

## ПОСТАНОВКА ЗАДАЧИ

Создать базу для отслеживания телефонных разговоров сотрудников компании.

Для решения этой задачи нам понадобятся 3 основные таблицы, но будут и вспомогательные. В первой таблице будет находиться информация о сотрудниках компании. Во второй – о телефонах, принадлежащих (или закрепленных) за сотрудниками и в третьей – данные о каждом совершенном звонке.

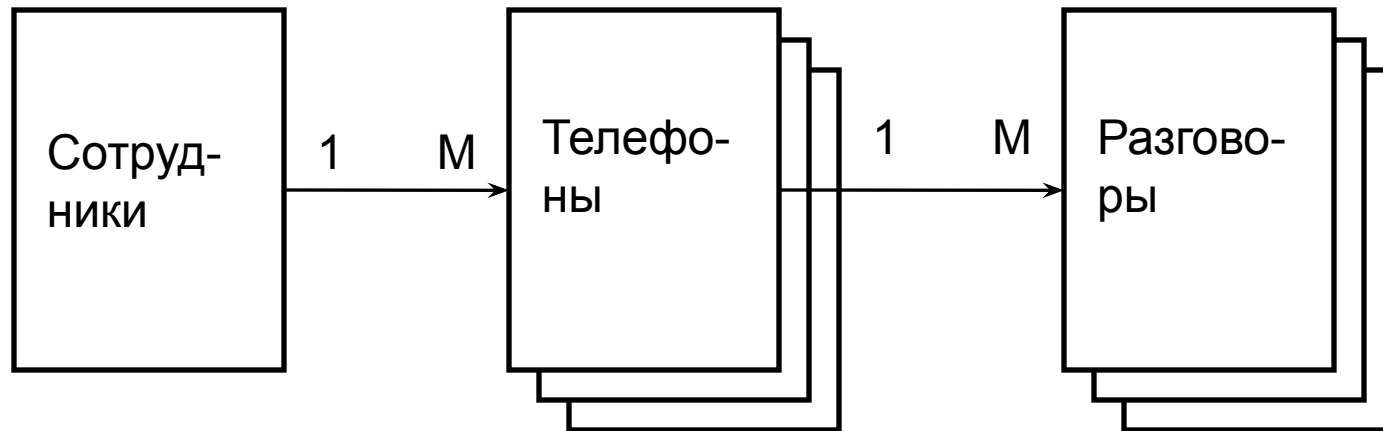


Рис.1 Общая структура взаимодействия таблиц

Обе связи в данной базе имеют размерность 1:M, так как за каждым сотрудником может быть закреплен не один а несколько телефонов. Например, стационарный и мобильный. Кроме того, у некоторые сотрудники могут вообще не иметь закрепленных за ними телефонов. (Однако, от этого они не перестают быть сотрудниками компании и, соответственно, должны присутствовать в списке сотрудников.) Но, каждый телефон закреплен за одним и только одним сотрудником. (Ситуацию один телефон на всех сотрудников комнаты мы не рассматриваем)

Аналогично и с разговорами. С каждого телефона может быть сделано (и каждым телефоном может быть принято) заранее не определенное число звонков. Но каждый звонок может быть связан с одним и только одним аппаратом. (Опять –таки ситуацию всяких “телефонных конференций” мы не рассматриваем.)

Итак, в соответствии с этим заданием, начнем создавать базу данных , используя в качестве СУБД “ACCESS-2007”

Откроем ACCESS и на странице “Приступая к работе с Microsoft Office Access” выберем “Новая база данных”

