

**МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ  
РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
"БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА"**

**Кафедра "Информационное и математическое обеспечение  
транспортных систем"**

**Т. Л. ШИНКЕВИЧ, Т. А. ГОЛДОБИНА**

# **БАЗЫ ДАННЫХ В ТАМОЖЕННОМ ДЕЛЕ**

**Учебно-методическое пособие**

**Гомель 2016**



МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ  
РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
"БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА"

Кафедра "Информационное и математическое обеспечение  
транспортных систем"

Т. Л. ШИНКЕВИЧ, Т. А. ГОЛДОБИНА

# БАЗЫ ДАННЫХ В ТАМОЖЕННОМ ДЕЛЕ

*Одобрено научно-методической комиссией  
гуманитарно-экономического факультета  
в качестве учебно-методического пособия для лабораторных занятий  
студентов специальности «Таможенное дело»*

Гомель 2016

УДК 004.65 (075.8)  
ББК 32.81  
Ш62

Рецензент – доцент кафедры "Информационное и математическое обеспечение транспортных систем", канд. экон. наук, *Л. А. Иоффе* (УО "БелГУТ")

**Шинкевич, Т. Л.**

Ш62 Базы данных в таможенном деле : учеб.-метод. пособие / Т. Л. Шинкевич, Т. А. Голдобина ; М-во трансп. и коммуникаций Респ. Беларусь, Белорус. гос. ун-т трансп. – Гомель : БелГУТ, 2016. – 78 с.  
ISBN 978-985-554-560-7

Приведены краткие теоретические сведения, методические указания и справочные материалы, раскрывающие основы проектирования, разработки и использования баз данных в СУБД *MS Access 2010* в таможенном деле. Включает варианты индивидуальных заданий и контрольные вопросы.

Предназначено для студентов I курса специальности "Таможенное дело" дневной формы обучения. Рекомендуется также студентам других специальностей в качестве дополнительного учебного материала по основам проектирования и разработки баз данных в СУБД *MS Access 2010*.

**УДК 004.65 (075.8)**  
**ББК 32.81**

**ISBN 978-985-554-560-7**

© Шинкевич Т. Л., Голдобина Т. А., 2016  
© Оформление. УО «БелГУТ», 2016

## О Г Л А В Л Е Н И Е

<b>Введение</b> .....	5
<b>1 Проектирование базы данных</b> .....	6
1.1 Общие сведения о проектировании баз данных .....	6
1.2 Нормализация данных, нормальные формы.....	7
1.3 Проектирование базы данных «Таможенный контроль» .....	8
<b>2 Создание базы данных</b> .....	14
2.1 Запуск MS Access .....	14
2.2 Особенности интерфейса MS Access 2010.....	15
<b>3 Разработка таблиц базы данных</b> .....	17
3.1 Создание и сохранение таблицы .....	17
3.2 Типы данных.....	19
3.3 Выбор ключевого поля.....	19
3.4 Свойства полей.....	20
3.5 Задание маски ввода .....	21
3.6 Задание свойства Подстановка.....	22
3.7 Межтабличные связи .....	23
3.8 Создание схемы данных .....	23
3.9 Задание условий целостности .....	25
3.10 Работа с таблицами .....	25
Порядок выполнения работы.....	26
Контрольные вопросы .....	29
<b>4 Разработка форм. Ввод информации в базу данных</b> .....	31
4.1 Создание форм .....	31
4.1.1 Создание простой формы в режиме Конструктора .....	32
4.1.2 Мастер форм .....	34
4.1.3 Создание подчиненной формы.....	34
4.2 Изменение форм.....	35
4.2.1 Изменение структуры формы в режиме Конструктора.....	35
4.2.2 Элементы управления .....	36
4.2.3 Создание заголовка формы .....	37
4.2.4 Создание вкладок на форме.....	38
Порядок выполнения работы.....	39
Контрольные вопросы .....	41
<b>5 Создание запросов к базе данных</b> .....	42
5.1 Общие сведения о запросах .....	42
5.2 Создание запросов на выборку.....	43
5.2.1 Создание запроса с помощью Мастера.....	43
5.2.2 Создание запроса на выборку в режиме Конструктора.....	44
5.2.3 Структура бланка запроса.....	45
5.2.4 Формирование условий отбора в запросах на совпадение.....	45

5.2.5	Параметрический запрос.....	46
5.3	Вычисления в запросах .....	47
5.3.1	Создание итоговых запросов в Конструкторе.....	47
5.3.2	Создание итоговых запросов с помощью Мастера.....	48
5.3.3	Построитель выражений. Создание вычисляемых полей в запросах.....	49
5.3.4	Использование встроенных функций для вычислений .....	50
5.4	Перекрестные запросы .....	51
5.4.1	Создание перекрестного запроса в Конструкторе .....	52
5.4.2	Создание перекрестного запроса с помощью Мастера .....	53
5.5	Запросы на изменение .....	53
5.5.1	Разрешения на выполнение запросов манипулирования данными в MS Access 2010.....	53
5.5.2	Формирование запроса на удаление записей из таблиц базы данных .....	54
5.5.3	Создание запроса на обновление данных в таблице.....	55
	Порядок выполнения работы.....	56
	Задания для самостоятельной работы .....	61
	Контрольные вопросы .....	63
<b>6</b>	<b>Отчеты.....</b>	<b>64</b>
6.1	Способы создания отчетов.....	64
6.2	Панель элементов.....	66
6.3	Создание отчета при помощи Мастера .....	67
6.4	Создание вычисляемых и итоговых полей .....	68
6.5	Создание диаграммы в отчете .....	68
	Порядок выполнения работы.....	69
	Контрольные вопросы .....	70
<b>7</b>	<b>Кнопочная форма .....</b>	<b>71</b>
7.1	Создание макросов.....	71
7.2	Главная кнопочная форма.....	72
	Порядок выполнения работы.....	74
	Контрольные вопросы .....	77
	<b>Список использованной и рекомендуемой литературы.....</b>	<b>78</b>

## ВВЕДЕНИЕ

Информация – самое ценное достижение человечества. Накапливаемые данные и знания необходимо сохранять, обрабатывать и результативно использовать. В современных экономических условиях успешное функционирование различных фирм, организаций и предприятий невозможно без развитой *информационной системы (ИС)*, которая позволяет автоматизировать сбор и обработку данных, помогает эффективно управлять предприятием.

Ядром современной системы управления предприятием являются *базы данных (БД)* и *системы управления базой данных (СУБД)*.

*База данных (БД)* – поименованная и организованная (структурированная) совокупность взаимосвязанных данных, которые отражают состояние объектов и их отношений в рассматриваемой предметной области и находятся под центральным программным управлением.

*Предметная область* – это некоторая область человеческой деятельности или часть реального мира, на основе которой создается БД и её структура. База данных содержит такой минимально необходимый объем информации, который позволяет удовлетворять запросы каждого из многочисленных пользователей.

Доступ пользователя к базе данных и все операции с данными реализуются программой, называемой *системой управления базами данных (СУБД)*. СУБД – это совокупность языковых и программных средств, предназначенных для создания, наполнения, обновления и удаления баз данных.

Организационной основой разработки любой системы служит ее *жизненный цикл*. Жизненный цикл информационной системы включает три основные стадии:

- проектирование;
- программную реализацию;
- эксплуатацию.

## 1 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

**Реляционная база данных** – это структура для хранения взаимосвязанной информации.

Все данные такой информационной структуры представлены в виде *двумерных таблиц*, каждая из которых содержит некоторое количество строк, называемых *записями*, и один или несколько столбцов, называемых *полями*.

### 1.1 Общие сведения о проектировании баз данных

*Проектирование баз данных* – это многоэтапный процесс, который включает: анализ информационной модели предметной области, выделение информационных объектов данной предметной области, определение их атрибутов, выделение ключевых атрибутов, описание типов данных, определение связей между информационными объектами.

Задачи проектирования БД:

- обеспечение хранения в БД всей необходимой информации;
- обеспечение получения требуемой информации по всем запросам;
- обеспечение целостности данных, правильности их содержания;
- сокращение избыточности и повтора данных;
- недопустимость наличия противоречий в данных;
- исключение потери данных.

Проектирование БД разбивают на два основных этапа:

- *инфологическое* (концептуальное);
- *даталогическое*, которое, в свою очередь, подразделяется на *логическое* и *физическое* проектирование.

На этапе *инфологического проектирования* выполняется изучение и описание предметной области, построение формализованной модели предметной области. Такая модель строится с использованием стандартных языковых средств, обычно графических, например, ER-диаграмм – диаграмм «Сущность-связь» (от англ. Entity-Relationship, т. е. сущность-связь). В них сущности изображаются в виде прямоугольников или ромбов, а связи между ними – линиями со стрелками, над которыми проставляется степень связи (1 или буква M, заменяющая слово «много»). Такая модель строится без ориентации на какую-либо конкретную СУБД.



*Инфологическая модель* отображает данные предметной области в виде совокупности информационных объектов (сущностей), их реквизитов (атрибутов) и связей между ними.

*Сущность* – любой различимый объект (объект, который мы можем отличить от другого), информацию о котором необходимо хранить в БД. Сущностями могут быть люди, места, товары, самолеты, вкус, цвет и т. д. Каждая сущность имеет несколько экземпляров. Например, сущность ЧЕЛОВЕК может иметь несколько экземпляров: Иванов, Петров, Сидоров и т. д.

*Атрибут* – поименованная характеристика сущности. Атрибуты используются для определения того, какая информация должна быть собрана о сущности. Примерами атрибутов для сущности ЧЕЛОВЕК являются ФАМИЛИЯ, ИМЯ, АДРЕС, ДАТА РОЖДЕНИЯ, НОМЕР ПАСПОРТА и т. д.

*Ключ* – минимальный набор атрибутов, по значениям которых можно однозначно найти требуемый экземпляр сущности. Минимальность означает, что исключение из набора любого атрибута не позволяет идентифицировать сущность по оставшимся. Для сущности ЧЕЛОВЕК ключом является атрибут НОМЕР ПАСПОРТА.

*Связь* – ассоциирование двух или более сущностей. Если бы назначением БД было только хранение отдельных, не связанных между собой данных, то её структура могла бы быть очень простой. Однако одно из основных требований к организации БД – это обеспечение возможности отыскания одних сущностей по значениям других, для чего необходимо установить между ними определенные связи.

*Целью логического проектирования* является выбор конкретной СУБД и преобразование инфологической (концептуальной) модели в логическую. Для реляционной БД этот этап состоит в разработке структуры таблиц, связей между ними и определении ключевых атрибутов. Структура таблицы определяется составом атрибутов сущности. Ключевой атрибут сущности образует уникальный ключ реляционной таблицы. Такой ключ называется *первичным ключом* таблицы.

*Физическое проектирование* – реализация даталогической модели средствами конкретной СУБД, а также выбор решений, связанных с физической средой хранения данных: выбор метода управления дисковой памятью, методов доступа к данным, методов сжатия данных и т. д.

## **1.2 Нормализация данных, нормальные формы**

*Нормализация* представляет собой процесс реорганизации данных путем ликвидации повторяющихся групп и иных противоречий в хранении данных. Таблица находится в данной *нормальной форме*, если она удовлетворяет определенному набору требований. Теоретически существует пять нормальных форм, но на практике обычно используются только первые три. Более

того, первые две нормальные формы являются по существу промежуточными шагами для приведения базы данных к третьей нормальной форме.

О нормализации таблиц в базе данных необходимо заботиться на раннем этапе проектирования БД, так как при введенных данных довольно трудно менять структуру базы. Иногда процесс нормализации порождает добавочные таблицы, которые не были включены в первоначальный проект. Если выяснить это как можно раньше, то не придется напрасно тратить силы на их разработку.

**Первая нормальная форма.** Для того чтобы таблица считалась нормализованной к первой нормальной форме, каждое из ее полей должно быть неделимым и не должно содержать никаких повторяющихся групп. Поле считается неделимым, если оно содержит только один элемент данных. Например, поле *Адрес*, которое содержит не только название улицы, но также и города, почтовый код, не является неделимым. Чтобы соответствовать первой нормальной форме, такие поля должны быть разбиты на несколько полей (название улицы, номер дома, номер квартиры, город, индекс). *Повторяющаяся группа* – это поле, которое повторяется внутри определения записи с целью хранения нескольких значений для атрибута.

**Вторая нормальная форма.** Для приведения таблицы ко второй нормальной форме необходимо, чтобы:

- 1) она находилась в первой нормальной форме;
- 2) все не ключевые поля полностью зависели от первичного ключа таблицы или от каждого поля в первичном ключе, если последний состоит из нескольких полей. Это значит, что каждое не ключевое поле должно однозначно определяться первичным ключом и полями, его составляющими.

**Третья нормальная форма.** Для того чтобы таблица была приведена к третьей нормальной форме, необходимо, чтобы она находилась в первой и второй нормальной формах и чтобы все не ключевые поля полностью зависели от первичного ключа таблицы и не зависели друг от друга.

### 1.3 Проектирование базы данных «Таможенный контроль»

Существует несколько подходов к проектированию баз данных:

- на основе предлагаемых полей – реквизитов, описывающих некоторую предметную область;
- на основе общей постановки;
- на основе бумажных документов.

Исследуем предметную область – работа пунктов пропуска, где происходит таможенный контроль – и построим ее формализованную инфологическую модель (рисунок 1).



Рисунок 1 – Упрощенная инфологическая модель

На основании инфологической модели можно выделить следующие информационные объекты (сущности) и присвоить им формализованные, удобные для дальнейшего использования, имена:

- ПунктыПропуска;
- Сотрудники;
- Перемещения;
- ЛицаПересГраницу.

*Примечания*

1 Сущность Перемещения – это сущность-отношение, которая соединяет сущности ПунктыПропуска и ЛицаПересГраницу.

2 Имена сущностей и их атрибутов применяются при реализации базы данных в конкретной СУБД (*MS Access*) и подлежат дальнейшей программной обработке. Поэтому в них нельзя использовать пробелы и некоторые специальные символы. Для удобства чтения рекомендуется каждое слово записывать с прописной буквы.

Каждую сущность опишем набором атрибутов:

- сущность ПунктыПропуска характеризуется, в частности, номером, адресом и телефоном;
- Сотрудники – табельным номером, фамилией, именем, отчеством и должностью;
- ЛицаПересГраницу – номером паспорта, фамилией, именем, отчеством, адресом и др.;
- Перемещения – номером пункта пропуска, номером паспорта, датой прохождения, видом и номером транспорта, наличием и описанием груза, денежной пошлиной.

Сгруппируем информацию в таблицу 1 для дальнейшего более детального анализа. Вместо сущности ЛицаПересГраницу удобнее использовать карто-теку лиц, пересекающих границу (далее – Картотека).

Таблица 1 – Сущности и их атрибуты

Сущность	Атрибуты
<b>ПунктыПропуска</b>	<b>НомерПунктаПропуска</b> Адрес Телефон
<b>Сотрудники</b>	<b>ТабельныйНомер</b>  Фамилия Имя Отчество Должность
<b>Картотека</b>	<b>НомерПаспорта</b> Фамилия Имя Отчество Адрес
<b>Перемещения</b>	<b>НомерПоПорядку</b>  ДатаПрохождения ВидТранспСредства НомерТранспСредства НаличиеГруза ОписаниеГруза Пошлина

Определим *ключевые атрибуты* для каждой сущности. Для сущности **ПунктыПропуска** ключом является атрибут **НомерПунктаПропуска** (этот атрибут однозначно определяет пункт пропуска в отдельном государстве), для сущности **Сотрудники** ключом служит атрибут **ТабельныйНомер**, для сущности **Картотека** в качестве ключа используем атрибут **НомерПаспорта**, для сущности **Перемещения** добавим ключевое поле **НомерПоПорядку**. Указанные ключевые атрибуты называются *первичным ключом*. По ним устанавливаются связи между сущностями.

Для того чтобы ответить на вопрос «Где дежурит?», к атрибутам сущности **Сотрудники** необходимо добавить атрибут **НомерПунктаПропуска**, который называется *внешним ключом* сущности **Сотрудники** и по нему устанавливается связь с сущностью **ПунктыПропуска** по его первичному ключу – **НомерПунктаПропуска**.

Рассмотрим сущность **Перемещения**. Она характеризуется следующими атрибутами: **НомерПоПорядку**, **НомерПунктаПропуска**, **НомерПаспорта**, **ДатаПрохождения**, **ВидТранспСредства**, **НомерТранспСредства**, **НаличиеГруза**, **ОписаниеГруза**, **Пошлина**. Атрибуты **НомерПунктаПропуска** и **НомерПаспорта** можно назвать внешними ключами сущности **Перемещения**, т. к. по ним можно установить связь с сущностью **ПунктыПропуска** по атрибуту

НомерПунктаПропуска и с сущностью **Картотека** по атрибуту НомерПаспорта. Атрибут НомерПунктаПропуска для сущности **ПунктыПропуска** является первичным ключом (однозначно определяет номер пункта пропуска), атрибут НомерПаспорта для сущности **Картотека** является первичным ключом, т. к. однозначно определяет лицо, пересекающее границу (рисунок 2).

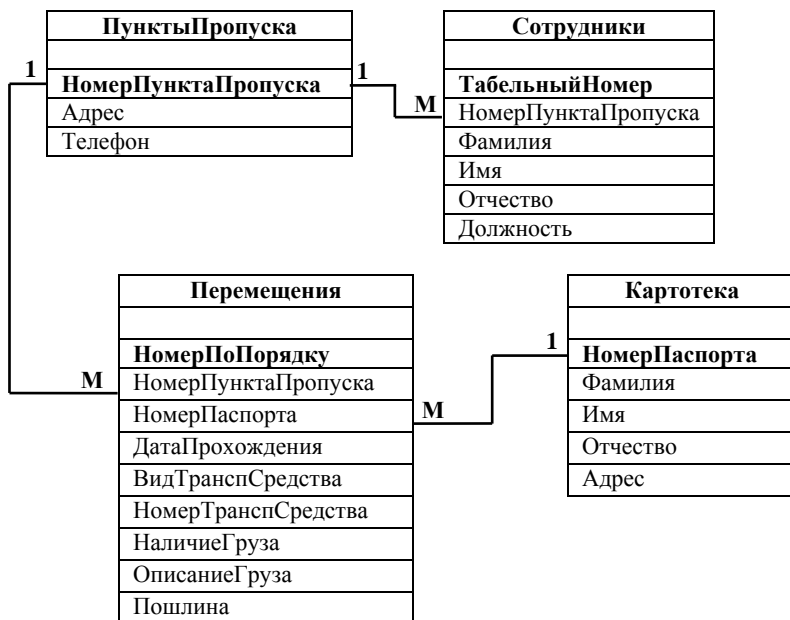


Рисунок 2 – Инфологическая модель базы данных  
«Таможенный контроль»

Установим связи между сущностями по ключу.

На каждом пункте пропуска работает несколько сотрудников. Поэтому связь ПунктыПропуска – Сотрудники имеет тип «один ко многим»: ключевое поле НомерПунктаПропуска сущности ПунктПропуска связывается с неключевым полем сущности Сотрудники.

Связь ПунктыПропуска – Перемещения «один ко многим».

Связь Картотека – Перемещения «один ко многим».

В результате получим информационно-логическую модель (см. рисунок 2).

Учитывая специфику инструментария создания и работы с базой данных, СУБД *MS Access*, приступим к разработке логической структуры взаимосвязанных таблиц. Каждой сущности поставим в соответствие таблицу базы данных, в которой определим ключевые поля, типы данных, хранящихся в базе данных, и свойства полей.

