером при открытии страницы с фреймами. Файл **menu.html** имеет следующую структуру:

```
<html>
<head>
<title></title>
</head>
<body>
<h1>Первый фрейм</h1>
</body>
</html>
```

Файл main.html также имеет указанную выше структуру. Следует сказать, что оба этих файла должны располагаться в том же каталоге, где расположен файл с фреймами.

### Задание 1.

Расположить эти фреймы по вертикали, используя атрибут **rows="..."**. В результате у вас должно получиться следующее:



Не все браузеры могут поддерживать фреймы. Для таких браузеров предусмотрено использование тэгов **<noframes>** и **</noframes>**, которые должны быть размещены перед тэгом **</frameset>**. Например, так:

```
...  
<noframes>  
<center><b>spoil yourself -get a new browser!</b></center>  
</noframes>  
</frameset>
```

Текст, размещенный внутри указанных тэгов, будет игнорироваться браузерами, поддерживающими фреймы.

### Задание 2.

1. Создайте набор фреймов, состоящий из меню в левой части экрана и фрейма для показа содержательных страниц – в правой. Например, так:

```
<html>  
<head> <title>Пример набора из двух фреймов</title>  
</head>
```

```
<frameset cols="140,*">
<frame name="menu" src="menu.htm">
<frame name="main" src="welcome.htm">
</frameset>
</html>
```

2. Загрузите страницу в браузер и посмотрите, что получилось.
3. Измените значения атрибутов тэгов <frameset> и <frame> и посмотрите, как изменяется внешний вид фреймов, загружая страницу в браузер.
4. Создайте страницы menu.htm и welcome.htm для более наглядной работы этого примера.
5. Создайте набор фреймов, состоящий из двух меню: в верхней и правой части экрана, а также фрейма для показа содержательных страниц в левой нижней части экрана. Например, так:

```
<html>
<head> <title>Пример набора из двух
фреймов</title>
</head>
<frameset cols="*,150">
<frameset rows="104,*">
<frame name="top" src="top.htm" noresize scrolling=no
marginheight=5 marginwidth=5>
<frame name="main" src="welcome.htm" noresize scrolling=auto
marginheight=5 marginwidth=5>
</frameset>
<frame name="right" src="right.htm">
</frameset>
</html>
```

6. Создайте страницы top.htm и right.htm для более наглядной работы этого примера.

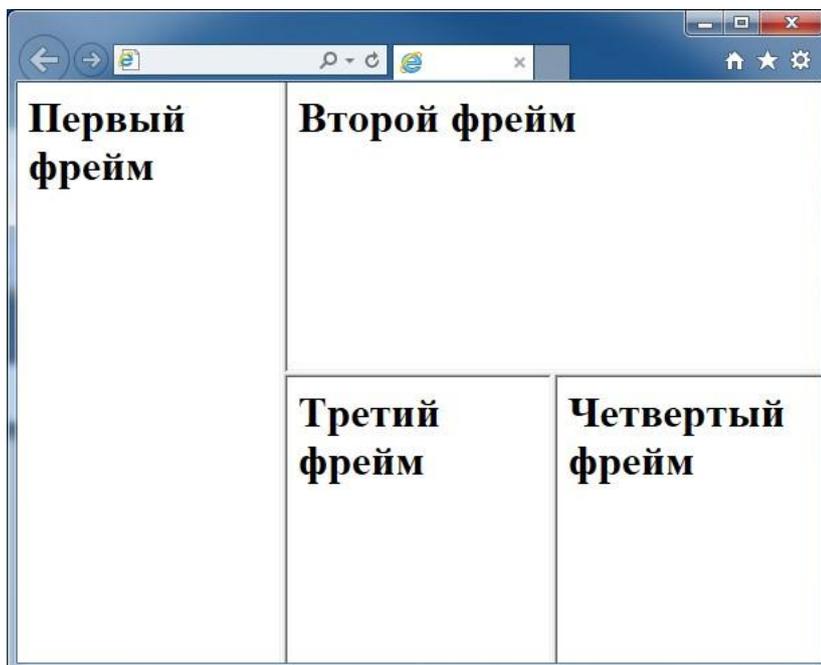
### **Задания для самостоятельной работы Задание**

#### **1.**

Распространенное явление - комбинация вертикальных и горизонтальных фреймов.

```
<frameset cols="*, 55%"> символ * означает все оставшееся место
<frame src="homepage.html" name="frame1">
<frameset rows="15%, 15%, 70%">
<frame src="menu.html" name="frame2">
<frame src="menu2.html" name="frame3">
<frame src="menu3.html" name="frame4">
</frameset>
</frameset>
```

Используя вложенные теги **FRAMESET**, разбейте страницу следующим образом:



### Задание 2.

Используя атрибут **target="..."**, сделать в фреймах гиперссылки, по нажатию на которые в эти же или в другие фреймы произойдет загрузка других страниц, полученных в предыдущих лабораторных работах.

## ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ВОПРОСОВ К ЗАЧЕТУ

1. Системы управления базами данных. Назначение, возможности.
2. Базы данных. Назначение. Преимущества.
3. Понятие реляционной базы данных. Её преимущества.
4. Основные этапы проектирования баз данных.
5. Объекты базы данных, их назначение и взаимосвязь.
6. Таблица. Её структурные элементы. Способы создания и работы с ней.
7. Поле таблицы. Типы данных поля. Свойства поля.
8. Понятие ключевого поля.
9. Поле со списком в таблице. Свойства поля со списком.
10. Запись в таблице. Сортировка и фильтрация записей в таблице.
11. Схема данных. Нарисуйте схему данных и поясните, что на ней изображено.
12. Первичный и внешний ключ. Правила целостности для связанных полей.
13. Связи между таблицами. Типы связей.
14. Главная и подчиненная таблицы. Очередность их заполнения.
15. Целостность данных. Преимущества целостной базы данных.
16. Запрос. Типы запросов. Конструирование запроса.
17. Сортировка и отбор записей в запросе. Построитель выражений.
18. Вычисления в запросе. Вычисляемые поля в запросе. Построитель выражений.
19. Условия отбора в запросе. Формирование запросов с операциями AND и OR.
20. Сортировка и группировка записей в запросе.
21. Форма. Назначение. Структура формы. Способы создания. Работа с формой.
22. Элементы управления, используемые в форме. Построитель выражений.

23. Отчет. Структура отчета. Способы создания и форматирования.
24. Нумерация записей в отчете.
25. Группировка и сортировка записей в отчете.
26. Элементы управления, используемые в отчете. Построитель выражений.
27. Управляющая форма. Назначение. Свойство Источник записей формы.
28. Элементы управления, используемые в управляющей форме. Построитель выражений.
29. Защита информации в базе данных.
30. Информационно-поисковый язык и информационно-поисковый словарь.
31. Подсистемы информационной системы.
32. Функции информационно-поисковых систем.
33. Обзор современных информационно-поисковых систем.
34. Справочно-правовая система «Консультант Плюс».
35. Справочная правовая система «Гарант».
36. Глобальная сеть Интернет.
37. Адресация в интернете.
38. Подключение к интернету.

# ПРИМЕРНЫЕ ТЕСТОВЫЕ ЗАДАНИЯ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ АТТЕСТАЦИИ

## Тест к теме «СУБД Microsoft Access»

### Вопрос № 1. СУБД - это:

1. Оболочка операционной системы, позволяющая более комфортно работать с файлами
2. Набор программ, обеспечивающих работу всех аппаратных устройств компьютера и доступ пользователя к ним
3. Программы, которые позволяют создавать БД, хранить их, модифицировать и т.д.
4. Программы по обслуживанию БД

### Вопрос № 2. База данных - это:

1. Совокупность данных со строгой внутренней организацией
2. Набор данных большого объема
3. Совокупность таблиц
4. Информация доступная для использования группой людей

### Вопрос № 3. Ключевое поле базы данных - это:

1. Поле типа "Счетчик"
2. Поле, значения которого могут быть типа "Счетчик" или "Числовой"
3. Поле, значения которого уникальны
4. Поле, значения которого логического типа

**Вопрос № 4.** Дана следующая база данных:

№ п/п	Фамилия	Имя	Отчество	Год рождения	Должность	Оклад
1.	Иванов	Федор	Сергеевич	1955	Переводчик	7500 руб.
2.	Петров	Виктор	Николаевич	1964	Консультант	8800 руб.
3.	Попов	Вадим	Валерьевич	1981	Кассир	5500 руб.
4.	Румянцева	Валентина	Рудольфовна	1958	Бухгалтер	11200 руб.
5.	Бабенко	Григорий	Иванович	1955	Директор	12000 руб.

Какое поле может являться ключевым?

1. Оклад
2. Фамилия
3. Должность
4. № п/п
5. Год рождения

**Вопрос № 5.** Запись табличной базы данных - это:

1. Совокупность столбцов таблицы
2. Столбец таблицы
3. Строка таблицы
4. Совокупность строк таблицы

**Вопрос № 6.** Поле табличной базы данных - это:

1. Совокупность столбцов таблицы
2. Столбец таблицы
3. Совокупность строк таблицы
4. Строка таблицы

**Вопрос № 7.** В записи реляционной базы данных может содержаться:

1. Неоднородная информация (данные разных типов)
2. Исключительно однородная информация (данные только одного типа)
3. Исключительно числовая информация
4. Только текстовая информация

**Вопрос № 8.** Дана следующая база данных:

№ п/п	Фамилия	Имя	Отчество	Год рождения	Должность	Оклад
1.	Иванов	Федор	Сергеевич	1955	Переводчик	7500 руб.
2.	Петров	Виктор	Николаевич	1964	Консультант	8800 руб.
3.	Попов	Вадим	Валерьевич	1981	Кассир	5500 руб.
4.	Румянцева	Валентина	Рудольфовна	1958	Бухгалтер	11200 руб.
5.	Бабенко	Григорий	Иванович	1955	Директор	12000 руб.