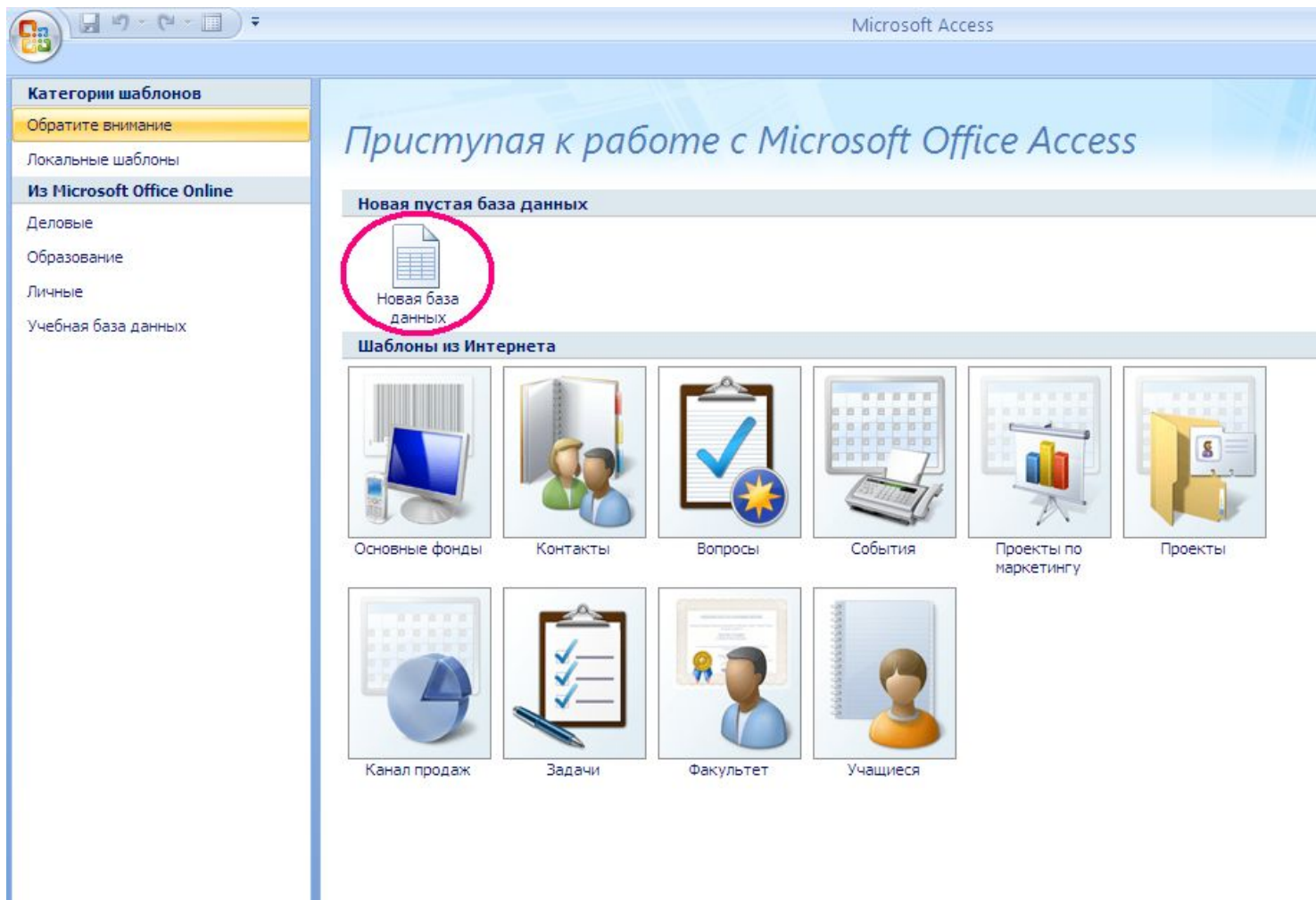


Рис.3. Страница “Приступая к работе с Microsoft Office Access”

После щелчка по этой кнопке справа появится диалог создания базы данных, где нам предлагается задать имя новой создаваемой базе данных. Назовем ее “Person” и оставим стандартное расширение “accdb”. (Что означает, что мы создаем базу формата “ACCESS – 2007”, а не совместимого с более ранними версиями ACCESS формата. Базы старого формата имеют расширение “mdb”)