<b>Таблица «ПунктыПропуска»</b>			
<b>Поле</b>	<b>Ключевое поле</b>	<b>Тип поля</b>	<b>Свойства поля</b>
<b>НомерПунктаПропуска</b>	<b>Ключ</b>	<b>Числовое</b>	
Адрес		Текстовое	
Телефон		Текстовое	

<b>Таблица «Сотрудники»</b>			
<b>Поле</b>	<b>Ключевое поле</b>	<b>Тип поля</b>	<b>Свойства поля</b>
<b>ТабельныйНомер</b>	<b>Ключ</b>	<b>Числовое</b>	
НомерПунктаПропуска		Числовое	Мастер подстановок из таблицы «ПунктыПропуска»
Фамилия		Текстовое	
Имя		Текстовое	
Отчество		Текстовое	
Должность		Текстовое	Мастер подстановок (фиксированный набор значений)

<b>Таблица «Картотека»</b>			
<b>Поле</b>	<b>Ключевое поле</b>	<b>Тип поля</b>	<b>Свойства поля</b>
<b>НомерПаспорта</b>	<b>Ключ</b>	<b>Текстовое</b>	
Фамилия		Текстовое	
Имя		Текстовое	
Отчество		Текстовое	
Адрес		Текстовое	

<b>Таблица «Перемещения»</b>			
<b>Поле</b>	<b>Ключевое поле</b>	<b>Тип поля</b>	<b>Свойства поля</b>
<b>НомерПоПорядку</b>	<b>Ключ</b>	<b>Счетчик</b>	
НомерПунктаПропуска		Числовое	Мастер подстановок из таблицы «ПунктыПропуска»
НомерПаспорта		Текстовое	Мастер подстановок из таблицы «Картотека»
ДатаПрохождения		Дата/Время	Маска ввода
ВидТранспСредства		Текстовое	Мастер подстановок (фиксированный набор: машина, мотоцикл и т. д.)
НомерТранспСредства		Числовое	
НаличиеГруза		Логическое	Да/Нет
ОписаниеГруза		Поле МЕМО	
Пошлина		Денежное	



Обратите внимание, что все таблицы приведены к третьей нормальной форме: поля таблиц неделимы и неключевые поля зависят только от ключей в каждой таблице.

После проделанной работы по проектированию базы данных можно приступать к созданию и разработке базы, используя инструмент СУБД *MS Access 2010*.

**Задание.** Проанализируйте заданную предметную область – работу пунктов пропуска. Выделите информационные объекты и обдумайте, какие у них могут быть атрибуты. Изобразите на листе бумаги (самостоятельно) инфологическую модель базы данных Таможенный контроль. Сравните ее с представленной на рисунке 2. Предложите дополнительные атрибуты рассмотренных сущностей или иные изменения инфологической модели и обсудите их с преподавателем.

## 2 СОЗДАНИЕ БАЗЫ ДАННЫХ

### 2.1 Запуск MS Access

Для запуска *MS Access 2010* в главном меню *Windows* (кнопка **Пуск**) необходимо выбрать **Все программы / Microsoft Office / Microsoft Access 2010**. Откроется окно (рисунок 3), в правой части которого предусмотрен блок **Новая база данных**. Далее следует указать папку для сохранения базы данных с помощью кнопки **Открыть** – , имя базы данных и нажать кнопку **Создать** – . Сохранить файл базы данных надо сразу, до создания таблиц. При этом создается пустая база данных, в которую впоследствии можно поместить таблицы, формы, запросы, отчеты и другие объекты.

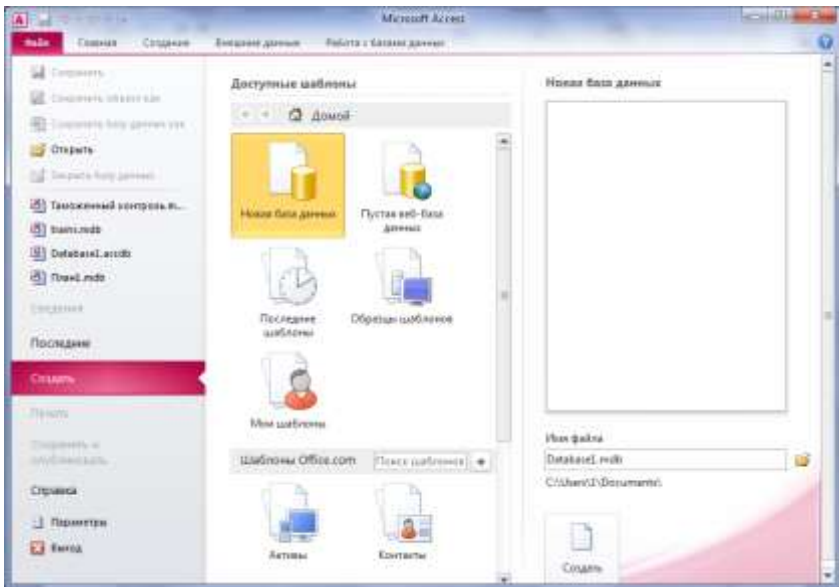


Рисунок 3 – Окно создания базы данных



Если базу данных планируется переносить на другой компьютер, открывать и использовать в более ранних версиях *MS Access*, то перед созданием файла базы данных выполняют **Файл / Параметры / Общие**, затем в списке *Формат файла по умолчанию для пустой базы данных* выбирают: *Access 2002–2003*. Рекомендуется также указать рабочий каталог для сохранения по умолчанию.

*Примечание* – В некоторых версиях *MS Access* параметры по умолчанию вступают в силу только при повторном запуске приложения. Поэтому после создания пустой базы данных следует убедиться, что она сохранена в требуемом формате. Файлы *MS Access 2002–2003* имеют расширение **\*.mdb**, а более поздних версий – **\*.accdb**.

## 2.2 Особенности интерфейса MS Access 2010

В верхней части окна расположен аналог главного меню более ранних версий *MS Office* – лента. *Лента* – набор вкладок, которые можно изменять (рисунок 4).



Рисунок 4 – Вкладка *Создание* ленты СУБД *MS Access*

В левой части ленты *MS Access* находится кнопка/вкладка **Файл** красного цвета, содержащая иерархическое меню.

Лента может присутствовать на экране постоянно, либо появляться после щелчка на заголовке вкладки. Смена режима отображения ленты происходит после щелчка на кнопке, расположенной слева от знака вопроса (рисунок 5).

Каждая вкладка содержит набор команд, относящихся к одной категории. Некоторые вкладки видны всегда, а некоторые появляются по мере необходимости. Так, вкладки **Файл**, **Главная**, **Создание**, **Внешние данные** и **Работа с базами данных** присутствуют во всех режимах.

Вкладка **Главная** содержит пункты меню, касающиеся правки, форматирования, сортировки, фильтрации, поиска, управления окнами. Вкладка **Создание** позволяет создавать новые объекты базы данных – таблицы, формы,

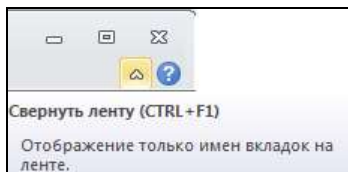



Рисунок 5 – Изменение режимов отображения ленты

запросы, отчеты, макросы. Вкладка **Внешние данные** включает операции импорта/экспорта. Работа с базами данных – это такие инструменты, как схема

данных, переход в редактор встроенного языка программирования VBA, анализ быстродействия и т. д.

Слева в окне *MS Access* находится навигационная панель, предоставляющая доступ к объектам базы данных. Щелчок по кнопке  выводит на экран меню (рисунок 6), в котором можно указать способ отображения объектов. Если выбран пункт **Все объекты Access**, то на навигационной панели будут отражаться все созданные объекты базы данных. Если выбрать пункт **Таблицы**, то будут отображаться только созданные таблицы и т. д.

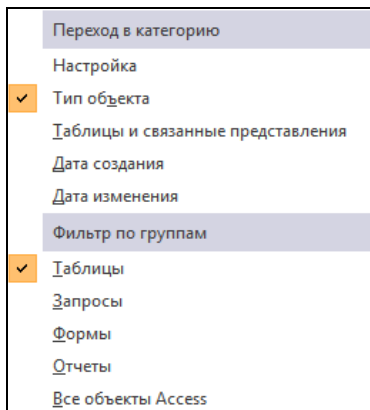


Рисунок 6 – Меню навигационной панели

## 3 РАЗРАБОТКА ТАБЛИЦ БАЗЫ ДАННЫХ

### 3.1 Создание и сохранение таблицы

В *MS Access* вся информация хранится в виде таблиц. Таблицей является совокупность данных, которые относятся к одному информационному объекту. Каждому информационному объекту соответствует отдельная таблица, что позволяет избежать повторения сохраняемых данных. Каждая строка таблицы называется *запись*, а столбец – *поле*. Запись содержит набор данных об одном объекте (например, номер паспорта, фамилию, имя, отчество, адрес), а поле – однородные данные обо всех объектах (например, номера паспортов всех лиц, пересекающих границу). Количество полей в записи и их тип определяются в процессе проектирования базы данных.

В *MS Access 2010* создать таблицу можно двумя основными способами:

- путем ввода данных – кнопка **Таблица**, расположенная на вкладке **Создание** ленты (рисунок 7, а);
- в режиме Конструктора – кнопка **Конструктор таблиц** вкладки **Создание** (рисунок 7, б).

Команда **Списки SharePoint** используется для создания таблиц на удаленном сервере (рисунок 7, в).

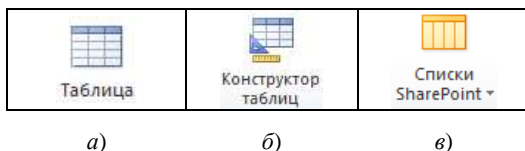


Рисунок 7 – Инструменты *MS Access* для создания таблиц

Для создания таблицы на вкладке **Создание** щелкают по кнопке **Конструктор таблиц**.

Конструктор таблиц содержит три столбца (рисунок 8):

- Имя поля – заголовок столбца;
- Тип данных – формат содержащихся в столбце данных;
- Описание – комментарий, который можно написать для себя, чтобы не забыть, какие данные будут храниться в этом поле.

При создании структуры таблицы для каждого поля в окне Конструктора последовательно вводятся имя, затем тип данных. По желанию можно ввести произвольное описание поля в столбец **Описание**. Переход к следующей ячейке строки осуществляется с помощью клавиши **Tab** или щелчком мыши на этой ячейке. Переход к заполнению следующего поля осуществляется клавишами **Tab**, **Enter** или щелчком мыши.

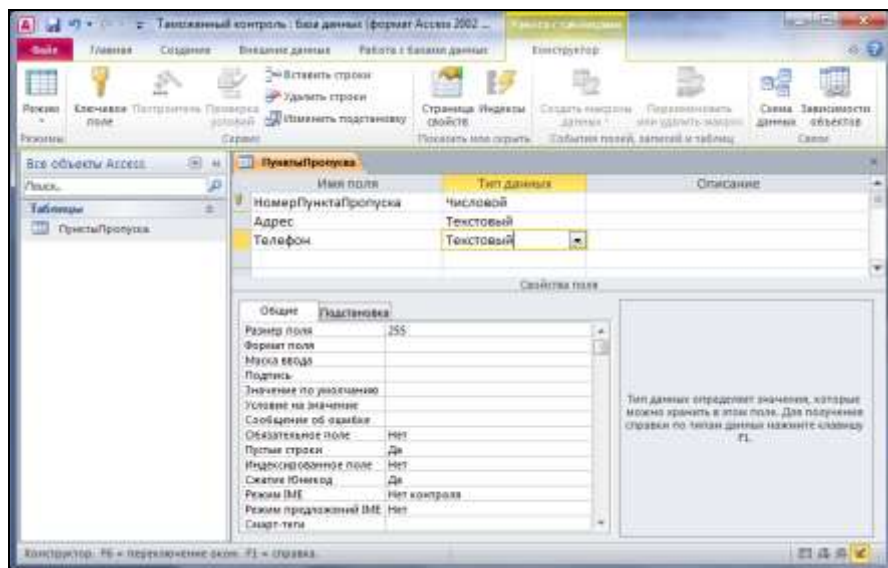


Рисунок 8 – Создание таблицы в режиме Конструктора

При закрытии окна Конструктора таблицы *MS Access* напомним, что созданную структуру таблицы надо сохранить. Для сохранения структуры выбирают пункт *Да* и вводят имя таблицы. При выборе пункта *Нет MS Access* не позволит выйти из режима Конструктора.

СУБД *MS Access* позволяет задавать объектам баз данных любые имена. Но существует неофициальное соглашение Лежински-Реддика, следуя которому желательно делать имена объектов понятными и для других людей. По соглашению, имя объекта начинается тегом, идентифицирующим тип объекта: тбл(tbl), фрм(frm), отч(rpt), зпр(qry) и др. Первая буква имени должна быть прописной, например тблСотрудники, тблПеремещения.

*Примечание* – Имена полей и других объектов базы данных на практике часто записывают латинскими буквами. Знаки препинания, пробелы и другие спецсимволы недопустимы.

## 3.2 Типы данных

*MS Access* позволяет выбрать один из следующих типов данных:

- **Текстовый** – предназначен для хранения обычного неформатированного текста ограниченного размера (до 255 символов);
- **Поле МЕМО** – специальный тип данных для хранения больших объемов текста (до 65 535 символов). Физически текст хранится не в поле, а в другом месте базы данных. В поле хранится указатель на него, но для пользователя такое разделение заметно не всегда;
- **Числовой** – для хранения действительных чисел;
- **Дата/время** – для хранения календарных дат и текущего времени;
- **Денежный** – для хранения денежных сумм;
- **Счетчик** – специальный тип данных для уникальных (не повторяющихся в поле) натуральных чисел с автоматическим наращиванием. Естественное использование – для порядковой нумерации записей;
- **Логический** – для хранения логических данных (могут принимать только два значения, например, *Да* или *Нет*);
- **Поле объекта OLE** – специальный тип данных, предназначенный для хранения объектов OLE, например, мультимедийных. Реально, конечно, такие объекты в таблице не хранятся. Как и в случае с полями МЕМО, они находятся в другом месте внутренней структуры файла базы данных, а в таблице хранятся только указатели на них (иначе работа с таблицами была бы чрезвычайно замедленной);
- **Гиперссылка** – специальное поле для хранения URL-адресов Web-объектов Интернета. При щелчке на ссылке автоматически происходит запуск браузера и воспроизведение объекта в его окне;
- **Мастер подстановок** – это не тип данных, а специальный объект, с помощью которого можно автоматизировать ввод данных в поле так, чтобы не вводить их вручную, а выбирать из раскрывающегося списка.

По умолчанию устанавливается тип данных *Текстовый*. Для изменения необходимо щелкнуть мышью на строке модифицируемого поля в ячейке столбца *Тип данных*, затем выбрать из раскрывающегося списка требуемый тип данных.

## 3.3 Выбор ключевого поля

В реляционной базе данных таблицы должны быть связаны друг с другом. Эта связь устанавливается с помощью *уникальных полей*, значения в которых не могут повторяться. Например, в базе данных «Таможенный контроль» в таблице ПунктыПропуска таким уникальным полем может быть НомерПунктаПропуска. Поле, которое однозначно идентифицирует запись в таблице, называется *первичным ключом*.

Для задания ключевого поля достаточно:

1) щелчком мыши слева от столбца Имя поля выделить поле, выбранное в качестве ключевого;



2) нажать кнопку **Ключевое поле** на панели инструментов или щелкнуть правой клавишей мыши и в открывшемся контекстном меню выбрать пункт *Ключевое поле* (рисунок 9).

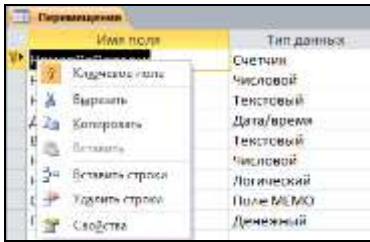


Рисунок 9 – Задание ключевого поля

ное поле, например, порядковый номер записи, присвоив ему тип данных *Счетчик*.

Если к моменту выхода из режима Конструктора первичный ключ для создаваемой таблицы объявлен не будет, *MS Access* выдаст запрос о необходимости включения в таблицу поля первичного ключа с вариантами ответа *Да* или *Нет*.

Для ключевого поля *MS Access* автоматически устанавливает в строке свойства **Обязательное поле** значение *Да (Yes)*, а в строке свойства **Индексированное поле** – значение *Да (совпадение не допускается)*. Если в качестве первичного ключа служит группа из нескольких полей, то все поля являются обязательными, но совпадения внутри каждого из них допускаются.

### 3.4 Свойства полей

Каждое поле имеет свой набор свойств, от которых зависит, как содержимое этого поля отображается на экране, хранится в базе данных, проверяется или обрабатывается. Некоторые свойства являются общими для всех типов полей, другие относятся исключительно к отдельным типам данных.

Некоторые свойства задаются по умолчанию. Свойства полей не являются обязательными. Их можно настраивать по желанию или вообще не задавать.

Все типы полей, кроме полей счетчика и объекта OLE, имеют следующие основные свойства:

- **Формат поля** – определяет, как должно отображаться содержимое поля;

- **Подпись** – определяет текст, который будет отображаться рядом с полем или в отчете;
- **Значение по умолчанию** – определяет значение, которое вводится в поле автоматически при добавлении новой записи;
- **Условие на значение** – определяет область или диапазон значений данных, допустимых в поле;
- **Сообщение об ошибке** – задает текст сообщения, которое появляется в строке состояния при попытке ввести в поле недопустимое значение;
- **Обязательное поле** – содержит значение *Да* или *Нет* в зависимости от того, обязательно ли должно это поле иметь какую-либо запись.

Следующие свойства относятся, в основном, к *текстовым полям*:

- **Размер поля** – максимально допустимое количество символов для данного поля;
- **Маска ввода** – обеспечивает ввод данных в указанном формате;
- **Индексированное поле** – указывает *MS Access*, должны ли создаваться индексы для этого поля. Индексирование поля ускоряет процесс сортировки, поиска и фильтрации содержимого. Поле, являющееся первичным ключом таблицы, индексировано всегда. Стоит задавать индексацию тех полей, по которым возможно проведение сортировки в формах или отчетах;
- **Сжатие Unicode** – представляет собой схему кодировки всех известных символов, для чего вместо одного байта каждый символ представлен двумя. Когда эта функция отключена, *MS Access* при сохранении компрессирует все символы, начинающиеся с байта 0, и распаковывает их при обращении к ним.

Поля числового и денежного типов данных имеют одно дополнительное свойство – *число десятичных знаков*, которое определяет, какое количество десятичных знаков после запятой будет отображаться на экране и храниться в поле.

Чтобы установить свойство поля, надо выделить строку поля. При этом в левой нижней части окна Конструктора откроется список свойств данного поля. Далее следует щелчком мыши установить курсор на строку свойства и выбрать необходимое значение.

### 3.5 Задание маски ввода


При вводе данных, например, в поле ДатаПрожождения, было бы удобно, чтобы в этом поле заранее вводилась точка в качестве разделителя – тогда ее не надо будет вводить вручную.

Форматы для ввода данных называются *масками ввода*.

В состав *MS Access* входит инструмент *Мастер ввода масок*. Чтобы задать маску ввода, необходимо:

- 1) установить курсор на поле, для которого надо задать маску ввода;

2) щелкнуть по строке свойства **Маска ввода**, при этом появится кнопка с тремя точками ;

3) после щелчка по кнопке  пользователю будет предложено сохранить таблицу, после чего откроется диалоговое окно **Создание масок ввода**;

4) выбрать тип маски ввода, например, *краткий формат даты* для поля ДатаПрохождения, и щелкнуть по кнопке **Готово**. Окно **Создание масок ввода** закроется, и маска ввода будет внесена в свойства поля.