Количество полей в ней равно?

1. 5
2. 6
3. 7
4. 4

**Вопрос № 9.** Дана следующая база данных:

№ п/п	Фамилия	Имя	Отчество	Год рождения	Должность	Оклад
1.	Иванов	Федор	Сергеевич	1955	Переводчик	7500 руб.
2.	Петров	Виктор	Николаевич	1964	Консультант	8800 руб.
3.	Попов	Вадим	Валерьевич	1981	Кассир	5500 руб.
4.	Румянцева	Валентина	Рудольфовна	1958	Бухгалтер	11200 руб.
5.	Бабенко	Григорий	Иванович	1955	Директор	12000 руб.

Количество записей в ней равно?

1. 4
2. 6
3. 7
4. 5

**Вопрос № 10.** Предположим, что некоторая база данных содержит поля "Фамилия", "Год рождения", "Доход". При поиске по условию (Год рождения 1958 and Доход) будут найдены фамилии лиц:

1. Имеющих доход менее 3500 и тех, кто родился в 1958 году и позже
2. Имеющих доход менее 3500 или тех, кто родился в 1958 и позже
3. Имеющих доход менее 3500 или тех кто родился позже 1958 года
4. Имеющих доход менее 3500 и тех кто родился позже 1958 года

**Вопрос № 11.** Содержит ли какую-либо информацию таблица, в которой нет полей?

1. Содержит информацию о структуре базы данных
2. Не содержит никакой информации
3. Таблица без полей существовать не может
4. Содержит информацию о будущих записях

**Вопрос № 12.** Таблицы в базах данных предназначены для...

1. Хранения данных базы
2. Отбора и обработки данных базы
3. Ввода данных базы и их просмотра
4. Автоматического выполнения группы команд

**Вопрос № 13.** Что из перечисленного не является объектом Access?

1. Таблицы
2. Ключи
3. Формы
4. Запросы

**Вопрос № 14.** Для чего предназначены запросы?

1. Хранения данных базы
2. Ввода данных базы и их просмотра
3. Автоматического выполнения группы команд
4. Отбора и обработки данных базы

**Вопрос № 15.** Для чего предназначены формы?

1. Хранения данных базы
2. Ввода данных базы и их просмотра
3. Отбора и обработки данных базы
4. Автоматического выполнения группы команд

**Вопрос № 16.** В каком диалоговом окне создают связи между полями таблиц базы данных?

1. Схема данных
2. Таблица связей
3. Схема связей
4. Таблица данных

**Вопрос № 17.** Без каких объектов не может существовать база данных?

1. Без отчетов
2. Без форм
3. Без запросов
4. Без таблиц

**Вопрос № 18.** Содержит ли какую-либо информацию таблица, в которой нет ни одной записи?

1. Пустая таблица не содержит никакой информации
2. Пустая таблица содержит информацию о структуре базы данных
3. Пустая таблица содержит информацию о будущих записях
4. Таблица без записей существовать не может

**Вопрос № 19.** В чем состоит особенность поля типа «Счетчик»?

1. Служит для ввода числовых данных
2. Имеет ограниченный размер
3. Имеет свойство автоматического наращивания
4. Служит для ввода действительных чисел

**Вопрос № 20.** Какую строку будет занимать запись Pentium после проведения сортировки по возрастанию в поле ОП?

Компьютер	ОП	Винчестер
Pentium	16	800 Мб
386DX	4	300 Мб
486DX	8	500 Мб
Pentium Pro	32	2 Гб

1. 1
2. 2
3. 3
4. 4

## Тест к теме «Глобальная сеть Интернет»

**Вопрос № 1.** Какой из способов подключения к Интернет обеспечивает наибольшие возможности для доступа к информационным ресурсам?

1. постоянное соединение по оптоволоконному каналу
2. удаленный доступ по коммутируемому телефонному каналу
3. постоянное соединение по выделенному телефонному каналу
4. терминальное соединение по коммутируемому телефонному каналу

**Вопрос № 2.** Модем - это...

1. почтовая программа
2. сетевой протокол
3. сервер Интернет
4. техническое устройство

**Вопрос № 3.** Модем, передающий информацию со скоростью 28 800 бит/с, может передать две страницы текста (3 600 байт) в течение...