Нажмем теперь кнопку “создать”. (Если необходимо изменить место расположения базы следует сначала кликнуть на изображение папки (синий кружок) и выбрать нужное место расположения базы)

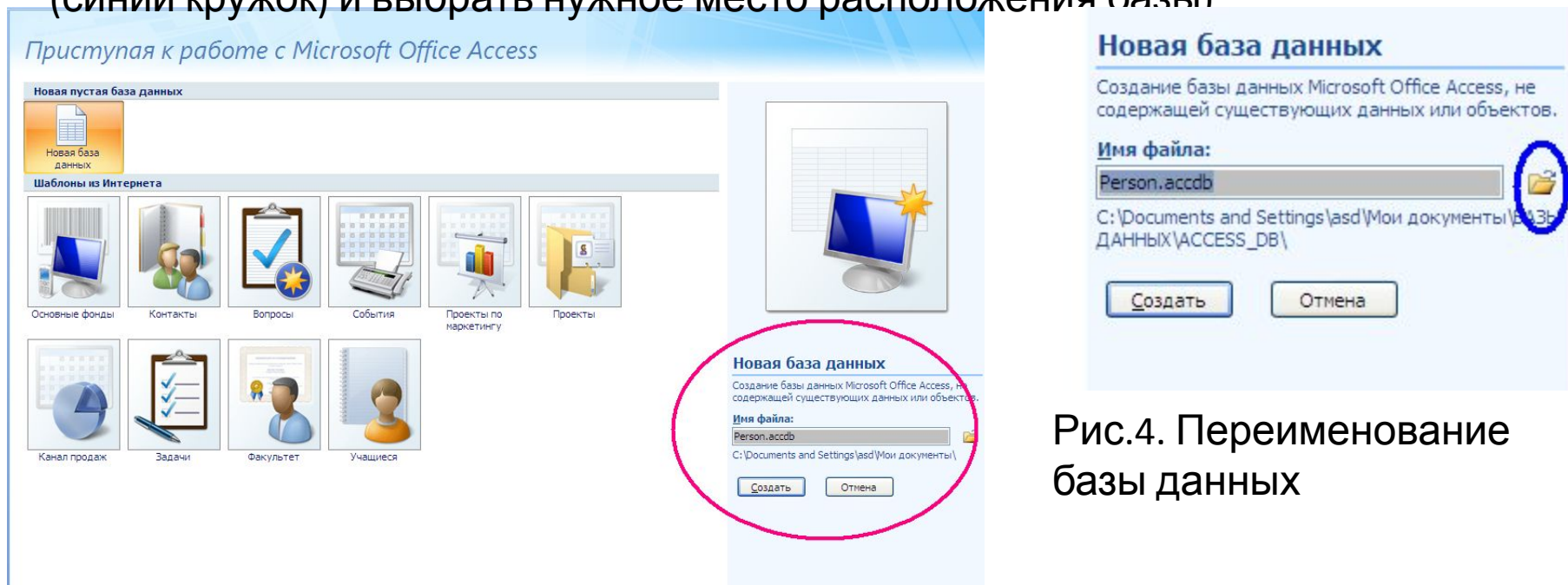


Рис.3. Диалог создания базы данных

Рис.4. Переименование базы данных

Появится окно создания новой таблицы. По умолчанию этой таблице присвоено имя “Таблица1”. Существует несколько способов создания таблиц. В данном случае (по умолчанию) используется “Режим таблицы”