Маску ввода можно задать и вручную, введя в строку **Маска ввода** шаблон ввода, например, для даты это 00.00.0000.

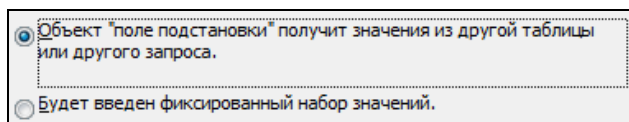
Хранение данных в базе вместе с маской ввода увеличивает размер базы данных, но позволяет облегчить ввод данных.

### 3.6 Задание свойства Подстановка

Поля в таблице могут содержать ограниченное число значений. Поэтому эти значения лучше не вводить, а выбирать из списка.



Чтобы создать для поля список значений, необходимо:

- 1) открыть таблицу в режиме Конструктора;
- 2) щелкнуть мышью в столбце **Тип данных** соответствующего поля;
- 3) выбрать в раскрывающемся списке значение *Мастер подстановок*, при этом будет запущен Мастер;
- 4) в диалоговом окне **Создание подстановки** выбрать опцию



и нажать **Далее**;

5) на следующем шаге Мастера, если в п. 4 установлена опция *будет введен фиксированный набор значений*, надо указать число столбцов (например 1), ввести значения данного поля и завершить формирование списка нажатием кнопки **Готово**. Если в п. 4 установлена опция *объект "поле подстановки" будет использовать значения из таблицы или запроса*, то надо выбрать имя таблицы или запроса, из которого будет выполнена подстановка, и нажать **Далее**;

6) выбрать с помощью кнопок  или  поля, которые будут отображаться в таблице подстановки, нажать **Далее**;

7) при необходимости выполнить сортировку, **Далее**;

8) если для подстановки в п. 6 было выбрано более одного поля, то опцию «Скрыть ключевой столбец» можно не отменять, в этом случае вместо цифры для выбора будет открыт список значений из других выбранных полей. Нажать кнопку **Готово**.



В области **Свойства поля** на вкладке **Подстановка** для выбранного поля появятся новые свойства.

После задания подстановки при заполнении таблицы появится возможность не вводить данные вручную, а выбрать из списка.

### 3.7 Межтабличные связи

*MS Access* позволяет строить реляционные базы данных, отдельные таблицы которых могут быть связаны между собой. Связь между таблицами определяет тип отношения между их полями. Как правило, связывают ключевое поле одной таблицы с соответствующим ему полем другой таблицы, которое называется *полем внешнего ключа*.


При наличии связей между таблицами *MS Access* автоматически будет выбирать связанные данные из таблиц в отчетах, запросах, формах.

Существуют три типа связей:

- один-к-одному – каждая запись таблицы А может быть связана только с одной записью таблицы Б; в этом случае ключевое поле таблицы А связывается с ключевым полем таблицы Б;
- один-ко-многим – одна запись в таблице А может быть связана со многими записями таблицы Б; в этом случае ключевое поле таблицы А связывается с неключевым полем таблицы Б;
- многие-ко-многим – каждая запись в таблице А может быть связана со многими записями в таблице Б, а каждая запись в таблице Б – со многими записями в таблице А.

Нельзя установить связи между полями разного типа: числовое поле можно связать только с числовым полем или полем *Счетчик*, но нельзя связать с текстовым полем.

### 3.8 Создание схемы данных

Чтобы создать связи между таблицами, надо на вкладке **Работа с базами данных** выбрать  **Схема данных**.

*Примечание* – Все изменения в схему данных можно вносить только при закрытых таблицах. Аналогично таблицы можно изменять, если схема данных закрыта.

В отобразившемся на экране окне **Схема данных** можно определить или просмотреть связи между таблицами базы данных.

Если связи между таблицами устанавливаются для базы данных впервые, то окно **Схема данных** окажется пустым и следует начать с добавления таблиц, для которых планируется создать связи. Для этого в окне **Добавление таблицы** выделяют название таблицы и нажимают кнопку **Добавить**. Таким образом, выбирают все таблицы, между которыми надо установить связи.

Для создания связи необходимо поле первичного ключа главной таблицы перетащить с помощью мыши к соответствующему полю связанной таблицы. На экране появится окно **Изменение связей**. В этом окне главная таблица и первичный ключ изображаются в левом столбце, а подчиненная таблица и внешний ключ – в правом столбце. Далее следует установить флажки «Обеспечение целостности данных», «Каскадное обновление связанных полей» и «Каскадное удаление связанных записей» (рисунок 10).

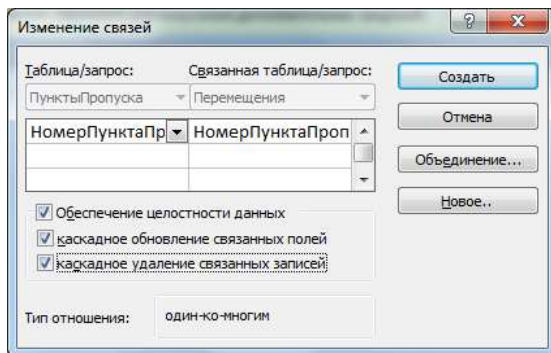


Рисунок 10 – Изменение связей между таблицами

В нижней части окна будет предложен тип связи между таблицами (один-ко-многим). После нажатия на кнопку **Создать** схема данных примет вид, представленный на рисунке 11.

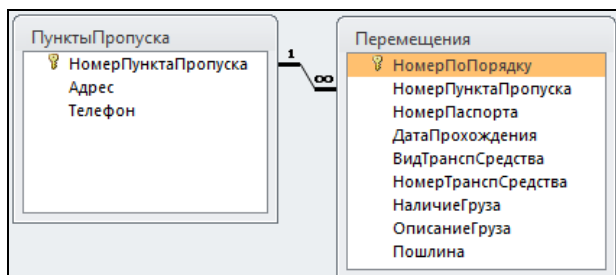


Рисунок 11 – Двухтабличная схема данных

Для удаления связи в схеме данных достаточно щелкнуть по ней правой клавишей мыши и выбрать в отобразившемся контекстном меню команду **Удалить** (рисунок 12).

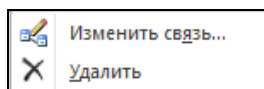


Рисунок 12 – Удаление связи между таблицами

*Примечание* – Как правило, MS Access блокирует многие изменения в таблицах, включенных в межтабличные связи. Поэтому прежде чем изменять структуру таблицы, например, переименовывать или добавлять поля, переназначать ключевое поле и т. п., следует удалить из схемы соответствующую связь, а ещё лучше – и модифицируемую таблицу, после чего выполнить сохранение и закрытие схемы данных.


### 3.9 Задание условий целостности

*Условие целостности* предназначено для того, чтобы обеспечить соответствие записей подчиненной таблицы записям в главной таблице. Оно предотвратит непреднамеренное удаление пользователем тех записей главной таблицы, которые связаны с записями в подчиненной таблице, воспрепятствует добавлению в подчиненную таблицу записей, которым нет соответствия в главной таблице.


Если допускается *каскадное обновление связанных полей*, то при изменении значений первичного ключа в главной таблице происходит автоматическое обновление значений внешнего ключа в подчиненной таблице.


Если разрешено *каскадное удаление связанных записей*, то удаление записи в главной таблице приводит к удалению всех связанных записей в подчиненной таблице.

Для задания условий целостности необходимо:

- на вкладке **Работа с базами данных** выбрать пункт  **Схема данных**. Откроется окно **Схема данных**;
- двойным щелчком мыши на существующих линиях связи открыть диалоговое окно **Изменение связей** и в нем установить требуемые флажки. Щелкнуть по кнопке **ОК**;
- закрыть окно **Схема данных**, сохраняя внесенные изменения.

### 3.10 Работа с таблицами

После сохранения и закрытия таблицы можно найти, выбрав с помощью кнопки  на навигационной панели пункт **Таблицы**.

*Изменение структуры таблицы* выполняется в режиме Конструктора (кнопка  ленты, расположенная на вкладке **Главная**).

Для *ввода данных* таблица открывается двойным щелчком мыши или выбором соответствующей команды из контекстного меню.



Все доступные *команды для работы с открытой таблицей* находятся в контекстном меню, вызванном на вкладке имени таблицы.

После задания всех полей и определения их свойств выполняется ввод данных в таблицу.

*ВНИМАНИЕ!* Данные вводятся сначала в главную таблицу, а затем в подчиненную.

Чтобы заполнить таблицу:

1) выделяют имя таблицы на навигационной панели и дважды щелкают левой клавишей мыши. Можно также в контекстном меню выбрать команду **Открыть**;

2) если в данный момент открыто окно Конструктора таблицы, необходимо перейти в режим отображения, нажав кнопку , расположенную на вкладке **Конструктор**. При этом вид кнопки изменится на , а ее повторное нажатие приведет к возврату в окно Конструктора таблицы;

3) информацию вводят в первое поле, щелкнув в нем мышью. Для перехода к следующему полю нажимают клавишу **Tab** или **Enter**. Затем можно продолжить ввод информации с нажатием клавиши **Tab** после заполнения каждой ячейки, пока не будет достигнут конец записи. Для перехода к следующей записи необходимо нажать **Tab** или **Enter** в конце строки.


Чтобы изменить ширину столбца, указатель мыши наводят на разделительную линию между заголовками столбцов и, когда указатель примет вид двунаправленной стрелки, линию перемещают в требуемом направлении.

### Порядок выполнения работы

1 Запустите *MS Access*: **Пуск / Все программы / Microsoft Office / Microsoft Access 2010**.

*Примечание* – Если базу данных планируется переносить на другой компьютер, то следует сразу изменить параметры файла (см. подразд. 2.1), чтобы базу данных можно было открыть в версиях 2002–2003.

2 Создайте новую базу данных с именем **Таможенный контроль** и сохраните ее в личной папке на диске **z:**.

3 В окне базы данных на вкладке **Создание** выберите команду  – **Конструктор таблиц**.

4 Создайте таблицу **ПунктыПропуска** в соответствии со структурой спроектированной БД (см. рисунок 2). Для этого в окне Конструктора последовательно введите имена полей, выберите тип поля из списка, задайте ключевое поле – **НомерПунктаПропуска**.

Таблица «ПунктыПропуска»			
Поле	Ключевое поле	Тип поля	Свойства поля
<b>НомерПунктаПропуска</b>	<b>Ключ</b>	<b>Числовое</b>	
Адрес		Текстовое	
Телефон		Текстовое	

Закройте Конструктор, сохраните таблицу с именем **ПунктыПропуска**.

5 Создайте в Конструкторе таблицу **Сотрудники** в соответствии со следующей структурой.

Таблица «Сотрудники»			
Поле	Ключевое поле	Тип поля	Свойства поля
<b>ТабельныйНомер</b>	<b>Ключ</b>	<b>Числовое</b>	
НомерПунктаПропуска		Числовое	Мастер подстановок из таблицы «ПунктыПропуска»
Фамилия		Текстовое	
Имя		Текстовое	
Отчество		Текстовое	
Должность		Текстовое	Мастер подстановок (фиксированный набор значений)

6 В таблице **Сотрудники** задайте подстановку поля НомерПунктаПропуска из таблицы **ПунктыПропуска**. Для этого в Конструкторе установите курсор на поле НомерПунктаПропуска и в списке *Тип данных* выберите **Мастер подстановки**. Далее следуйте указаниям Мастера:

- установите переключатель в строке *Объект "столбец подстановки" будет использовать значения из таблицы или запроса*, нажмите **Далее**;
- выберите таблицу **ПунктыПропуска**, значения из которой будут содержать столбец подстановки, **Далее**;
- переместите НомерПунктаПропуска из списка *Доступные поля* в список *Выбранные поля*, **Далее**, **Готово**.

Сохраните таблицу **Сотрудники**.

7 Для поля Должность таблицы **Сотрудники** задайте подстановку, выбрав опцию *Будет введен фиксированный набор значений*, **Далее**. Укажите следующие должности: начальник смены, дежурный караул, начальник поста (рисунок 13).

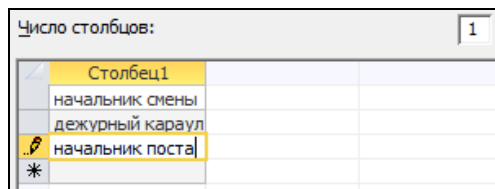


Рисунок 13 – Создание подстановки

8 Создайте в Конструкторе таблицу **Картотека**, опираясь на следующую информационную структуру.

Таблица «Картотека»			
Поле	Ключевое поле	Тип поля	Свойства поля
<b>НомерПаспорта</b>	<b>Ключ</b>	<b>Текстовое</b>	
Фамилия		Текстовое	
Имя		Текстовое	
Отчество		Текстовое	
Адрес		Текстовое	

9 Создайте в Конструкторе таблицу **Перемещения**.

Таблица «Перемещения»			
Поле	Ключевое поле	Тип поля	Свойства поля
<b>НомерПоПорядку</b>	<b>Ключ</b>	<b>Счетчик</b>	
НомерПунктаПропуска		Числовое	Мастер подстановок из таблицы «ПунктыПропуска»
НомерПаспорта		Текстовое	Мастер подстановок из таблицы «Картотека»
ДатаПрохождения		Дата/Время	Маска ввода
ВидТранспСредства		Текстовое	Мастер подстановок (фиксированный набор: машина, мотоцикл и т. д.)
НомерТранспСредства		Числовое	
НаличиеГруза		Логическое	Да/Нет
ОписаниеГруза		Поле MEMO	
Пошлина		Денежное	

10 Для полей таблицы **Перемещение** создайте подстановки из соответствующих таблиц.

11 Для поля ДатаПрохождения задайте *свойство поля* – **Маска ввода**.

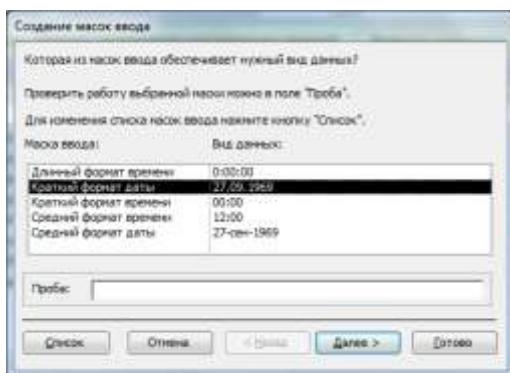


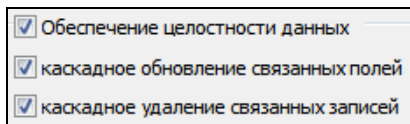


Рисунок 14 – Задание свойства Маска ввода

Для этого воспользуйтесь кнопкой  в конце строки **Маска ввода** общих свойств поля ДатаПрохождения. Выберите *Краткий формат даты* и нажмите **Готово** (рисунок 14).

12 Установите связи между таблицами. Для этого на вкладке **Работа с базами данных** выберите  – **Схема данных**. Так как при создании таблиц были заданы подстановки, то связи уже установлены, необходимо их изменить, установив опции:



13 Измените все связи между таблицами в схеме данных и сохраните внесенные изменения. Итоговая схема данных представлена на рисунке 15.

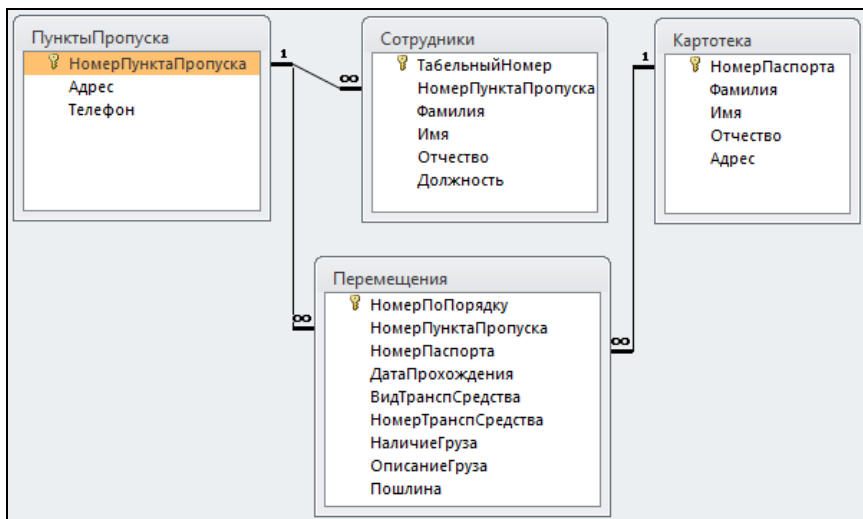


Рисунок 15 – Схема данных «Таможенный контроль»

### Контрольные вопросы

- 1 Проектирование базы данных.
- 2 Нормализация данных.
- 3 Создание базы данных. Настройка параметров.
- 4 Создание таблицы. Режим Конструктора.
- 5 Описание полей базы данных.
- 6 Типы данных полей таблицы.

- 7 Свойства полей.
- 8 Назначение и задание маски ввода.
- 9 Выбор ключевого поля.
- 10 Назначение и использование Мастера подстановки.
- 11 Непосредственный ввод данных в таблицу.
- 12 Редактирование данных в таблице.
- 13 Удаление записи из таблицы.
- 14 Взаимосвязи между таблицами. Создание схемы данных. Добавление и удаление таблиц.
- 15 Типы связей между таблицами.
- 16 Добавление и изменение связей между таблицами. Обеспечение целостности данных.
- 17 Каскадное удаление записей и обновление полей таблиц.



## 4 РАЗРАБОТКА ФОРМ. ВВОД ИНФОРМАЦИИ В БАЗУ ДАННЫХ

### 4.1 Создание форм

*Форма* – объект базы данных, предназначенный для ввода данных в таблицу, *кнопочная форма* – для открытия других объектов базы данных.

Перед созданием формы необходимо создать структуру таблицы или запроса, для которых будет создаваться форма.

Формы обеспечивают пользователям возможность ввода данных без необходимости знать, как устроена таблица. Одна форма может содержать данные из многих таблиц, благодаря чему обеспечивается возможность ввода данных в различные таблицы из одного места. Макет формы можно сконструировать таким образом, что форма на экране будет выглядеть точно так же, как и первоисточник: заявление о приеме, квитанция, заполняемая клиентами, и прочие документы, используемые для сбора данных, подлежащих вводу в базу данных.

Существует несколько способов создания форм. Доступ к ним осуществляется с помощью области **Формы** вкладки **Создание** ленты (рисунок 16).

*Примечание* – Обращайте внимание на всплывающие подсказки, возникающие при наведении на ту или иную кнопку ленты. Помимо названия они могут содержать краткое описание работы инструмента.

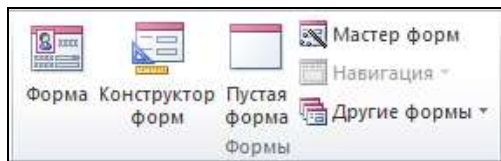


Рисунок 16 – Инструментарий для создания форм в MS Access

Рассмотрим *способы создания форм*.

**Форма.** Позволяет автоматически создать форму, которая выводит в окне данные по одной записи. При создании такой формы таблица или запрос, для которой создается форма, должна быть открыта или выделена в навигационной панели. В нижней части формы расположены элементы управления для перемещения по записям и добавления новых записей, окно для ввода крите-

риев поиска записей и информация о включенном/выключенном фильтре. Если в момент создания формы в таблице был включен фильтр, то его действие сохраняется в форме. Поля записей в данном виде формы располагаются в столбце.

**Конструктор форм.** Поля и другие элементы управления размещаются на форме вручную.

**Пустая форма.** Создает макет формы, не связанной ни с какими таблицами. Отличается от режима Конструктора только тем, что первоначально открывается в режиме Формы, а не Конструктора.