1. 1 минуты
2. 1 часа
3. 1 секунды
4. 1 дня

**Вопрос № 4.** Электронная почта (e-mail) позволяет передавать...

1. только сообщения
2. только файлы
3. сообщения и приложенные файлы
4. видеоизображения

**Вопрос № 5.** Какой протокол является базовым в Интернет?

1. HTTP
2. HTML
3. TCP
4. TCP/IP

**Вопрос № 6.** Компьютер, подключенный к Интернет, обязательно имеет...

1. IP-адрес
2. Web-сервер
3. домашнюю web-страницу
4. доменное имя

**Вопрос № 7.** Гиперссылки на web - странице могут обеспечить переход...

1. только в пределах данной web - страницы
2. только на web - страницы данного сервера
3. на любую web - страницу данного региона
4. на любую web - страницу любого сервера Интернет

**Вопрос № 8.** Задан адрес электронной почты в сети Internet:

user\_name@int.glasnet.ru. Каково имя владельца электронного адреса?

1. int.glasnet.ru
2. user\_name
3. glasnet.ru
4. ru

**Вопрос № 9.** Браузеры (например, Microsoft Internet Explorer) являются...

1. серверами Интернет
2. антивирусными программами
3. трансляторами языка программирования
4. средством просмотра web-страниц

**Вопрос № 10.** Web-страницы имеют формат (расширение)...

1. \*.txt
2. \*.htm
3. \*.doc
4. \*.exe

**Вопрос № 11.** Модем - это устройство, предназначенное для ...

1. вывода информации на печать
2. хранения информации
3. обработки информации в данный момент времени
4. передачи информации по телефонным каналам связи

**Вопрос № 12.** Количество пользователей Интернет во всем мире составляет примерно ...

1. 1 млн.
2. 10 млн.
3. 50 млн.
4. 200 млн.

**Вопрос № 13.** В качестве гипертекстовых ссылок можно использовать ...

1. только слово
2. только картинку
3. любое слово или любую картинку
4. слово, группу слов или картинку, при подведении мыши к которым ее курсор принимает форму человеческой руки

**Вопрос № 14.** Web-страница - это ...

1. документ, в котором хранится информация сервера
2. документ, в котором хранится вся информация по сети
3. документ, в котором хранится информация пользователя
4. сводка меню программных продуктов

**Вопрос № 15.** Адресация - это ...

1. количество бод (символов/сек), пересылаемой информации модемом
2. способ идентификации абонентов в сети
3. адрес сервера
4. почтовый адрес пользователя сети

**Вопрос № 16.** Скорость передачи информации по магистральной оптоволоконной линии обычно составляет не меньше, чем ...

1. 28,8 бит/с
2. 56,6 Кбит/с
3. 100 Кбит/с
4. 1 Мбит/с

**Вопрос № 17.** Какой из адресов соответствует домену второго уровня?

1. [www.fizika.ru](http://www.fizika.ru)
2. [interweb.spb.ru/present](http://interweb.spb.ru/present)
3. [www.junior.ru/nikolaeva](http://www.junior.ru/nikolaeva)
4. [www.junior.ru/nikolaeva/word.htm](http://www.junior.ru/nikolaeva/word.htm)

**Вопрос № 18.** Компьютерные телекоммуникации - это ...

1. соединение нескольких компьютеров в единую сеть
2. перенесение информации с одного компьютера на другой с помощью дискет
3. дистанционная передача данных с одного компьютера на другой
4. обмен информацией между пользователями о состоянии работы компьютера

**Вопрос № 19.** Домен - это ...

1. единица измерения информации
2. часть адреса, определяющая адрес компьютера пользователя в сети
3. название программы, для осуществления связи между компьютерами
4. название устройства, осуществляющего связь между компьютерами

**Вопрос № 20.** Задан адрес электронной почты в сети Интернет: user\_name@mtu-net.ru Каково имя компьютера, на котором хранится почта?

1. mtu-net.ru
2. ru
3. mtu-net
4. user\_name

Полковская Марина Николаевна  
Бузина Татьяна Сергеевна

**Учебное пособие**

**по информационно-коммуникационным технологиям**  
для студентов направлений подготовки 13.03.01 Теплоэнергетика и  
теплотехника, 13.03.02 Электроэнергетика и электротехника, 35.03.06  
Агроинженерия

Корректурa и редактирование авторов  
Компьютерный набор М.Н. Полковская

Лицензия на издательскую деятельность  
ЛР № 0704444 от 11.03.98 г.  
Формат 60x84. Печ.л. 3,75.  
подписано в печать 25.05.2020.  
Тираж 100 экз.

Издательство Иркутского ГАУ  
664038, Иркутская обл., Иркутский р-н,  
пос. Молодежный