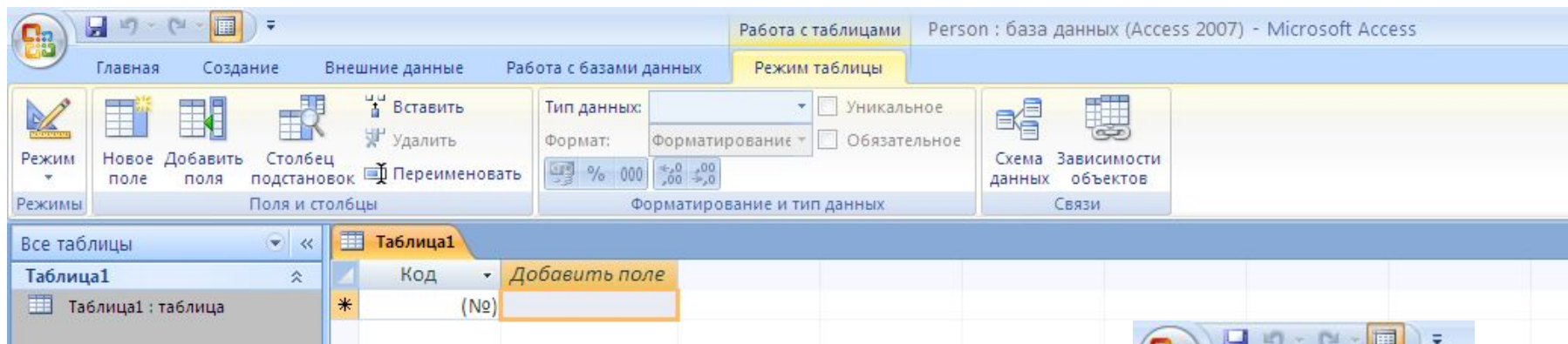


Рис.5. Создание базы данных в режиме таблицы

Мы можем либо продолжить создание БД в этом режиме, либо перейти в режим конструктора. Выбор режима осуществляется нажатием на кнопку “Режим” и выбора одного из предложенных режимов. (Рис. 6). Пока не будем менять режим работы – останемся в режиме таблицы. Обратим внимание, что по умолчанию СУБД автоматически создала нам поле с именем “Код” и типом данных “Счетчик”. (Об этом свидетельствует символ “(№)” в поле “Код” на рис.5.)

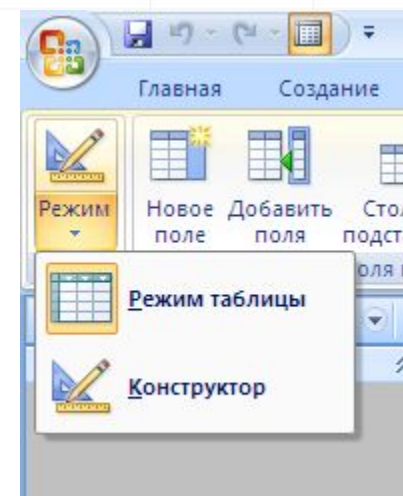


Рис.6. Выбор режима работы с базой данных

Посмотрим на свойства этого поля более внимательно. Для этого выберем это поле в таблице (Щелкнем на слове “Код” в заголовке таблицы. ) При этом на панели “Форматирование и тип данных” отразятся текущие свойства этого поля (рис.7.) . Мы видим, что тип данных действительно “Счетчик”. Кроме того помечено, что данное поле – уникальное.

Тип поля “Счетчик” означает, что данные для этого поля автоматически генерируются СУБД в виде последовательности автоматически возрастающих на единицу целых чисел, начиная с 1. Тем самым гарантируется уникальность, т.е. неповторяемость значений в данном поле.

Теперь давайте сохраним эту автоматически созданную заготовку таблицы в базу данных. Для этого нажмем на кнопку “сохранить” (рис. 8). Поскольку таблица ранее не сохранялась нас попросят указать имя новой таблицы. Пусть это будет “Person”. Введем и нажмем ОК (рис.9)