**Мастер форм.** Наиболее общий пошаговый способ создания форм на основе существующей таблицы или запроса.

**Навигация.** Создание управляющих кнопочных форм. Их назначение – вызывать другие формы.

**Другие формы.** Здесь можно создать веб-форму (*Несколько элементов*), сводные таблицу и диаграмму, разделенную форму, которая совмещает в одном окне форму и таблицу. С помощью пункта **Таблица** можно создать форму, имеющую такой же вид, как и таблица в режиме Таблицы.

#### 4.1.1 Создание простой формы в режиме Конструктора

Для создания новой формы с помощью Конструктора необходимо:

1) выбрать **Конструктор форм** в области **Формы** вкладки **Создание** ленты. На экране появится окно, содержащее макет будущей формы (рисунок 17);

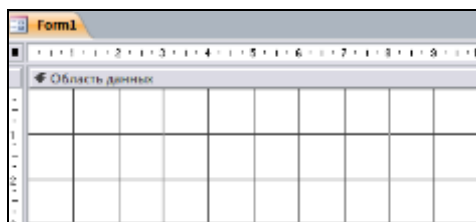



Рисунок 17 – Макет пустой формы

2) установить связь с источником данных, т. е. с таблицей, для которой создается форма, например, с таблицей **ПунктыПропуска**. Для этого в области

**Сервис** вкладки **Конструктор** выбирают кнопку , которая открывает страницу изменения свойств объекта.

Далее на вкладке **Данные** окна свойств, в строке **Источник записей** из списка имеющихся объектов выбирают таблицу, например **ПунктыПропуска** (рисунок 18);

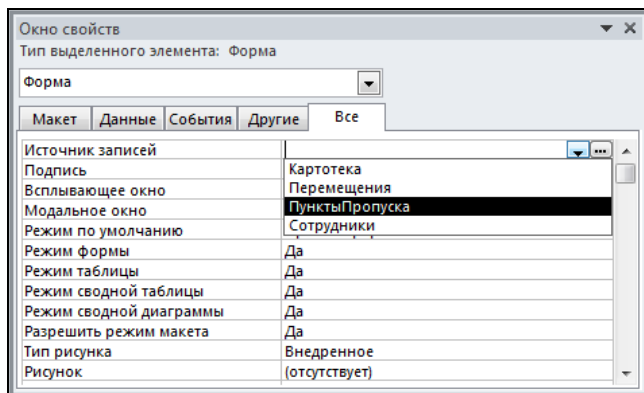



Рисунок 18 – Выбор источника записей для формы

3) вызвать список полей таблицы **ПунктыПропуска** (рисунок 19) с помощью кнопки , расположенной в области **Сервис**, и, перетаскивая поля из списка с помощью мыши, расположить их на макете формы.

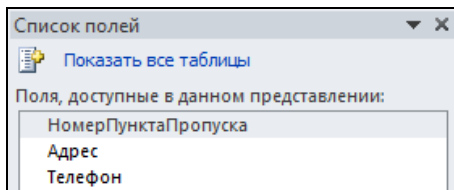
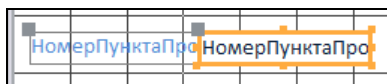


Рисунок 19 – Выбор полей таблицы **ПунктыПропуска**

На макете формы отображаются сами поля и надписи к ним, например:



Выделить поле и надпись можно щелчком мыши. Для перемещения поля или надписи по отдельности, курсор мыши наводят непосредственно на черный квадрат в верхнем левом углу требуемого объекта, при этом вид курсора видоизменяется, и только после этого выполняют перемещение;

4) закрыть форму, сохранив ее.

*Примечание* – Если поле и его надпись необходимо разместить в разных разделах формы, то выделяют либо поле, либо надпись, вырезают, а затем вставляют в другой раздел (см. п. 4.2.1).

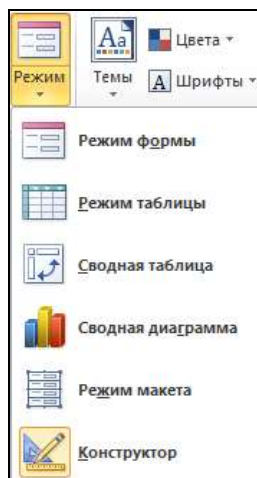


Рисунок 20 – Доступные режимы работы с формой

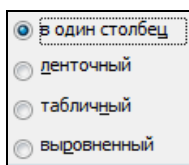





Рисунок 21 – Возможный внешний вид формы

В процессе проектирования формы в режиме Конструктора в любой момент можно перейти в режим Формы, чтобы оценить результат работы. Для этого достаточно в списке **Режим**, расположенном слева на вкладке **Конструктор** ленты, выбрать **Режим формы**. При этом лента перейдет на вкладку **Главная**. Всевозможные режимы для работы с формой показаны на рисунке 20.

#### 4.1.2 Мастер форм

Для создания простых и настраиваемых форм в области **Формы** выбирают  **Мастер форм**. Использование Мастера позволяет включать в форму поля из различных таблиц.

Для создания формы с помощью Мастера:

1) в появившемся окне **Создание форм** в списке **Таблицы/запросы** выбирают таблицу, из которой в форму будут включены поля, а в списке **Доступные поля** – поля для создаваемой формы. С помощью кнопок  или  поля переносят в список *Выбранные поля*. В списке **Таблицы/запросы** последовательно можно выбирать таблицы, из которых необходимо включать поля в форму. При включении в форму полей из различных таблиц будет создана много-табличная форма. После завершения отбора нажимают **Далее**;


2) выбирают *внешний вид формы*. Для этого активизируют один из переключателей, представленных на рисунке 21, **Далее**;

3) вводят *имя формы*, выбирают одну из опций: «открыть форму для просмотра и ввода данных» или «изменить макет формы», **Готово**.

#### 4.1.3 Создание подчиненной формы

*Подчиненная форма* – это вложенная форма, позволяющая отображать данные из связанной таблицы в тех случаях, когда одной записи из главной таблицы может соответствовать несколько записей подчиненной таблицы (связь *один-ко-многим*). Подчиненная форма внедряется в главную связь. Если главная форма уже достаточно сложна, то можно выбрать отображение подчиненной формы не как вложенной, а как связанной. В этом случае на главной форме автоматически создается кнопка, щелчком по которой открывается подчиненная форма.

Чтобы для таблиц, связанных *один-ко-многим*, создать форму:

- 1) в области **Формы** выбирают  **Мастер форм** ;
- 2) в списке **Таблицы/запросы** выбирают главную таблицу и копируют требуемые поля в список *Выбранные поля*;
- 3) в списке **Таблицы/запросы** выбирают подчиненную таблицу и копируют требуемые поля в список *Выбранные поля*, нажимают **Далее**;
- 4) указывают вид представления данных: хотите ли вы, чтобы форма основывалась на главной или на связанной форме, при этом в качестве исходной можно выбрать главную или подчиненную таблицу, **Далее**.


*Примечание* – Опции **Подчиненные формы** и **Связанные формы** определяют отношения между таблицами. Выбрав опцию **Подчиненные формы**, создают главную форму, содержащую подчиненную форму. Если выбрать параметр **Связанные формы**, то в главной форме будет содержаться кнопка, щелчком на которой открывается подчиненная форма;

- 5) выбирают *внешний вид* подчиненной формы, **Далее**;
- 6) вводят *заголовок главной формы*. Для подчиненной формы следует оставить имя, предложенное программой;
- 7) устанавливают одну из опций: «открыть форму для просмотра или ввода данных» или «изменить макет формы». При выборе второй опции после нажатия **Готово** на экране отобразится проект формы.

## 4.2 Изменение форм

### 4.2.1 Изменение структуры формы в режиме Конструктора

Форма в режиме Конструктора имеет три раздела и несколько различных элементов управления (рисунок 22):

- **Заголовок формы** – находится в начале первой страницы формы и используется для размещения заголовка формы. Если заголовок и примечание отсутствуют, то их можно добавить из области **Колонтитулы** щелчком по кнопке  **Заголовок** ;

- **Примечание формы** – располагается в конце последней страницы формы и используется для размещения советов, адресованных пользователю, и другой полезной информации;

- **Область данных** – основная область, в которой располагаются записи данных.

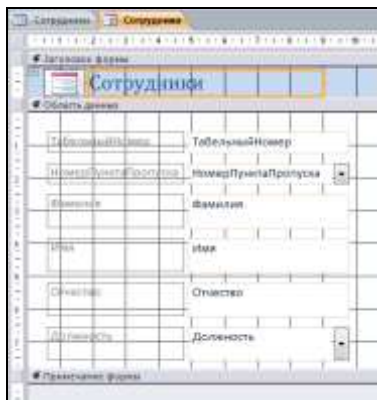


Рисунок 22 – Конструктор формы

Для *изменения размера области данных* надо переместить указатель мыши в конец области данных, туда, где начинается раздел примечаний. Когда указатель примет вид двунаправленной стрелки, разделительную полосу раздела «Примечание формы» перетаскивают вниз, увеличивая тем самым размер раздела «Область данных». Аналогично изменяется размер раздела заголовка.

#### 4.2.2 Элементы управления

*Элементами управления* в формах называются все объекты, с которыми можно выполнять какие-то действия, в том числе и текстовые поля.

Для *выделения объекта* достаточно щелкнуть на нем мышью – вокруг объекта появятся маркеры выделения.

Для *выделения нескольких элементов управления* необходимо установить указатель над левым верхним объектом и, нажав клавишу мыши, протянуть по всем объектам выделения. После того как кнопка мыши будет отпущена, все элементы управления, через которые прошла линия, будут выделены.

Если элементы управления, которые необходимо выделить, не располагаются рядом, можно выделить один элемент, а затем выделять другие, удерживая клавишу **Shift**.

*Удаление* выделенных элементов управления производится клавишей **Delete**.

Элементы управления могут быть:

- **присоединенными**, т. е. связанными с полем базовой таблицы или запроса. Такие элементы управления используются для отображения, ввода или обновления значений из полей базы данных;

- **свободными** – предназначены для вывода на экран линий, прямоугольников, рисунков или независимых данных.

Элементы управления, которыми может пользоваться разработчик, представлены на **Панели элементов** (рисунок 23), а их назначение – в таблице 2.

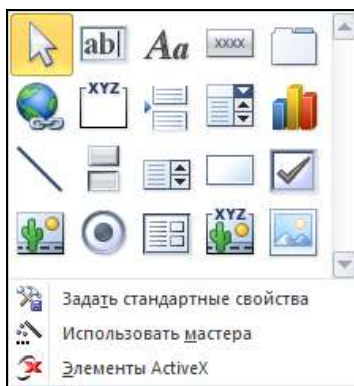























Рисунок 23 – Панель элементов


Таблица 2 – Назначение кнопок панели элементов

Кнопка	Назначение	Кнопка	Назначение	Кнопка	Назначение
	Выбрать		Группа переключателей		Прямоугольник
	Поле		Разрыв страницы		Флажок
	Надпись		Поле со списком		Свободная рамка объекта
	Кнопка		Диаграмма		Переключатель
	Вкладка		Линия		Подчиненная форма
	Гиперссылка		Выключатель		Присоединенная рамка объекта
	Набор вкладок		Список		Рисунок


Выбор элемента управления выполняется одним щелчком на его значке в Панели элементов, после чего следующим щелчком на поле формы отмечается место, куда он должен быть поставлен. Вместе с элементом в поле формы вставляется его присоединенная надпись. По умолчанию эта надпись стандартная, например, для переключателей – это *Переключатель1*, *Переключатель2* и т. д. Редактированием свойства элемента управления (доступ к свойствам открывается через контекстное меню) можно дать элементу управления более содержательную подпись.

Изменение размеров элементов управления выполняется так же, как это принято для графических объектов – перетаскиванием маркеров изменения размера, которые расположены по углам и сторонам выделенных элементов управления.

#### 4.2.3 Создание заголовка формы

Для создания заголовков и других свободных надписей в форме служит элемент управления **Надпись** – .


Для создания заголовка необходимо:

- 1) щелкнуть на панели элементов по кнопке **Надпись** – .
- 2) переместить указатель мыши в область заголовка, он примет вид прописной буквы «А» с перекрестием;
- 3) щелкнуть на том месте формы, где необходимо начать ввод текста;
- 4) после ввода текста щелкнуть на любом месте формы, чтобы закрыть элемент управления;
- 5) выделить созданный элемент и использовать панель форматирования для изменения надписи.

#### 4.2.4 Создание вкладок на форме

Для создания вкладки на форме следует:



1) перейти на вкладку **Создание** ленты;

2) в области **Формы** выбрать способ создания  – **Конструктор форм**;

3) щелкнуть на панели элементов по кнопке  – **Вкладка**;

4) переместить указатель мыши в область данных, он примет вид тонкого крестика со значком вкладки;

5) щелкнуть в том месте области данных, где планируется начать создание формы с вкладками и, не отпуская левую клавишу мыши, растянуть область формы;

6) появится пустая форма с вкладками. На нее следует добавить поля, используя кнопку  – **Добавить поля**. При этом отобразится список полей (рисунок 24), в котором с помощью кнопки  надо открыть требуемую таблицу и перетащить поля на вкладки;

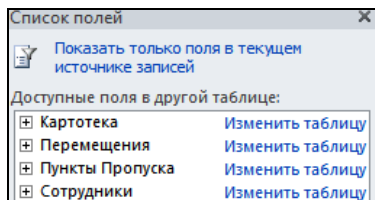


Рисунок 24 – Доступные таблицы для добавления полей

7) созданную по умолчанию надпись Вкладка1 необходимо изменить по смыслу, выбрав в контекстном меню вкладки команду **Свойства / Макет / Подпись** (рисунок 25);

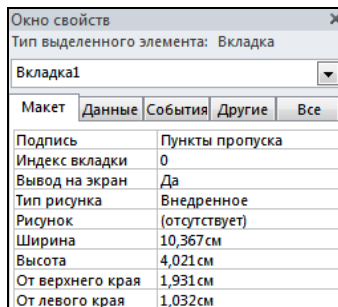


Рисунок 25 – Изменение подписи вкладки

8) сохранить форму.



## Порядок выполнения работы

1 Создайте одноименные формы для всех таблиц, используя способ



**Форма** с вкладки **Создание** ленты.

2 Используя созданные формы, введите в таблицы:

– **ПунктыПропуска** 4 записи;

– **Сотрудники** по 5 записей для каждого пункта пропуска;


– **Картотека** 20 записей;


– **Перемещение** 30 записей, включая лиц из **Картотеки**, т. к. задан мастер подстановки.

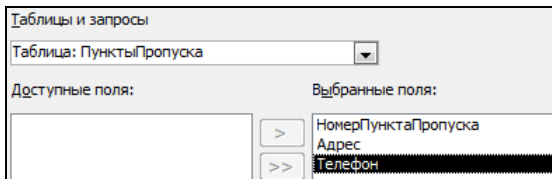
Пункты пропуска ввести реально существующие. Остальные данные выбрать произвольно.

3 Для ввода данных в таблицы, связанные *один-ко-многим*, используют сложные формы, позволяющие вводить данные одновременно в две связанные таблицы, например, **ПунктыПропуска** и **Сотрудники**. Оптимальным способом создания сложных форм является **Мастер форм**.


Создайте сложную форму. Для этого:

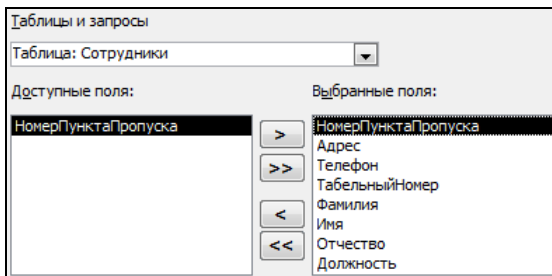
– вызовите  **Мастер форм** с вкладки **Создание** ленты;

– в списке **Таблицы и запросы** выберите таблицу **ПунктыПропуска** и с помощью кнопки  переместите все поля из области *Доступные поля* в область *Выбранные поля*;



Таблицы и запросы	
Таблица: ПунктыПропуска	
Доступные поля:	Выбранные поля:
	НомерПунктаПропуска
	Адрес
	Телефон

– в списке **Таблицы и запросы** выберите таблицу **Сотрудники** и с помощью кнопки  переместите в область *Выбранные поля* все поля, кроме **НомерПунктаПропуска**, **Далее**;



Таблицы и запросы	
Таблица: Сотрудники	
Доступные поля:	Выбранные поля:
НомерПунктаПропуска	НомерПунктаПропуска
	Адрес
	Телефон
	ТабельныйНомер
	Фамилия
	Имя
	Отчество
	Должность



## Контрольные вопросы

- 1 Для чего предназначены формы?
- 2 Инструментарий для создания форм.
- 3 Использование инструмента Форма.
- 4 Создание формы с помощью Мастера форм.
- 5 Создание формы в режиме Конструктора.
- 6 Разделы формы в режиме Конструктора.
- 7 Создание многотабличной (сложной) формы с помощью Мастера форм.
- 8 Подчиненные и связанные формы.
- 9 Состав панели элементов.
- 10 Классификация элементов управления.
- 11 Создание заголовка формы.
- 12 Создание формы с вкладками.
- 13 Создание элемента управления «поле со списком».
- 14 Изменение структуры формы.
- 15 Перемещение элементов управления.

## 5 СОЗДАНИЕ ЗАПРОСОВ К БАЗЕ ДАННЫХ

### 5.1 Общие сведения о запросах

*Запросы* используются для выбора информации из базы данных по критериям, заданным пользователем, с последующим просмотром, изменением и анализом данных различными способами.

Запросы также можно использовать в качестве источников записей для форм, отчетов. В *MS Access 2010* запросы можно создать с помощью Конструктора и с помощью Мастера. На вкладке **Создание** ленты расположены соответствующие кнопки (рисунок 27).

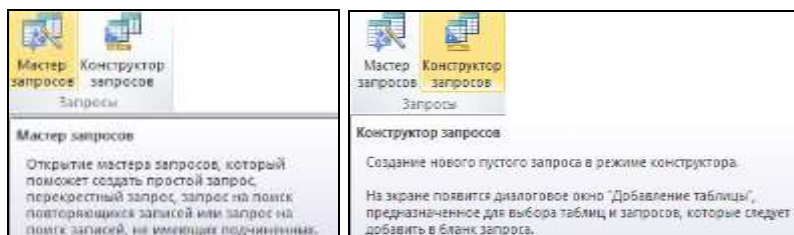


Рисунок 27 – Кнопки создания запросов

В *MS Access* можно создавать несколько видов запросов.

**1 Запросы на выборку** возвращают данные из одной или нескольких таблиц и отображают их в виде таблицы, записи в которой можно обновлять (с некоторыми ограничениями). Запросы на выборку можно также использовать для группировки записей и вычисления сумм, средних значений, подсчета записей и нахождения других типов итоговых значений.

**2 Запросы на совпадение** (запросы с *условием*) позволяют выбрать записи из одной или нескольких взаимосвязанных таблиц, соответствующие некоторому критерию, который указывается в качестве условия отбора записей на этапе разработки запроса.

**3 Запросы с параметрами** – это запросы, выполнение которых начинается с отображения диалогового окна – приглашения ввести данные. Можно, например, разработать запрос, выводящий приглашение на ввод нескольких единиц данных, например, двух дат. В этом случае *MS Access* возвращает все

записи, приходящиеся на интервал времени между этими датами. Запросы с параметрами удобно использовать в качестве основы для форм, отчетов и страниц доступа к данным.

**4 Итоговые запросы** позволяют выполнять простейшее агрегирование данных с последующим получением объединенных совокупных показателей, например, суммарных итоговых значений, средних, наибольших, наименьших значений, количественных характеристик и других показателей.