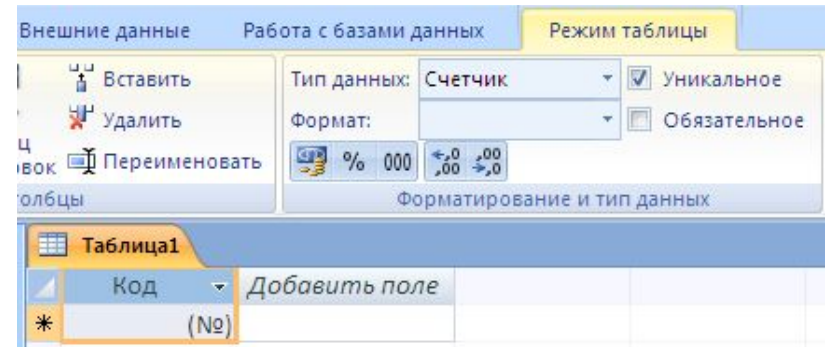


Рис.7. Свойства поля данных “Код” таблицы “Таблица1”



Рис.8. Кнопка “Сохранить”

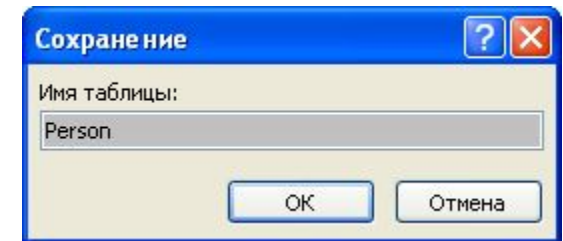


Рис.9. Диалог сохранения таблицы.

Важное техническое замечание. Операцию сохранения следует выполнять достаточно часто, т.к. все что мы делаем происходит в оперативной памяти компьютера и при сбое в его работе мы рискуем потерять наши наработки, которые не были сохранены на жестком диске.

Теперь обратим внимание на столбец “Добавить поле”. Этот столбец позволяет нам создавать столбцы просто на основании вводимых данных. При этом режиме все параметры создаваемого поля определяются СУБД автоматически на основании введенной информации.

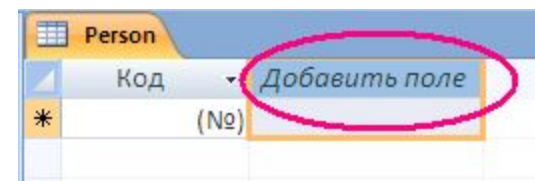


Рис.10. Столбец “Добавить поле”

Поскольку здесь работает “автомат” мы должны ввести такие данные, чтобы СУБД смогла “понять” что же мы хотим. Приведем несколько примеров. Введем, например, значение 12345 и нажмем “Enter”.

СУБД автоматически сгенерировало для нас поле с именем “Поле1”. Какой тип данных получился – “Числовой”. Т.е. в поле с именем “Поле1” мы сможем хранить только числа. Если это то, что нам надо – то все в порядке. А если мы собираемся хранить в этом поле, например, номер документа?

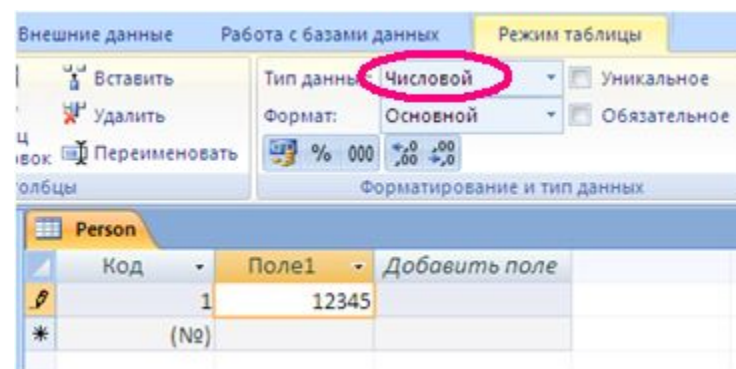


Рис.11. Автоматически сгенерированное числовое поле

Ведь несмотря на то, что по внешнему виду номер документа очень часто это число, но

- во-первых это не всегда так. 123/10 или 04/1028-Т – это тоже вполне допустимые номера документов, а
- во-вторых по своей сути номер документа не может участвовать в математических операциях. Например, складывать номера 2-х документов достаточно бессмысленно.

Следовательно в этом случае “автомат” определения типа поля сработал не так, как нам бы хотелось и “не угадал” наше намерение.

Еще один пример. Введем : 10.04.2010. Какой тип данных получился? – Дата/время. Если это так – то все отлично. Но ведь это может быть и номер документа. Например цифры до первой точки – номер документа по порядку в пределах месяца, а далее месяц и год когда документ был принят.

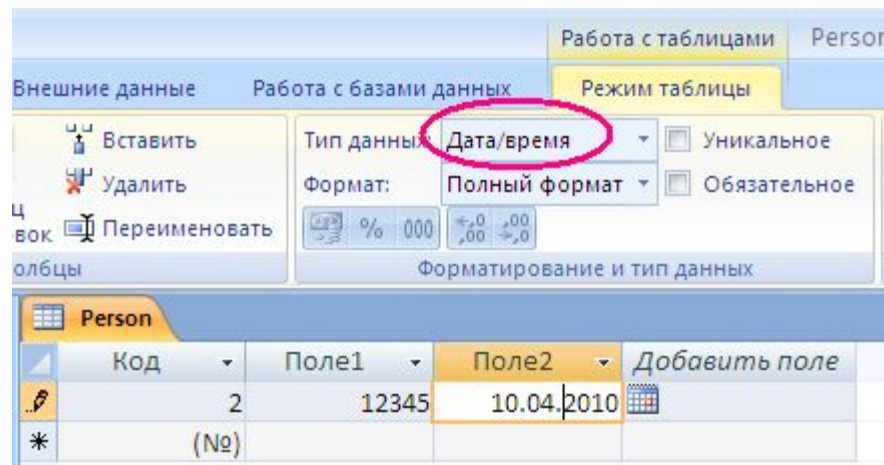


Рис.12. Автоматически сгенерированное поле типа “дата”



Отсюда вывод – следует тщательно продумывать какие именно данные мы вводим в качестве образца, на основании которых СУБД будет генерировать для нас поле. Кроме того, всегда необходимо контролировать результаты автоматической генерации полей. И, при необходимости, корректировать результаты работы автомата. Сделать это очень легко.

Достаточно нажать на кнопку раскрытия ниспадающего меню рядом с типом данных. После этого выбираем новый тип данных. После щелчка на нем произойдет изменение типа данных текущего поля. Точно также мы можем определить (или переопределить) все другие параметры создаваемого нами

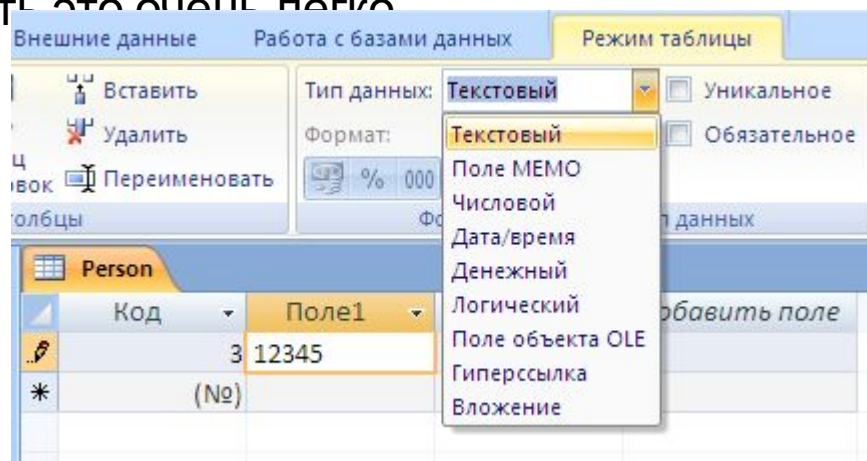


Рис.13. Переопределение типа поля.