**5 Перекрестные запросы** используют для расчетов и представления данных в структуре, облегчающей их анализ. Перекрестный запрос подсчитывает сумму, среднее арифметическое, количество значений или выполняет другие статистические расчеты, после чего результаты группируются в виде таблицы по двум наборам данных, один из которых определяет заголовки столбцов, а другой – заголовки строк.

**6 Запросы на изменение** – это запросы, которые за одну операцию изменяют или перемещают несколько записей. Существует четыре типа запросов на изменение:

- на *удаление* записей – удаляет группу записей из одной или нескольких таблиц;
- *обновление* записей – вносит общие изменения в группу записей одной или нескольких таблиц, например, если необходимо увеличить на 5 % зарплату сотрудников определенной категории;
- *добавление* записей – добавляет группу записей из одной или нескольких таблиц в конец одной или нескольких таблиц. Например, если появилось несколько новых сотрудников, а также база данных, содержащая сведения о них;
- *создание* таблицы создает новую таблицу на основе всех или части данных из одной или нескольких таблиц.



**7 Запросы SQL** – это запросы, создаваемые при помощи инструкций структурированного языка запросов SQL.

## 5.2 Создание запросов на выборку

### 5.2.1 Создание запроса с помощью Мастера

Для создания запроса с помощью Мастера:

- 1) на вкладке **Создание** ленты в области **Запросы** нажимают на кнопку **Мастер запросов**;
- 2) в окне **Новый запрос** выбирают **Простой запрос** и нажимают **ОК**;
- 3) в появившемся диалоговом окне выполняют следующие действия:
  - выбирают из списка **Таблицы и Запросы** таблицу, поля которой будут использованы в запросе, например, **Сотрудники**;

- переносят при помощи кнопок  или  поля, например, ТабельныйНомер, НомерПунктаПропуска, Фамилия, Имя, Отчество, Должность из списка *Доступные поля* в область *Выбранные поля*;

- нажимают кнопку **Далее**;

4) в последнем диалоговом окне задают *имя запроса*, например, **Запрос 1**, и нажимают **Готово**.

Если необходимо включить в запрос поля из нескольких таблиц, то сначала выбирают поля из одной таблицы, затем из второй и т. д.

### 5.2.2 Создание запроса на выборку в режиме Конструктора

Для создания запроса:

1) щелкают по кнопке  – **Конструктор запросов**;

2) в окне **Добавление таблицы** на вкладке **Таблицы** устанавливают курсор на таблицу, поля которой будут использованы в запросе, и нажимают кнопку **Добавить**. Таким образом, можно добавить несколько таблиц в запрос. Но эти таблицы должны быть связаны, иначе запрос будет работать неверно. Закрывают окно **Добавление таблицы**;

3) двойным щелчком мыши по полю, которое необходимо включить в запрос, добавляют поля из окна таблицы в бланк запроса (рисунок 28).

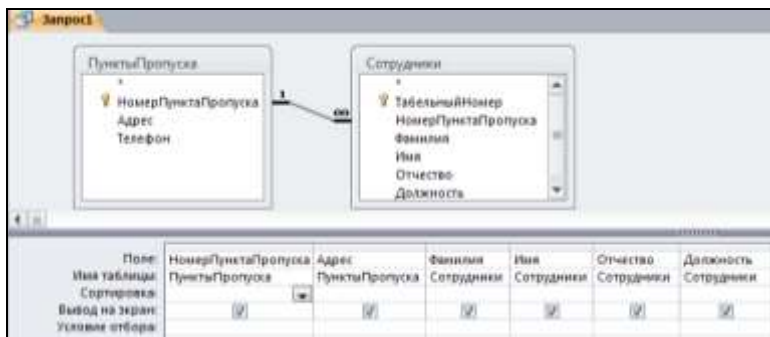





Рисунок 28 – Простой запрос в режиме Конструктора

Далее с помощью кнопки , расположенной на вкладке **Конструктор**

ленты, можно запустить запрос на выполнение. При этом кнопка  на

вкладке **Конструктор** будет заменена кнопкой  вкладки **Главная**. Этими кнопками выполняют переключение из режима Конструктора в режим просмотра записей, внося изменения в структуру запроса и сразу просматривая результат работы запроса.

При попытке закрыть окно Конструктора на экране появится сообщение о том, что можно сохранить запрос, присвоив ему имя.

### 5.2.3 Структура бланка запроса

Запрос в режиме Конструктора состоит из двух основных частей, представленных на рисунке 28:

- списка полей объектов БД, на основе которых создается запрос (в данном случае это поля таблиц **ПунктыПропуска** и **Сотрудники**);
- бланка запроса.

*Бланк запроса* представляет собой таблицу, столбцы которой являются полями запроса, а строки – свойствами полей запроса.

Рассмотрим *свойства полей запроса*:

- **поле** – в данной строке выводится название поля запроса;
- **имя таблицы** – имя таблицы, в которой находится поле запроса;
- **сортировка** – свойство, предназначенное для выбора типа сортировки записей соответствующего поля запроса. Возможные варианты сортировки: по возрастанию, по убыванию, отсутствует;
- **вывод на экран** – регулирует вывод на экран записей соответствующего поля:  – не выводить,  – выводить;
- **условие отбора...или** – условия отбора записей соответствующего поля.

### 5.2.4 Формирование условий отбора в запросах на совпадение

Запрос на совпадение позволяет выбрать информацию по определенному условию, введенному в строке **Условие отбора** для заданного поля. Условия отбора для различных типов данных формируются при помощи логических операций и операций отношения, представленных в таблице 3.

Таблица 3 – Операции для формирования условий отбора

Логические операции		
And	Логическое умножение	Необходимо одновременное выполнение заданных условий
Or	Логическое сложение	Достаточно выполнение одного из заданных условий
Not	Логическое отрицание	Инверсия условия
Like	Совпадение	Совпадение по маске
Операции сравнения		
>	Больше	
<	Меньше	
>=	Больше или равно	
<=	Меньше или равно	
<>	Неравно	

Примеры формирования условий отбора представлены в таблице 4.

Таблица 4 – Формирование условий отбора

Тип данных	Постановка задачи: вывод записей	Пример формирования условия отбора
Текстовый	с фамилией сотрудника Ярошенко	Ярошенко
Дата/время	с датой прохождения границы 09.12.2015	09.12.2015
Числовой	с номером пункта пропуска 1 или 2	1 Or 2
Дата/время	с датой прохождения границы 01.12.2015 или 09.12.2015	#01.12.2015# Or #09.12.2015#
Числовой	с номером пункта пропуска между 2 и 4	Between 2 And 4
Числовой	с табельным номером в интервале (1000; 2000)	>1000 And <2000
Числовой	с табельным номером в интервале [1000; 2000]	>=1000 And <=2000
Числовой	с табельным номером больше 2000	>2000
Дата/время	с датой прохождения границы между 01.04.2016 и 15.04.2016	Between #01.04.2016# And #15.04.2016#
Дата/время	со временем прохождения границы с 6:00 до 10:00	>=#6:00:00# And <=#10:00:00#

Примеры использования масок при формировании условий отбора записей в запросах представлены в таблице 5.

Таблица 5 – Использование масок в условиях отбора

Тип данных	Постановка задачи: вывод записей	Маска
Текстовый	в фамилии сотрудника встречается сочетание «ич»	*ич*
Текстовый	фамилия сотрудника начинается на «В»	В*

### 5.2.5 Параметрический запрос

*Параметрический* запрос (запрос с параметром) – это запрос, который при выполнении требует ввода с клавиатуры интересующего пользователя значения критерия отбора записей.

Пусть, например, требуется выбрать информацию о перемещениях через границу в определенном пропускном пункте. Для этого необходимо создать запрос на выборку данных из таблицы **Перемещения**. Затем для поля **НомерПунктаПропуска**, которое предполагается использовать как параметр, в ячейку строки **Условие отбора** бланка запроса (рисунок 29) ввести текст-приглашение, заключенное в квадратные скобки – **[Введите номер пункта пропуска]**. Текст-приглашение начинается со слова **Введите** и должно отличаться от имени поля, но может включать его.

Квадратные скобки являются для *MS Access* признаком параметра.

При запуске параметрического запроса отображается диалоговое окно, содержащее текст-приглашение, прописанное в бланке запроса. Пользователь



должен в поле ввода указать значение параметра, являющееся критерием отбора записей. Только после этого отобразится результат, соответствующий заданному критерию.

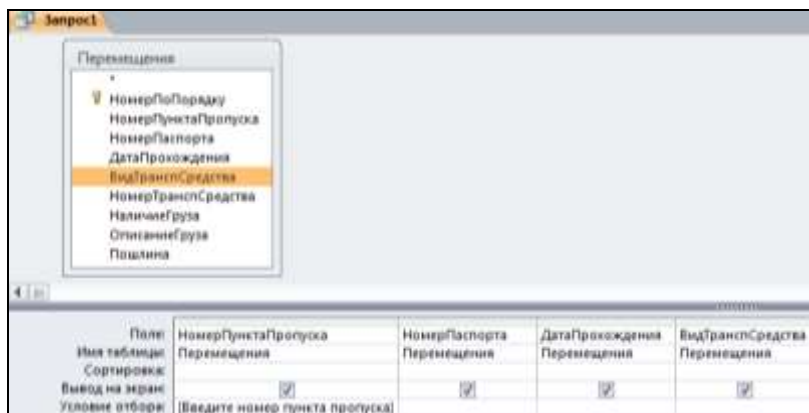


Рисунок 29 – Создание параметрического запроса

Чтобы запросить у пользователя один или несколько символов для поиска записей, которые начинаются с этих символов или содержат их, создается запрос с параметрами, использующий оператор *Like* и подстановочный знак "звездочка" (\*).

Например,

- выражение *Like* [Введите первый символ для поиска: ] & \* выполняет поиск слов, начинающихся с указанного символа;
- выражение *Like* \* & [Введите любой символ для поиска: ] & \* выполняет поиск слов, которые содержат указанный символ.


## 5.3 Вычисления в запросах

### 5.3.1 Создание итоговых запросов в Конструкторе

Итоговые запросы позволяют производить простейшие вычисления сумм, средних, наибольших и наименьших значений и др. на основе запросов на выборку. Например, можно подсчитать количество лиц, которые пересекли границу по определенным датам. Для этого необходимо сгруппировать данные по полю ДатаПрохождения и найти количество перемещений, приходящихся на каждую дату.

Для создания этого запроса можно воспользоваться Конструктором:

- 1) на вкладке **Создание** в области **Запросы** щелкнуть по кнопке **Конструктор запросов**;

- 2) в окне **Добавление таблицы** выбрать таблицу **Перемещения** и добавить в бланк запроса;
- 3) поместить поля **ДатаПрохождения** и **НомерПаспорта** в бланк запроса;
- 4) щелкнуть по кнопке  панели инструментов, после чего в нижней части таблицы появится строка **Групповая операция**.



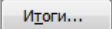
Теперь для каждого столбца таблицы можно выбрать либо группировку, либо одну из следующих операций: **Sum (Сумма)**, **Avg (Среднее)**, **Min (Минимум)**, **Max (Максимум)** **Count (Количество значений)** и т. д.;

5) окончательно запрос для подсчета количества лиц, которые пересекли границу по определенным датам, должен содержать группировку по полю **ДатаПрохождения** и подсчет количества значений (**Count**) по полю **НомерПаспорта**.

### 5.3.2 Создание итоговых запросов с помощью Мастера

Для создания итоговых запросов можно использовать и Мастер запросов.

Предположим, что надо подсчитать суммарную пошлину по каждому пункту пропуска. Для создания такого запроса необходимо:

- 1) на вкладке **Создание** в области **Запросы** выбрать **Мастер запросов**;
- 2) в окне **Новый запрос** выбрать **Простой запрос** и нажать **ОК**;
- 3) в появившемся диалоговом окне выполнить следующие действия:
  - выбрать из списка **Таблицы и Запросы** таблицу **ПунктыПропуска**, поля из которой будут использованы в запросе;
  - перенести при помощи кнопки  поле **НомерПунктаПропуска** из списка **Доступные поля** в список **Выбранные поля**;
  - выбрать из списка **Таблицы и Запросы** таблицу **Перемещения**;
  - перенести при помощи кнопки  поле **Пошлина** из списка **Доступные поля** в список **Выбранные поля**, нажать **Далее**;
- 4) выбрать опцию **Итоговый**, нажать кнопку ;
- 5) для поля **Пошлина** указать функцию **Sum** (рисунок 30), нажать **ОК**, затем кнопку **Далее**;

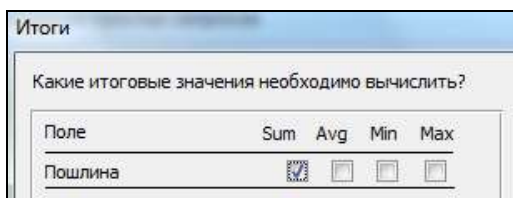
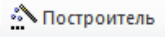


Рисунок 30 – Выбор функции подсчета итогов

- 6) в последнем диалоговом окне задать **имя запроса** и нажать **Готово**.

### 5.3.3 Построитель выражений. Создание вычисляемых полей в запросах

Построитель выражений используется для удобства создания выражений и вычисляемых полей в базе данных.

Вызвать окно построителя выражений можно командой **Построить** контекстного меню или кнопкой  панели инструментов.

Окно **Построитель выражений** (рисунок 31) состоит из двух основных частей. В верхнюю часть (окно ввода формулы) вводится выражение. Нижняя часть окна разбита на три области. В левой области выводится список объектов базы данных, в центральной – список полей объектов базы данных. Справа значения выражений уточняются.

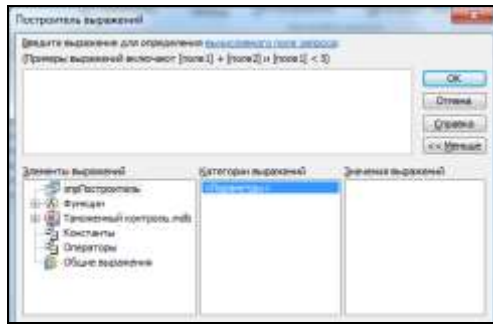


Рисунок 31 – Окно построителя выражений

Пусть требуется создать запрос, переводящий пошлину в евро.

Для этого выполняют следующие действия:

1) создают запрос в Конструкторе, включив в него поля **НомерПунктаПропуска**, **ДатаПрохождения**, **НомерПаспорта**, **Пошлина** из таблицы **Перемещения** и сохраняют его;

2) в бланке запроса устанавливают курсор в строке **Поле** нового столбца и из контекстного меню выбирают команду **Построить**. Появится окно построителя выражений, представленное на рисунке 31;

3) вводят выражение (рисунок 32) и нажимают **ОК**. Выбор полей при этом выполняется двойным щелчком по названию поля в центральной области построителя. Набор основных арифметических и логических операций осуществляется при помощи соответствующих кнопок или клавиатуры;

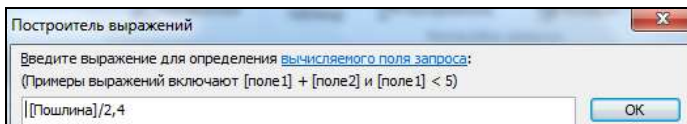


Рисунок 32 – Создание выражения

4) в бланке запроса появится новое поле с именем **Выражение 1**. Эту фразу следует заменить, введя новое имя поля, например, **Пошлина в евро** (рисунок 33);

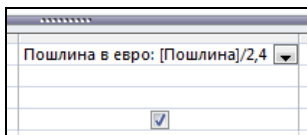


Рисунок 33 – Вычисляемое поле в режиме Конструктора

5) далее можно изменить свойства поля «Пошлина в евро». Для этого из контекстного меню поля выбирают опцию **Свойства**. В окне свойств (рисунок 34) указывают: **Формат поля** – Евро, **Число десятичных знаков** – 1. Сохраняют запрос.

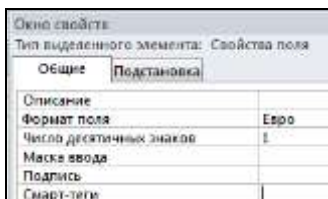


Рисунок 34 – Свойства поля

### 5.3.4 Использование встроенных функций для вычислений

Для создания вычисляемых полей в *MS Access* можно использовать встроенные функции (рисунок 35). При выборе категории в правой части окна построителя выражений открывается список функций соответствующей категории. В таблице 6 приведены некоторые функции.

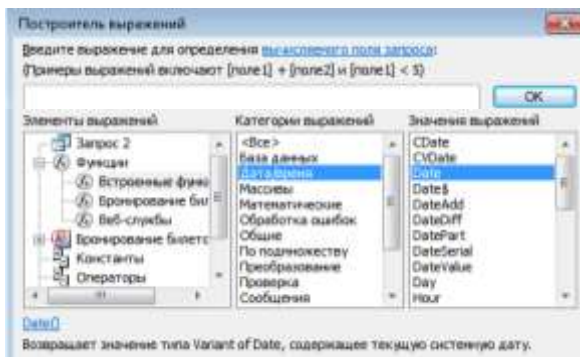


Рисунок 35 – Встроенные функции *MS Access*

Таблица 6 – Некоторые встроенные функции MS Access

Функции Дата/Время	
Date()	возвращает текущую системную дату
Day( <i>date</i> )	из даты выделяет день месяца
Month( <i>date</i> )	из даты выделяет месяц года
Year( <i>date</i> )	из даты выделяет год
DateDiff("d"; <i>date1</i> ; <i>date2</i> )	число интервалов времени между указанными датами ( <i>d</i> – дней; <i>m</i> – месяцев; уууу – лет и т. д.)
Текстовые функции	
Left( <i>stringexpr</i> ; <i>n</i> )	выделяет из строки символов <i>n</i> символов слева
Right( <i>stringexpr</i> ; <i>n</i> )	выделяет из строки <i>n</i> символов справа
Len( <i>stringexpr</i> )	возвращает длину строки <i>stringexpr</i>
Функции управления	
Иф( <i>expr</i> ; <i>truepart</i> ; <i>falsepart</i> )	проверяет условие <i>expr</i> , если оно истинно, то выполняется <i>truepart</i> , иначе – <i>falsepart</i>

Примеры использования встроенных функций представлены в таблице 7.

Таблица 7 – Применение встроенных функций

Пример	Результат
Year(Date())	возвращает 2016
Year([ДатаПрохождения])	возвращает год из даты прохождения
DateDiff("d";[ДатаПрохождения];Date())	возвращает количество дней, прошедших между текущей датой и датой прохождения границы
Left([Фамилия];1)	возвращает первую букву фамилии
Left([Имя];1)	возвращает первую букву имени
Left([Отчество];1)	возвращает первую букву отчества
Иф([НаличиеГруза]=0;[Пошлина];[Пошлина]*2)	если груз отсутствует, пошлина не меняется, иначе увеличивается в 2 раза
Иф([ВидТранспСредства]="мотоцикл";[Перемещение]!/([Пошлина]*0,95);[Перемещение]!/([Пошлина]))	если вид транспортного средства – мотоцикл, то пошлина уменьшается на 5 %, иначе – остается прежней

## 5.4 Перекрестные запросы

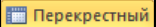
Перекрестный запрос, предназначенный для представления данных в структурированном, удобном для пользователя виде, можно создать двумя способами:

- в режиме Конструктора (вручную);
- с помощью Мастера, выбрав способ создания запроса *Перекрестный*.

### 5.4.1 Создание перекрестного запроса в Конструкторе

Пусть требуется создать запрос, в котором данные по пошлине сгруппированы в таблицу, в строках которой расположены номера пунктов пропуска, а в столбцах – даты прохождения границы.