Например, мы можем определить поле как “уникальное”. Это означает, что СУБД будет следить за “уникальностью” вводимых значений и не позволит ввести в БД две записи с одинаковыми значениями в данном поле.

Кроме того, поле может быть помечено как “обязательное”. При наличии данного флага СУБД гарантирует отсутствие в БД записей с “пустым” значением данного поля.

Кроме того, для полей некоторых типов на панели форматирования могут быть определен формат представления информации.

Так, для полей типа “дата/время” может быть уточнен формат представления данных (рис.14)

Для числовых полей также может быть выбран один из следующих форматов (рис.15). Кроме того, нажав на соответствующую кнопку для числовых полей можно:

- применить денежный формат:

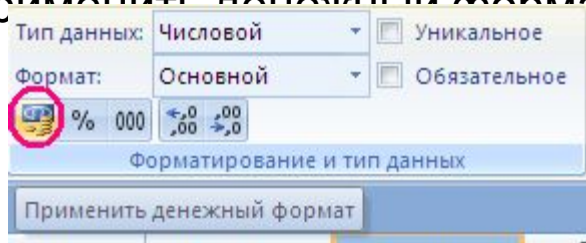


Рис.16. Кнопка денежного формата

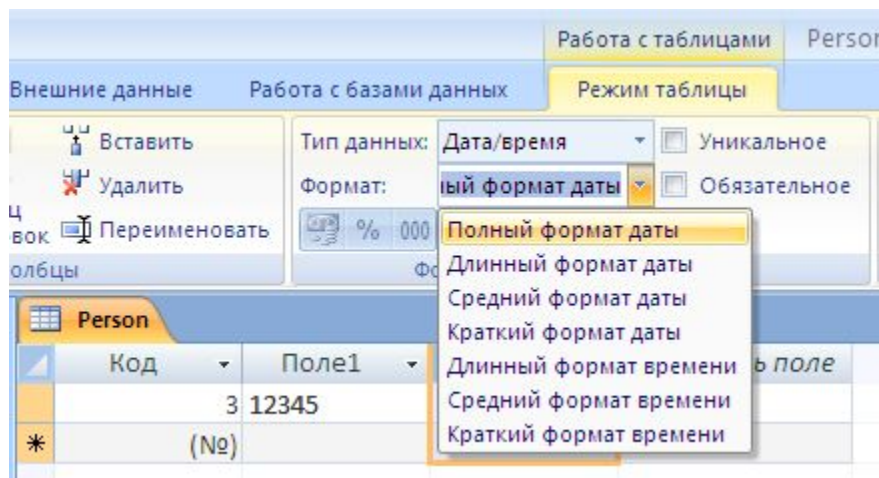


Рис.14. Определение формата даты/времени для полей этого типа.

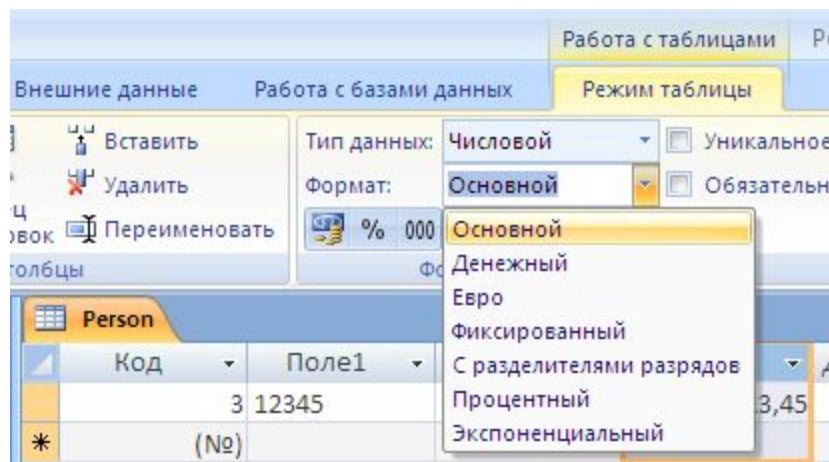


Рис.15. Определение формата для числовых полей.