Для создания такого перекрестного запроса в *режиме Конструктора*:

- 1) выполняют последовательно **Создание / Конструктор запросов**;
- 2) добавляют таблицу или таблицы, из которых в запрос будут выбираться поля (таблица **Перемещения**);
- 3) последовательно перетаскивают поле:
  - значения которого будут заголовками строк перекрестной таблицы (поле **НомерПунктаПропуска**);
  - значения которого будут заголовками столбцов перекрестной таблицы (поле **ДатаПрохождения**);
  - по которому подводится итог с использованием статистических функций (поле **Пошлина**);
- 4) на вкладке **Конструктор** нажимают кнопку  **Перекрестный**, расположенную в группе **Тип запроса**. В результате в бланке запроса появятся две новые строки: **Групповая операция** и **Перекрестная таблица**;
- 5) в строке **Перекрестная таблица** для полей со значениями в роли заголовков строк выбирается из раскрывающегося списка значение **Заголовки строк** (для поля **НомерПунктаПропуска**), а для поля со значениями в роли заголовков столбцов (для поля **ДатаПрохождения**) – **Заголовки столбцов**;
- 6) в строке **Групповая операция** для поля, по которому подводится итог (поле **Пошлина**), из раскрывающегося списка выбирают необходимую статистическую функцию, а в строке **Перекрестная таблица** – **Значение** (рисунок 36);

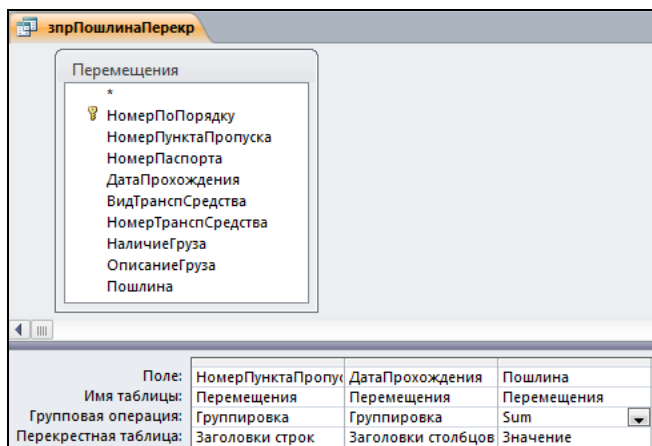


Рисунок 36 – Перекрестный запрос

7) для вывода суммарной пошлины по пропускному пункту за все дни необходимо добавить еще одно поле Пошлина и в строке **Групповая операция** из списка выбрать функцию суммирования (*Sum*), а в строке **Перекрестная таблица – Заголовки строк**;

8) далее в контекстном меню поля изменяют название, выбрав в свойствах **Подпись** и введя новое название итогового поля – *Итого*;

9) запрос сохраняют и запускают на выполнение.

#### 5.4.2 Создание перекрестного запроса с помощью Мастера

Далее для создания перекрестного запроса *с помощью Мастера*:

1) выполняют последовательно **Создание / Мастер запросов / Перекрестный запрос** и следуют указаниям Мастера;

2) выбирают таблицу или запрос, который будет использован в качестве основы для создания перекрестного запроса, в нашем случае это таблица **Перемещение, Далее**.

*Примечание* – Для включения в перекрестный запрос полей из нескольких таблиц сначала создают обычный запрос, содержащий все необходимые поля;

3) выбирают поле, данные из которого будут использоваться в качестве заголовков строк (поле **НомерПунктаПропуска**), **Далее**;

4) выбирают поле, данные из которого будут использоваться в качестве заголовков столбцов (поле **ДатаПрохождения**), **Далее**;

5) выбирают поле, содержащее числа, над которыми будут проводиться итоговые вычисления (поле **Пошлина**), и итоговую функцию для этих вычислений (функция *Sum*). Указывают, требуется ли итоговый результат по каждой строке, **Далее**;

б) вводят имя перекрестного запроса и щелкают по кнопке **Готово**.

*Примечание* – При создании перекрестного запроса поле, по которому выбирается функция, должно быть числовым или денежным.

### 5.5 Запросы на изменение

#### 5.5.1 Разрешения на выполнение запросов манипулирования данными в MS Access 2010

В *MS Access 2010* выполнение всех запросов на удаление, обновление по умолчанию заблокировано, поэтому при попытке их запустить ничего не происходит. Для того чтобы выполнить такой запрос, следует щелкнуть на кнопке **Включить содержимое** на специальной строке безопасности, расположенной ниже ленты (рисунок 37).



Рисунок 37 – Строка безопасности

Включение активного содержимого также означает разрешение запуска макросов и модулей, созданных в приложении.

### 5.5.2 Формирование запроса на удаление записей из таблиц базы данных

При удалении записей из базы данных необходимо **помнить** следующее.

1 Если между таблицами установлена связь с обеспечением целостности данных и с каскадным удалением записей, то создается запрос на удаление записей только из главной таблицы.

2 Если между таблицами установлена связь с обеспечением целостности данных, но без каскадного удаления записей, то сначала создается запрос на удаление записей из подчиненной таблицы, а затем – из главной.

Пусть требуется создать запрос на удаление записей из таблиц с номером пропускного пункта 2. Для этого:

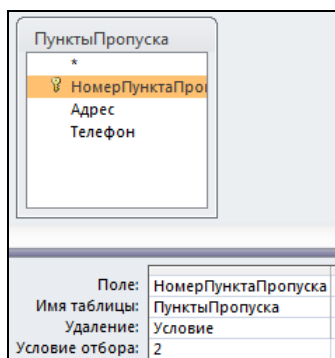


Рисунок 38 – Запрос на удаление записей из главной таблицы

#### Примечания

- 1 Проверьте, удалены ли записи из подчиненной таблицы.
- 2 Обратите внимание на вид значка запроса.


Пусть надо создать запрос на удаление из таблиц записей с номером пункта пропуска 3, если *не задано каскадное удаление записей*. Для этого:

- 1) открывают схему данных и изменяют связь, убрав каскадное удаление;
- 2) удаляют записи из подчиненной таблицы. Для этого:

- выбирают режим Конструктор;
- добавляют главную таблицу **ПунктыПропуска** и подчиненную таблицу

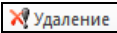
**Сотрудники** (связь *один-ко-многим*);

- 1) проверяют целостность данных и каскадное удаление (см. схему данных);
- 2) удаляют записи из главной таблицы. Для этого:

- выбирают режим Конструктор;
- добавляют главную таблицу **ПунктыПропуска**, помещают поле **НомерПунктаПропуска** в бланк запроса;
- на панели инструментов нажимают кнопку  **Удаление**. Появится строка **Удаление**. В строку **Условие отбора** вводят условие 2 (рисунок 38);
- сохраняют и запускают запрос на выполнение. В результате появится окно, в котором для удаления записей выбирают *Да*, для отмены – *Нет*.



- перетаскивают звездочку (\*) из подчиненной таблицы **Сотрудники** в бланк запроса и поле **НомерПунктаПропуска** из главной таблицы **ПунктыПропуска** в бланк запроса;

- на панели инструментов нажимают кнопку . Появится строка **Удаление**, а ниже – строка **Условие отбора**, в которую вводят условие удаления записей, т. е. 3;

- сохраняют и запускают запрос на выполнение; при этом будут удалены записи из подчиненной таблицы с номером пункта пропуска 3;

- в окне Конструктора удаляют подчиненную таблицу **Сотрудники**;
- удаляют записи из главной таблицы.

Если *подчиненных таблиц несколько*, то следует:

- 1) добавить в бланк запроса вторую подчиненную таблицу, удалить записи из нее, затем удалить подчиненную таблицу в окне Конструктора. И так поступают со всеми подчиненными таблицами;

- 2) удалить записи из главной таблицы.

### 5.5.3 Создание запроса на обновление данных в таблице


Пусть требуется создать запрос, который увеличит значение поля **Пошлина** в таблице **Перемещения** на 5 %, начиная с указанной даты.

Для создания запроса:

- 1) последовательно выполняют **Создание / Конструктор запросов**;

- 2) добавляют таблицу **Перемещения**, из которой будут в запрос выбираться поля;

- 3) помещают поля **ДатаПрохождения** и **Пошлина** в бланк запроса и сохраняют запрос, не закрывая **Конструктор**;


- 4) нажимают кнопку  на панели инструментов вкладки **Конструктор**. При этом в таблице конструктора запросов появится новая строка **Обновление**;


- 5) для поля **ДатаПрохождения** в строке **Условие отбора** указывают диапазон дат, для которых необходимо обновить пошлину, например, **>#15.05.2016#**;

- 6) для поля **Пошлина** в строке **Обновление** вводят формулу с помощью Построителя выражений: **[Пошлина]\*1,05** или **[Пошлина] + [Пошлина]/100\*5** (рисунок 39). Запрос сохраняют;

Поле:	<b>Пошлина</b>	ДатаПрохождения
Имя таблицы:	Перемещения	Перемещения
Обновление:	[Пошлина]*1,05	
Условие отбора:		>#15.05.2016#

Рисунок 39 – Запрос на обновление

7) с помощью кнопки  панели инструментов можно просмотреть все записи, которые подлежат обновлению;

8) с помощью кнопки  панели инструментов запрос запускают на выполнение. На экране появится окно (рисунок 40), в котором следует подтвердить обновление или отказаться от него.

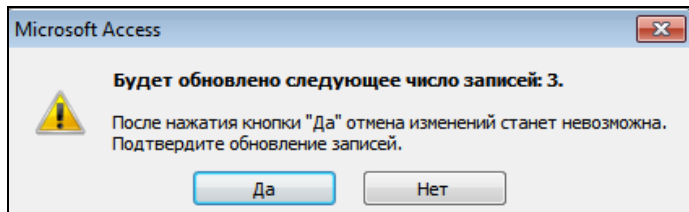




Рисунок 40 – Подтверждение обновления в базе данных

При каждом запуске запроса поле **Пошлина** в таблице **Перемещения** будет обновляться для дат прохождения после 15.05.2016.

### Порядок выполнения работы

1 Создайте запрос на выборку, выводящий данные из таблицы **Сотрудники**. Для этого в окне базы данных перейдите на вкладку **Создание** и в области **Запросы** выберите  – **Мастер запросов**, затем **Простой запрос** и нажмите **ОК**. В появившемся диалоговом окне выполните следующие действия:

– выберите из списка **Таблицы и Запросы** таблицу **Сотрудники**, поля из которой будут использованы в запросе;

– при помощи кнопки  переместите поля **ТабельныйНомер**, **НомерПунктаПропуска**, **Фамилия**, **Имя**, **Отчество**, **Должность** из списка **Доступные поля** в область **Выбранные поля**, **Далее**;

– в последнем диалоговом окне дайте имя запросу **зпрСотрудники** и нажмите кнопку **Готово**.

2 Создайте с помощью Конструктора запрос на выборку, выводящий данные из таблицы **ПунктыПропуска**. Для этого выполните следующие действия:

– на вкладке **Создание** выберите команду  – **Конструктор запросов**;

– в появившемся диалоговом окне **Добавление таблицы** на вкладке **Таблицы** установите курсор на таблицу **ПунктыПропуска**, нажмите последовательно **Добавить**, **Заккрыть**;

– двойным щелчком мыши по каждому полю добавьте поля из окна таблицы в бланк запроса (рисунок 41);

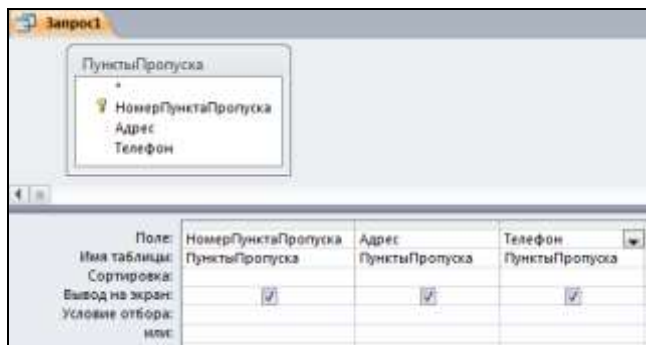


Рисунок 41 – Запрос в режиме Конструктора


– закройте запрос, сохранив его с именем **зпрПунктыПропуска**.


3 Создайте запрос на выборку, выводящий данные полей НомерПунктаПропуска, Адрес, Телефон из таблицы **ПунктыПропуска** и полей Фамилия, Имя, Отчество, Должность из таблицы **Сотрудники**. Отсортируйте записи по полю НомерПунктаПропуска. Имя запроса **зпрСортировка**.

Для создания запроса вызовите Конструктор запросов и выполните следующие действия:

– в окне **Добавление таблицы** перейдите на вкладку, содержащую объекты, данные из которых будут использованы в запросе. Двойным щелчком мыши выберите таблицы **ПунктыПропуска** и **Сотрудники**, которые будут добавлены в окно запроса, и нажмите кнопку **Заккрыть**. Обратите внимание, таблицы **Сотрудники** и **ПунктыПропуска** связаны по полю НомерПунктаПропуска;

– в бланке запроса двойным щелчком мыши выберите поля, которые будут использоваться в запросе;

– просмотрите результат выполнения запроса, нажав кнопку  панели инструментов;

– вернитесь в Конструктор запросов, используя кнопку переключения режима ;

– чтобы отсортировать записи по полю НомерПунктаПропуска, выберите для указанного поля из раскрывающегося списка в строке **Сортировка** пункт по возрастанию;

– просмотрите внесенные изменения в режиме Таблицы;

– закройте запрос, сохраняя его с именем **зпрСортировка**.

4 Создайте запрос на совпадение, который выберет информацию о сотрудниках только из пункта пропуска с номером 2.

В качестве основы для создания запроса возьмите запрос **зпрСотрудники**. Откройте его и выполните команду **Файл / Сохранить объект как**. Введите новое имя запроса **зпрСотрудники2** (рисунок 42) и нажмите **ОК**.

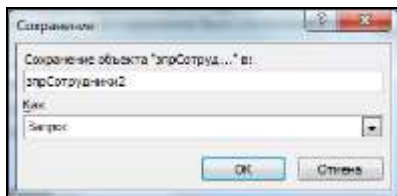



Рисунок 42 – Сохранение объекта с другим именем

Далее выполните следующие действия:

- откройте запрос **зпрСотрудники2** в режиме Конструктора;
- в бланке запроса в строке **Условие отбора** для поля **НомерПунктаПропуска** выведите 2;

- просмотрите результат выполнения запроса, нажав одну из кнопок



или , расположенных на панели инструментов;

- закройте запрос, выполняя его сохранение.

5 Создайте параметрический запрос для вывода информации об определенном пункте пропуска. Для этого выполните действия:

- сохраните запрос **зпрПунктыПропуска** с именем **зпрПунктыПропускаПарам** и откройте его в режиме Конструктора;

- для поля **НомерПунктаПропуска** введите в ячейку строки **Условие отбора** бланка запроса (рисунок 43) текст-приглашение, заключенное в квадратные скобки, [**Введите номер пункта пропуска**];

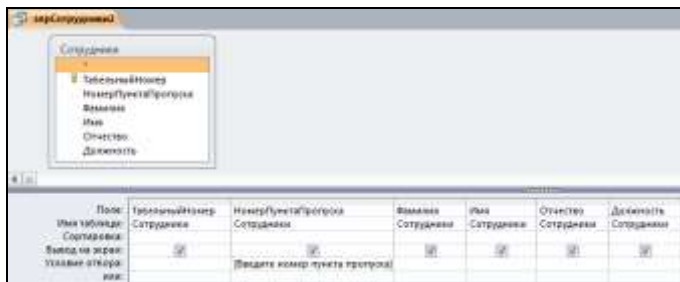


Рисунок 43 – Параметрический запрос в Конструкторе

- сохраните запрос и запустите его на выполнение;

– в отобразившееся диалоговое окно (рисунок 44) введите значение параметра, например, номер пункта пропуска 1, и нажмите **ОК**;

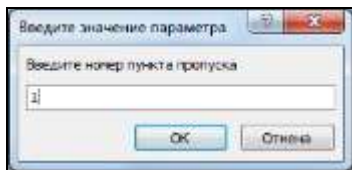


Рисунок 44 – Ввод критерия отбора записей


– просмотрите полученный результат.

6 Создайте параметрический запрос с именем **зпрДатаПрохПарам** для вывода информации о перемещениях через границу в определенный диапазон дат. Для этого создайте запрос на выборку на основе таблицы **Перемещения** и в ячейке **Условие отбора** соответствующего поля **ДатаПроезда** наберите

**Between [Введите начальную дату:] And [Введите конечную дату:]**

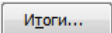
Запустите запрос на выполнение, введите начальную и конечную даты, получите и проанализируйте результат.

7 Создайте итоговый запрос **зпрПошлина**, вычисляющий суммарную пошлину по каждому пункту пропуска. Для этого:

– в окне базы данных выберите вкладку **Создание**, затем  – **Мастер запросов**;

– в появившемся диалоговом окне выберите команду **Простой запрос**.

– выберите из таблицы **Перемещение** поля **НомерПунктаПропуска** и **Пошлина**, нажмите кнопку **Далее**;

– укажите опцию **Итоговый** и нажмите кнопку ;

– для поля **Пошлина** выберите функцию **Sum**, нажмите **ОК**, **Далее**. В итоговых запросах рассчитываются итоги только по числовым полям БД;

– введите имя запроса **зпрПошлина** и нажмите **Готово**.

8 Создайте запрос с именем **зпрСколькоПП**, подсчитывающий, сколько человек прошло через каждый пункт пропуска. Для этого используйте таблицу **Перемещения**, группировку по полю **НомерПунктаПропуска**, функцию **Count** для поля **НомерПаспорта** (рисунок 45).

Поле:	НомерПунктаПропуска	НомерПаспорта
Имя таблицы:	Перемещения	Перемещения
Групповая операция:	Группировка	Count
Сортировка:		
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:		

Рисунок 45 – Итоговый запрос в Конструкторе

9 Создайте запрос с именем **зпрСколькоВсего**, подсчитывающий, сколько всего перемещений было через все пункты пропуска. Для этого следует использовать таблицу **Перемещения**, функцию **Count** для поля **ДатаПрохождения** (рисунок 46).

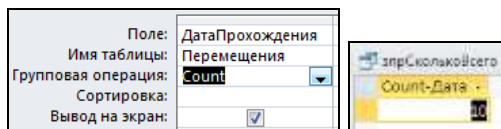


Рисунок 46 – Подсчет количества прохождений границы

10 Посчитайте, сколько человек прошло через все пункты пропуска 04.05.2016. Запрос сохраните с именем **зпрСколькоДата** (рисунок 47).

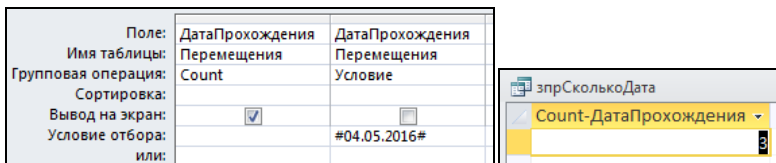
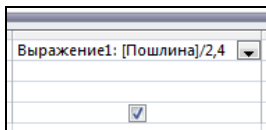


Рисунок 47 – Подсчет количества прохождений за указанную дату

11 Создайте запрос с именем **зпрПошлинаЕвро**, переводящий размер пошлины в евро. Для этого выполните следующие действия:

- вызовите **Конструктор запросов**, расположенный на вкладке **Создание**;
- добавьте таблицу **Перемещения** в бланк запроса;
- добавьте поля **НомерПунктаПропуска** и **Пошлина** в бланк запроса;
- сохраните запрос с именем **зпрПошлинаЕвро**;
- в бланке запроса установите курсор в строке **Поле** нового столбца;
- вызовите контекстное меню и выберите команду **Построить**. Появится окно построителя выражений;
- двойным щелчком мыши подставьте поле **Пошлина** из средней части окна в верхнюю часть окна. Наберите знак деления (/). Набор основных арифметических и логических операций осуществляется при помощи соответствующих кнопок или клавиатуры. Введите курс евро. Нажмите **ОК**;
- в бланке запроса появится новое поле:



– вместо фразы *Выражение1* введите новое имя поля *Пошлина в евро* (см. рисунок 33);

– измените свойства поля «Пошлина в евро». Для этого выберите в контекстном меню поля опцию **Свойства**. В свойствах укажите **Формат поля – Евро**, **Число десятичных знаков – 1** (см. рисунок 34);

– сохраните запрос **зпрПошлинаЕвро**.

*Примечание* – Если вместо числового значения курса в формулу ввести в квадратных скобках [Введите курс евро], то при каждом запуске запроса на выполнение будет запрашиваться курс евро.

12 Создайте в Конструкторе перекрестный запрос с именем **зпрПошлинаПерекрК**, в котором данные по пошлине сгруппированные в таблицу, в строках которой расположены номера пунктов пропуска, а в столбцах – даты прохождения границы (см. п. 5.4.1). Выполните расчеты суммарной пошлины по пропускному пункту за все дни.

13 Создайте с помощью Мастера перекрестный запрос с именем **зпрПошлинаПерекрМ** с условием, аналогичным п. 13 (см. п. 5.4.2).

14 Создайте запрос на удаление записей из таблиц с номером пропускного пункта 2 (см. п. 5.5.2).

15 Создайте запрос, который увеличит значение поля **Пошлина** в таблице **Перемещения** на 5 %, начиная с 15.05.2016 (см. п. 5.5.3).

### **Задания для самостоятельной работы**

1 Создать запрос на выборку, выводящий данные из таблицы **Картотека**, используя Конструктор. Запрос сохранить с именем **зпрКартотека**.

2 Создать запрос на выборку, выводящий данные из таблицы **ПунктыПропуска**, используя Конструктор. Данные отсортировать по возрастанию по полю Адрес. Запрос сохранить с именем **зпрПунктыПропускаАдрес**.

3 Создать запрос на выборку, выводящий данные из таблицы **Сотрудники**, используя Конструктор. Данные отсортировать по убыванию по полю Фамилия. Запрос сохранить с именем **зпрСотрудникиФамилия**.

4 На основе запроса **зпрСотрудникиФамилия** создайте запрос с параметром по полю Должность. Запрос сохранить с именем **зпрСотрудникиДолжность**.

5 Создать запрос на выборку, выводящий все поля из таблицы **Перемещения** с величиной пошлины больше 10 руб. Запрос сохранить с именем **зпрПошлина10**.

6 Создать запрос, увеличивающий пошлину в 2 раза для лиц, проходящих через 2-й пропускной пункт, для этого:

– создать запрос, включив поля **НомерПропускногоПункта** и **Пошлина** из таблицы **Перемещения**;

– сохранить запрос с именем **зпрПошлина2**;

– области **Тип запроса** Конструктора запросов выбрать команду **Обновление**;

– для поля НомерПропускногоПункта в строке **Условие отбора** введите 2, для поля Пошлина постройте выражение обновления (пошлину увеличить в 2 раза) с помощью строителя выражений;

– повторно сохраните запрос.

7 Создать итоговый запрос, вычисляющий максимальную пошлину по каждому пропускному пункту. Запрос сохранить с именем **зпрПошлинаМакс**.

8 Создать итоговый запрос, вычисляющий суммарную пошлину по всем пропускным пунктам. Запрос сохранить с именем **зпрПошлинаСумм**.

9 Создать запрос, выводющий все поля из таблицы **Перемещения** и создать вычисляемое поле, в котором вычислить пошлину в долларах. Запрос сохранить с именем **зпрПошлинаДоллар**.

10 На основе **зпрПошлина10** создать запрос, выводющий все записи с пошлиной больше 10 руб. и меньше 50 руб. Запрос сохранить с именем **зпрПошлина10\_50**.

11 Создать запрос, выводющий все записи из таблицы **Перемещения** с видом транспорта, который начинается на букву А. Запрос сохранить с именем **зпрТранспортА**.

12 Создать запрос, выводющий все записи из таблицы **Перемещения** за период с 01.01.2016 по текущую дату. Запрос сохранить с именем **зпрПеремещенияДата**.

13 Создать запрос, в котором отобразить фамилию сотрудника и его инициалы (Например, Иванов С. Н.). Запрос сохранить с именем **зпрИнициалы**.

14 Вывести список всех лиц, пересекших границу, упорядоченный по фамилии, имени, отчеству, адресу. Запрос сохранить с именем **зпрФамилия**.

15 Вывести список всех сотрудников по пропускным пунктам, фамилия которых начинается на букву С. Запрос сохранить с именем **зпрСотрудникиС**.

16 Вывести список лиц, прошедших таможенные пункты в 2016 году. Запрос сохранить с именем **зпрГод2016**.

17 Вывести список машин, пересекших границу в определенный день, значение которого является параметром запроса. Запрос сохранить с именем **зпрДень**.

18 Рассчитать суммарную пошлину по каждому пропускному пункту. Запрос сохранить с именем **зпрПошлинаСуммПП**.

19 Увеличить пошлину на 10 % для лиц, у которых есть груз. Запрос сохранить с именем **зпрГруз10**.

20 Если пошлина больше 150 руб., ввести дополнительную пошлину, равную 20 % от суммы пошлины. Запрос сохранить с именем **зпрДопПошлина**.

21 Вывести данные о первом человеке, который пересек границу в 2016 году. Запрос сохранить с именем **зпрПервый**.



22 Рассчитать, сколько дней прошло между первым и последним пересечением границы. Запрос сохранить с именем **зпрСколькоДней**.

23 Вывести список лиц, перевозивших строительные материалы. Запрос сохранить с именем **зпрОписаниеГруза**.

24 Посчитать процент лиц, не перевозивших груз. Запрос сохранить с именем **зпрПроцентБезГруза**.

### **Контрольные вопросы**

- 1 Что такое запрос?
- 2 Основные виды запросов.
- 3 Назовите основные режимы создания запросов на выборку.
- 4 Что такое параметрический запрос?
- 5 Как формируются условия отбора в запросах?
- 6 Для чего используется Построитель выражений. Как его вызвать?
- 7 Структура Построителя выражений.
- 8 Встроенные функции *MS Access*.
- 9 Как создать вычисляемое поле в запросе?
- 10 Для чего используются запросы на обновление?
- 11 Для чего используются итоговые запросы?
- 12 Другие виды запросов и их использование.
- 13 Структура бланка запроса.
- 14 Какая команда используется для открытия запроса в режиме редактирования?
- 15 Какая команда используется для просмотра результатов выполнения запроса в окне БД?
- 16 Как сохранить запрос с новым именем?

## 6 ОТЧЕТЫ

### 6.1 Способы создания отчетов

*Отчет* – объект базы данных, позволяющий подготовить материалы базы данных к печати, создав выходные документы в соответствии с требованиями конкретного пользователя.

Существует несколько способов создания отчетов. Доступ к ним осуществляется с помощью области **Отчеты** вкладки **Создание** (рисунок 48).

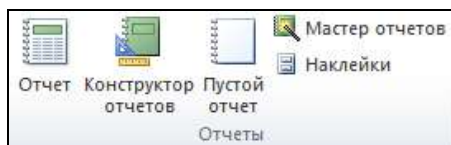


Рисунок 48 – Вкладка для создания новых отчетов

*Примечание* – При наведении курсора мыши на кнопки вкладки возникает всплывающая подсказка, содержащая полезную информацию.

Кнопка **Отчет** позволяет создать простой отчет на базе таблицы, выделенной на навигационной панели.

**Конструктор отчетов** создает пустой отчет в режиме конструирования. Окно конструктора отчета состоит из следующих основных областей, размещение полей в каждой из которых решает определенные задачи:

- заголовок отчета;
- верхний колонтитул;
- нижний колонтитул;
- область данных;
- примечание отчета.

Добавить или удалить заголовок/примечание можно с помощью контекстного меню, в котором выбирают **Колонтитулы страницы** или **Заголовок/примечание отчета**. Заголовок печатается один раз в самом начале отчета, а примечание – в конце отчета. Колонтитулы при печати отображаются на каждой странице.

Для добавления полей необходимо на вкладке **Конструктор** в области **Сервис** выбрать **Добавить поля** (рисунок 49).

Появится окно **Список полей**, содержащее все таблицы базы данных, в случае, если ни одна таблица (запрос) не была выбрана (рисунок 50, а). Если предварительно в окне свойств на вкладке **Данные** была выбрана таблица (запрос), то в верхней части окна списка полей будут отображаться только поля выбранной таблицы, а в нижней – список полей в связанных таблицах (рисунок 50, б).

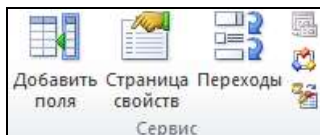
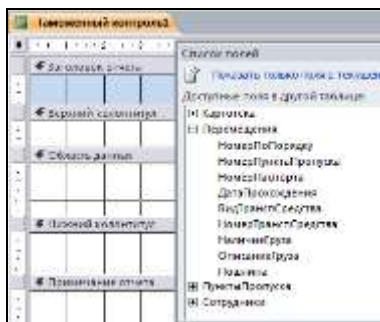
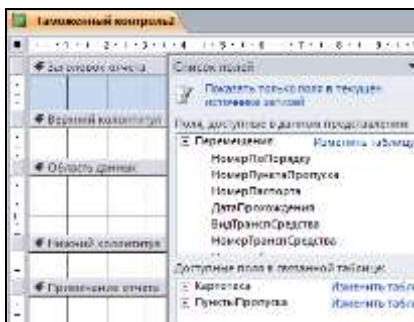


Рисунок 49 – Область Сервис



а)



б)

Рисунок 50 – Конструктор отчета:

а – источник для отчета предварительно не выбран;

б – источник для отчета выбран

Для изменения свойств на вкладке **Конструктор** в области **Сервис** выбирают **Страница свойств** (см. рисунок 49).

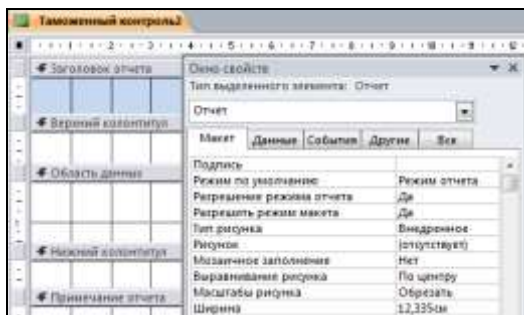



Рисунок 51 – Общий вид окна отчета

В окне свойств можно установить свойства самого отчета, его разделов и элементов управления (рисунок 51). Набор свойств в окне зависит от того, какой объект является активным (т. е. выделен).

Из списка полей в отчет можно добавлять поля нескольких связанных таблиц.

Для работы с отчетами в *MS Access* предусмотрено четыре режима:


- **Представление** – просмотр готового отчета на экране монитора, но без возможности распечатать, в макете дополнительно демонстрируется разметка, которую можно изменить;

- **Предварительный просмотр** – можно установить размер бумаги, параметры страницы, альбомную или книжную ориентацию бумаги. Выход из этого режима выполняется щелчком на пиктограмме ;

- **Режим макета** – позволяет изменять внешний вид отчета;

- **Конструктор**.

Переключение между режимами возможно, в частности, с помощью списка

, расположенного на вкладке **Главная** или **Конструктор**. Обратите внимание, что только при работе над отчетом в режимах **Конструктор** и **Макет** отображаются упомянутая ранее вкладка **Конструктор** и другие дополнительные вкладки **Упорядочить**, **Формат** и **Параметры страницы**.

## 6.2 Панель элементов


При конструировании отчета используется также **Панель элементов** для создания различных элементов управления (рисунок 52). Вызвать её можно из контекстного меню или на вкладке **Конструктор**, область **Элементы управления**.



Рисунок 52 – Панель элементов

К основным элементам управления относятся:



1) **Надпись** – предназначена для ввода поясняющего текста в любую часть отчета. Для создания надписи выполняют следующие действия:

- выбирают кнопку  на Панели элементов;
- фиксируют курсор в месте вставки и вводят текст;

2) **Поле** – предназначено для вывода значения записей из полей базы данных. Состоит из связанной с ним надписи и области вывода.

Для создания поля, выводящего данные из таблицы или запроса, достаточно перетащить их из списка полей в область данных отчета;

3) **Переключатели и флажки** – предназначены для выбора одного из двух вариантов: *Истина* или *Ложь*. Используются для логических полей. Для создания переключателя (флажка) выполняют следующие действия:

- выбирают левой клавишей мыши переключатель  или флажок  на панели элементов;


- наводят указатель мыши на требуемое поле в списке полей и перетаскивают его в область данных отчета.

Форматирование отдельных элементов отчета осуществляется кнопками, расположенными на вкладке **Формат**.

### 6.3 Создание отчета при помощи Мастера

Пусть необходимо создать отчет, в котором выводятся данные из таблицы **Перемещения** (поля НомерПунктаПропуска, ДатаПрохождения, Пошлина) и из таблицы **Картотека** (поля НомерПаспорта, Фамилия, Имя, Отчество).

Для создания этого отчета с помощью Мастера:

1) в окне базы данных выбирают вкладку **Создание**, затем в области **Отчеты** –  **Мастер отчетов**;

2) в появившемся диалоговом окне выполните следующие действия:

- в списке **Таблицы и запросы** выбирают таблицу **Перемещения**, поля из которой будут использованы в отчете;

- переносят поля **НомерПунктаПропуска**, **ДатаПрохождения**, **Пошлина** для отчета из списка *Доступные поля* в список *Выбранные поля*;

- повторяют вышеуказанные действия для таблицы **Картотека**, **Далее**;


3) в появившемся диалоговом окне выбирают вид представления данных – **Картотека**, **Далее**;

4) выбирают при помощи кнопки  поле для группировки – **НомерПунктаПропуска**, **Далее**;

5) при необходимости выполняют сортировку записей по полю **ДатаПрохождения**;

6) нажимают кнопку **Итоги** и устанавливают функцию **Sum** для поля **Пошлина**, **ОК**, **Далее**;

Какие итоговые значения необходимо вычислить?

Поле	Sum	Avg	Min	Max
Пошлина		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



- 7) выбирают *Макет отчета, Ориентацию страницы, Далее;*
- 8) выбирают стиль оформления отчета, *Далее;*
- 9) в последнем диалоговом окне задают имя отчета, например **Отчет 1**, и нажимают кнопку **Готово**.

#### 6.4 Создание вычисляемых и итоговых полей

*Вычисляемые поля* предназначены для вывода результатов расчета по одному или нескольким полям базы данных.

*Итоговые поля* предназначены для вывода итоговых значений по числовым полям базы данных.


Для создания вычисляемых (итоговых) полей:

- 1) открывают отчет в Конструкторе;
- 2) выбирают кнопку **Поле** –  на Панели элементов;
- 3) фиксируют поле в области данных (при создании вычисляемого поля) или в примечании отчета (при создании итогового поля);
- 4) выбирают команду **Свойства** из контекстного меню элемента поля **Свободный**, где переходят на вкладку **Все**. В строке **Данные** нажимают кнопку вызова построителя выражений ;
- 5) вводят расчетное выражение, например, =Avg([Пошлина]);
- 6) закрывают окно свойств поля;
- 7) вводят текст надписи, связанной с полем.

*Примечание* – При создании вычисляемых и итоговых полей расчетные формулы можно вводить с клавиатуры. При вводе необходимо соблюдать регистр букв в названиях полей.

#### 6.5 Создание диаграммы в отчете


Для построения диаграммы в отчете необходимо:


- 1) создать пустой отчет в Конструкторе;
- 2) выбрать щелчком мыши на панели элементов кнопку **Диаграмма** ;
- 3) щелкнуть мышью в области данных отчета. Будет запущен Мастер диаграмм;
- 4) выбрать таблицу или запрос для создания диаграммы. Нажать кнопку **Далее;**
- 5) выбрать поля, которые надо отобразить на диаграмме, **Далее;**
- 6) выбрать тип диаграммы, **Далее;**
- 7) перетащить кнопки полей в область образца.


Если требуется изменить вид группировки или вычислений, то дважды щелкают мышью на диаграмме числовое поле или поле даты. Нажать кнопку **Далее**;


8) ввести название диаграммы. Нажать кнопку **Готово**.

### Порядок выполнения работы

1 Используя  – **Мастер отчетов**, создайте отчет на основе таблицы **Картотека**. Измените заголовок отчета, введя в область заголовка текст *Картотека лиц, пересекающих границу*. Отчет сохраните с именем **отчКартотека**.

2 Используя  – **Конструктор отчетов**, создайте отчет на основе таблицы **ПунктыПропуска**. В область данных переместите все поля из таблицы **ПунктыПропуска**.

В заголовок отчета введите текст *Пункты пропуска*. Добавьте в заголовок отчета дату и время создания отчета, используя кнопку  **Дата и время**, расположенную в разделе **Колонтитулы** вкладки **Конструктор**. Разместите элементы полей отчета так, чтобы они не перекрывались. Выделите все надписи полей с нажатой клавишей **Ctrl** и в контекстном меню выберите **Выровнять / Слева**. Отчет сохраните с именем **отчПункты**.

3 Используя  – **Мастер отчетов** для данных из таблицы **Перемещения**, составьте итоговый отчет. В отчете сгруппируйте данные по пунктам пропуска. По каждому пункту пропуска рассчитайте суммарную пошлину. Выберите альбомную ориентацию бумаги, проследите, чтобы по ширине все поля поместились на один лист. В заголовке отчета введите текст *Итоговый отчет по пунктам пропуска*. Разместите поля отчета так, чтобы они не перекрывались. Отчет сохраните с именем **отчИтоги**.

4 Откройте **отчИтоги** в Конструкторе. Сохраните отчет под новым именем **отчИтогиЕвро**, используя команду **Файл / Сохранить объект как**. Отредактируйте **отчИтогиЕвро** следующим образом:

1) создайте в области данных поле, вычисляющее Пошлину в евро:  
 $=[\text{Пошлина}]/24150$ , имя поля – Пошлина в Евро;

3) создайте в примечании отчета итоговые поля, вычисляющее минимальное и максимальное значения Пошлины по всем записям базы данных:

$=\text{Min}([\text{Пошлина}])$ , имя поля – Минимальная пошлина;

$=\text{Max}([\text{Пошлина}])$ , имя поля – Максимальная пошлина;

4) создайте в примечании отчета итоговое поле, вычисляющее среднее значение Пошлины по всей базе данных:

$=\text{Avg}([\text{Пошлина}])$ , имя поля – Среднее значение пошлины;

5) для всех итоговых полей установите формат Евро;

6) в нижний колонтитул вставьте нумерацию страниц: область Колонти-тулы, Номер страницы;

7) сохраните изменения в отчете.

5 Используя Конструктор отчетов, создайте диаграмму на основании **зпрПошлина**. Задайте название диаграммы – Итоговая пошлина по пунктам пропуска. Установите подписи данных на диаграмме. Диаграмму сохраните с именем Итоговая диаграмма.

**Задание.** Отредактируйте **Отчет 1** следующим образом:

1) откройте **Отчет 1** в Конструкторе;

2) создайте в области данных поле, вычисляющее пошлину в евро:

$$=[\text{Пошлина}]/2,4;$$

3) создайте в примечании отчета итоговое поле, вычисляющее минимальное и максимальное значения пошлины по всей базе данных:

$$=Min([\text{Пошлина}]);$$

$$=Max([\text{Пошлина}]);$$

4) создайте в примечании отчета итоговое поле, вычисляющее среднее значение пошлины по всей базе данных:

$$=Avg([\text{Пошлина}]);$$

5) в нижний колонтитул вставьте нумерацию страниц: область **Колонти-тулы, Номер страницы**.

6) сохраните отчет с именем **Отчет 2**.

### Контрольные вопросы

1 Что такое отчет?

2 Основные режимы создания отчетов.

3 Структура отчета. Настройка структурных элементов отчета.

4 Элемент управления Надпись. Назначение. Создание.

5 Элемент управления Поле. Назначение.

6 Вычисляемые и итоговые поля. Назначение. Создание.

7 Функции вычисления итогов.

8 Добавление диаграммы в отчет.




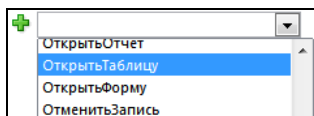
## 7 КНОПОЧНАЯ ФОРМА

### 7.1 Создание макросов

*Макрос* – набор из одной или нескольких макрокоманд, каждая из которых выполняет определенное действие.

Для создания макроса:

- 1) в окне базы данных на вкладке **Создание** в области **Макросы** выбирают кнопку  – **Макрос**;
- 2) в списке указывают действие, которое должен выполнить макрос:



Откроется окно, представленное на рисунке 53;

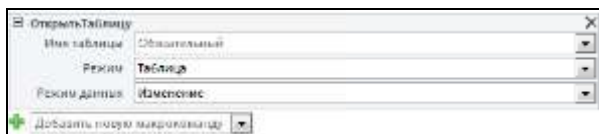


Рисунок 53 – Окно настройки макроса

3) в поле **Имя таблицы** выбирают имя таблицы, для открытия которой создается макрос. В списке **Режим** указывают режим открытия таблицы: Таблица, Конструктор, Просмотр, Сводная таблица или Сводная диаграмма. В списке **Режим данных** выбирают, для чего будет открыта таблица: для добавления данных, для изменения данных или только для чтения;

4) закрывают макрос, выполняя сохранение. При сохранении указывают имя макроса, соответствующее выполняемому действию, например, «Открыть таблицу ПунктыПропуска».

*Примечание* – Макросы для открытия таблиц и запросов будут использованы при создании кнопок Главной кнопочной формы. Макросы для открытия форм и отчетов создавать необязательно. В MS Access 2010 и выше соответствующие команды предусмотрены при разработке Главной кнопочной формы.

## 7.2 Главная кнопочная форма

*Главная кнопочная форма* предоставляет интерфейс пользователя для доступа ко всем объектам базы данных. Для создания кнопочной формы:

1) на вкладке **Работа с базами данных** в области **Администратор** выбирают **Диспетчер кнопочных форм**. Если область **Администратор** отсутствует на вкладке, то следует выполнить команду **Файл / Параметры / Настройка ленты** и поместить команду **Диспетчер кнопочных форм** во вновь созданную пользовательскую группу;

2) на вопрос «Не удастся найти кнопочную форму в этой базе данных. Создать кнопочную форму?», выбирают ответ **Да**;

3) в диалоге **Диспетчер кнопочных форм** представлена **Главная кнопочная форма**, которая открывается по умолчанию (рисунок 54).

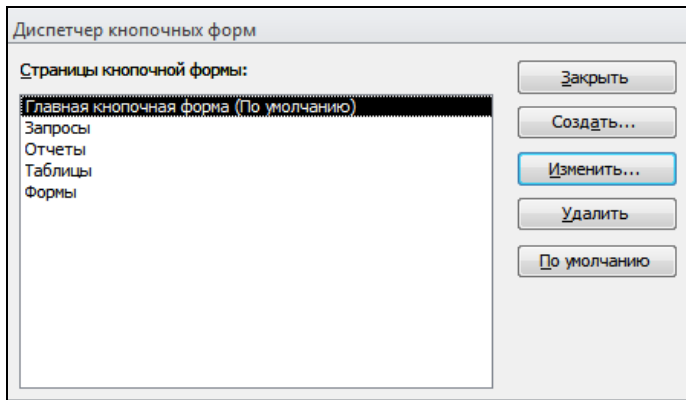


Рисунок 54 – Создание кнопочных форм

В этом окне могут быть сформированы дополнительные кнопочные формы, к которым будет осуществляться переход из главной кнопочной формы. Для этого надо нажать кнопку **Создать** и ввести название кнопочной формы. Например, для создания кнопочной формы **Таблицы** нажимают **Создать**, вводят с клавиатуры **Таблицы** и нажимают **ОК**. Аналогично создаются кнопочные формы **Запросы**, **Формы**, **Отчеты** (см. рисунок 54).

Кнопка **Изменить** вызывает диалог **Изменение страницы кнопочной формы** и позволяет настраивать элементы выделенной кнопочной формы. Например, выделив кнопочную форму **Таблицы** и нажав кнопку **Изменить**, можно создать кнопки для открытия всех таблиц базы данных и кнопку перехода на главную кнопочную форму (рисунок 55). В диалоге **Изменение элемента кнопочной формы** задается *текст* подписи к кнопке, *команда*, выполняемая по нажатию этой кнопки, и *выбирается имя объекта*, который должен быть открыт. Команды могут быть следующие: открыть отчет, от-

крыть форму, выполнить макрос, выйти из приложения. Для открытия таблиц и запросов необходимо выбрать команду **Выполнить макрос**. Макросы для открытия таблиц и запросов создаются заранее.

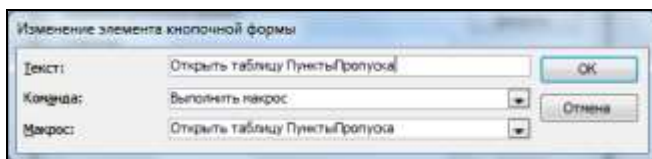


Рисунок 55 – Изменение элемента кнопочной формы

При создании кнопки возврата на главную кнопочную форму в окне задают: **Текст** – На главную, **Команда** – Перейти к кнопочной форме, **Кнопочная форма** – Главная кнопочная форма. Кнопочная форма Таблицы представлена на рисунке 56.

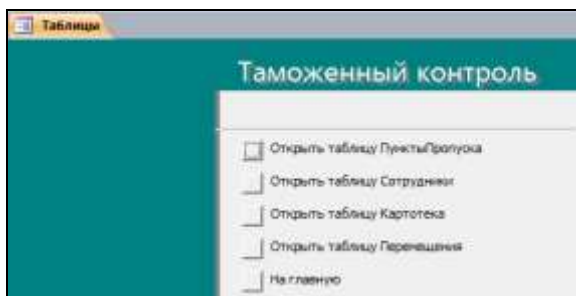


Рисунок 56 – Кнопочная форма Таблицы

Для связывания Главной кнопочной формы с кнопочными формами Таблицы, Запросы и др. и создания кнопок Главной кнопочной формы:

1) выбирают имя Главная кнопочная форма (По умолчанию) и нажимают **Изменить**;

2) нажимают кнопку **Создать** и в окне **Изменение элемента кнопочной формы** в строке **Текст** вводят **Таблицы**, в строке **Команда** выбирают из списка **Перейти к кнопочной форме**, в строке **Кнопочная форма** выбирают имя кнопочной формы, к которой будет выполняться переход – **Таблицы**. Нажимают **ОК**. Указанные действия повторяют несколько раз для создания перехода к кнопочным формам **Формы, Запросы, Отчеты**.

Для создания кнопки **Выход**:

1) нажимают кнопку **Создать** и в окне **Изменение элемента кнопочной формы** в строке **Текст** вводят текст **Выход**;

2) в строке **Команда** выбирают из списка **Выйти из приложения**.

*Примечание* – Используйте кнопки **Вверх** и **Вниз** для правильного размещения кнопок на форме.

Таким образом, созданы кнопки на панели Главной кнопочной формы (рисунок 57). Для возврата в окно **Диспетчер кнопочных форм** нажимают кнопку **Заккрыть**.

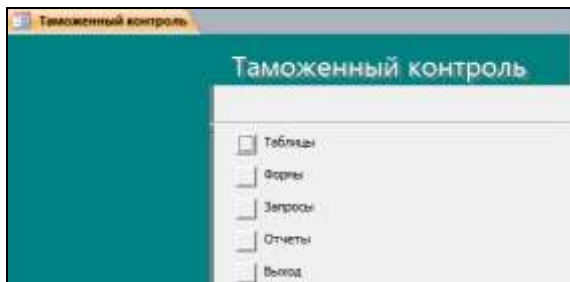




Рисунок 57 – Главная кнопочная форма

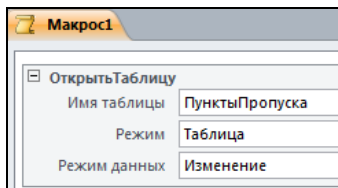
При создании кнопочной формы с помощью Диспетчера кнопочных форм на вкладке **Таблицы** автоматически создается таблица «**Switch board Items**», которая описывает текст и действия кнопок.

**ВНИМАНИЕ!** Нельзя удалять таблицу «**Switch board Items**» и Главную кнопочную форму!

Чтобы *изменить кнопочную форму* или *добавить кнопки*, надо на вкладке **Работа с базами данных** в области **Администратор** выбрать **Диспетчер кнопочных форм**. Будет открыта главная кнопочная форма, в которой можно выполнить добавление кнопок, изменения.

### Порядок выполнения работы

1 Создайте макрос для открытия таблицы **ПунктыПропуска**. Для этого на вкладке **Создание** в области **Макросы** выберите кнопку , затем в списке действий  **Добавить новую макрокоманду** – **ОткрытьТаблицу**. Выберите имя открываемой таблицы **ПунктыПропуска**, укажите режим открытия **Таблица** и режим данных **Изменение**.



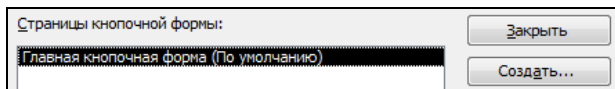
Закройте макрос, выполнив сохранение с именем **Открыть таблицу ПунктыПропуска**.

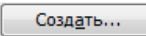
2 Создайте еще три макроса для открытия таблиц **Сотрудники**, **Картотека**, **Перемещения**.

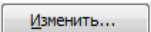
3 Создайте макросы для открытия семи запросов по выбору.

4 Создайте главную кнопочную форму. Для этого на вкладке **Работа с базами данных** в области **Администратор** выберите **Диспетчер кнопочных форм**; на вопрос «Не удастся найти кнопочную форму в этой базе данных. Создать кнопочную форму?» выберите **Да**.

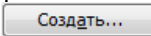
В диспетчере кнопочных форм присутствует одна страница, т. е. форма, которая будет открываться по умолчанию.

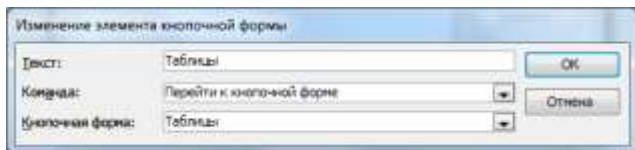


5 Используя кнопку , добавьте еще четыре страницы кнопочной формы: **Таблицы**, **Формы**, **Запросы** и **Отчеты** (см. рисунок 54). Они представляют собой кнопочные формы (пока пустые), которые будут открываться с главной формы.

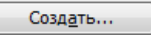
6 Измените главную кнопочную форму: добавьте на нее элементы (кнопки), с помощью которых будет открываться созданные ранее страницы кнопочной формы. Для этого выделите **Главная кнопочная форма (По умолчанию)** и нажмите .

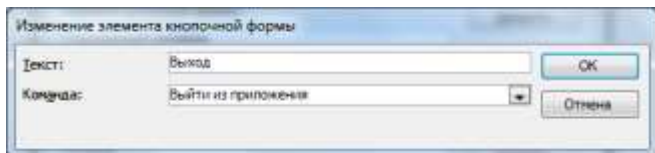
В окне **Изменение страницы кнопочной формы** введите новое название главной кнопочной формы *Таможенный контроль*.


Используя кнопку , добавьте на главную кнопочную форму элемент (кнопку), щелчком на которой будет открываться форма **Таблицы**, и настройте ее.



Аналогично добавьте на главную кнопочную форму кнопки **Формы**, **Запросы**, **Отчеты** и измените их для открытия одноименных кнопочных форм.

7 Добавьте на главную кнопочную форму кнопку **Выход**. Для этого нажмите , введите текст *Выход* и выберите команду *Выйти из приложения*.



Измененная страница главной кнопочной формы должна выглядеть, как представлено на рисунке 58. Нажмите кнопку .

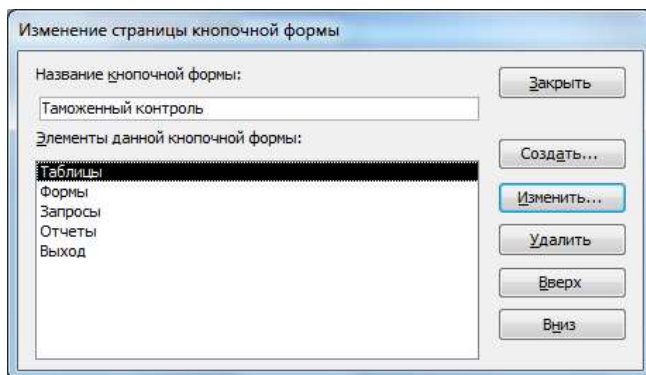
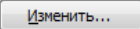
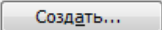
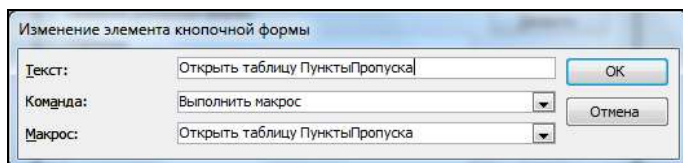


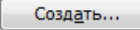
Рисунок 58 – Изменение главной страницы кнопочной формы

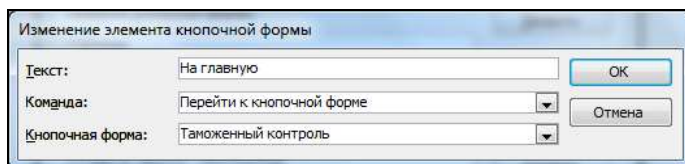
8 В окне **Диспетчер кнопочных форм** выделите страницу **Таблицы** и нажмите . В окне **Изменение страницы кнопочной формы** можно переименовать форму.


Создайте элементы данной кнопочной формы. Для этого щелкните , введите текст *Открыть таблицу ПунктыПропуска*, из раскрывающегося списка выберите команду *Выполнить макрос* и укажите макрос с именем *Открыть таблицу ПунктыПропуска*.



9 Аналогично создайте кнопки для открытия остальных таблиц базы данных.

10 На кнопочной форме **Таблицы** создайте кнопку для возвращения на главную кнопочную форму. Для этого щелкните , введите текст *На главную*, из раскрывающегося списка выберите команду *Перейти к кнопочной форме* и укажите кнопочную форму *Таможенный контроль*.



Измененная страница кнопочной формы **Таблицы** должна выглядеть, как представлено на рисунке 59. Нажмите кнопку .

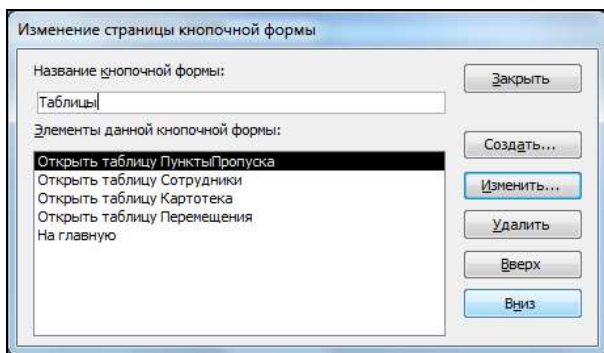



Рисунок 59 – Изменение кнопочной формы Таблицы

11 Находясь в окне **Диспетчер кнопочных форм**, выделите страницу **Формы**, нажмите кнопку . Добавьте на нее элементы, позволяющие открывать формы базы данных и кнопку возвращения на главную форму. Для создания кнопок открывающих формы, достаточно в списке команд выбрать *Открыть форму для изменения*.

12 Аналогично измените формы **Запросы** и **Отчеты**.

13 Закройте диспетчер кнопочных форм. Откройте форму **Кнопочная форма** из списка всех объектов *MS Access* и проверьте корректности ее работы. При необходимости внесите изменения в **Диспетчере кнопочных форм**.

### Контрольные вопросы

- 1 Что такое макрос? Назначение макросов.
- 2 Порядок создания макросов.
- 3 Для чего предназначена главная кнопочная форма?
- 4 Как создать главную кнопочную форму.
- 5 Создание страниц кнопочной формы.
- 6 Создание и изменение элементов кнопочных форм.

## **СПИСОК ИСПОЛЬЗОВАННОЙ И РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ**

1 **Бекаревич, Ю. Б.** Самоучитель Microsoft Access 2013 / Ю. Б. Бекаревич, Н. В. Пушкина. – СПб. : БХВ-Петербург, 2014. – 464 с.

2 **Информатика : Базовый курс:** учеб. пособие для студ. высш. техн. учеб. заведений / С. В. Симонович [и др.] ; под общ. ред. С. В. Симоновича. – СПб. : Питер, 2005. – 640 с.

3 **Иоффе, Л. А.** Основы компьютерных информационных технологий : учеб.-метод. пособие / Л. А. Иоффе, Т. Л. Шинкевич, Т. А. Голдобина ; М-во образования Респ. Беларусь, Белорус. гос. ун-т трансп. – Гомель : БелГУТ, 2010. – 166 с.

4 **Лыч, Ю. П.** Технологии организации, хранения и обработки данных / Ю. П. Лыч, Т. Л. Шинкевич. – Гомель : БелГУТ, 2003. – Ч. 1 : Основы создания баз данных : пособие для сам. работы. – 157 с.

5 **Лыч, Ю. П.** Технологии организации, хранения и обработки данных / Ю. П. Лыч. – Гомель : БелГУТ, 2003. – Ч. 2 : Основы программирования в СУБД Access : пособие для сам. работы. – 68 с.

6 **Морозевич, А. Н.** Основы экономической информатики : учеб. пособие / А. Н. Морозевич. – Минск : БГЭУ, 2006. – 438 с.

7 **Петров, Г. А.** Базы данных : пособие / Г. А. Петров, С. В. Тихов, В. П. Яковлев. – СПб. : СПбГТУ РП, 2015. – 74 с.



Учебное издание

*ШИНКЕВИЧ Татьяна Леонидовна*  
*ГОЛДОБИНА Татьяна Александровна*

БАЗЫ ДАННЫХ В ТАМОЖЕННОМ ДЕЛЕ

Учебно-методическое пособие

Редактор *А. А. Павлюченкова*  
Технический редактор *В. Н. Кучерова*  
Корректор *Т. А. Пугач*

Подписано в печать 20.09.2016 г. Формат 60×84  $\frac{1}{16}$ .  
Бумага офсетная. Гарнитура Times. Печать на ризографе.  
Усл. печ. л. 4,65. Уч.-изд. л. 4,11. Тираж 150 экз.  
Зак. № . Изд. № 60.

Издатель и полиграфическое исполнение:  
Белорусский государственный университет транспорта.  
Свидетельство о государственной регистрации издателя, изготовителя,  
распространителя печатных изданий  
№ 1/361 от 13.06.2014.  
№ 2/104 от 01.04.2014.  
Ул. Кирова, 34, 246653, г. Гомель