- применить процентный формат:
- применить финансовый формат:

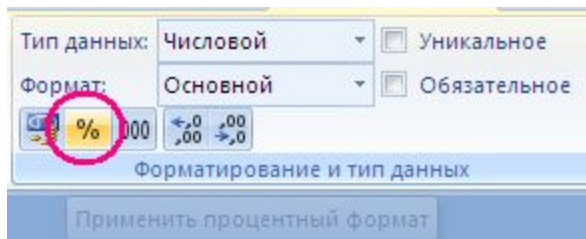


Рис.17.Кнопка процентного формата

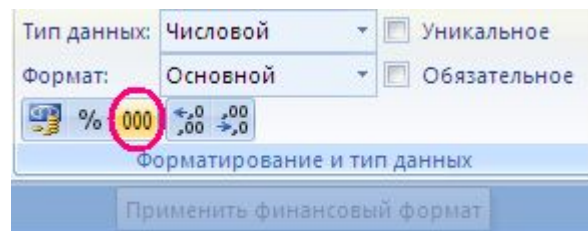


Рис.18.Кнопка финансового формата

- увеличить или уменьшить количество десятичных разрядов после запятой:

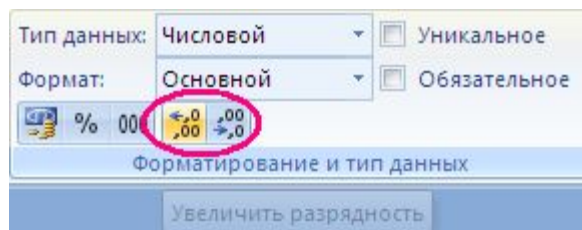


Рис.19.Кнопки изменения разрядности числа после запятой

Теперь познакомимся с возможностями еще одной панели – “Поля и столбцы”

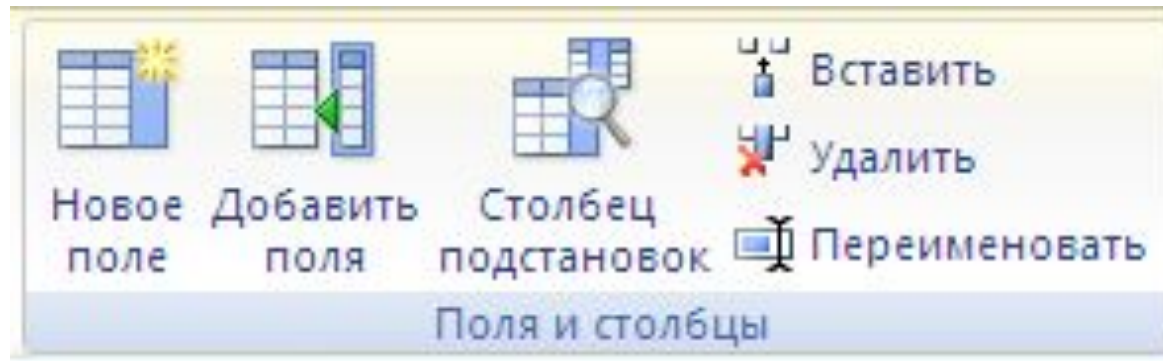


Рис.20. Панель “Поля и столбцы”

Как видно из самих названий, кнопки на этой панели позволяют нам:

- Создать новое поле
- Добавить поле на основе определения поля в другой таблице
- Определить для поля столбец подстановок
- Вставить новое поле между 2-мя существующими полями (или перед первым полем) базы данных
- Удалить поле
- Переименовать поле

Ознакомившись со всеми основными возможностями по созданию и манипулированию свойствами полей баз данных, создадим базу данных Person такой, какой она должна быть. А поэтому, сначала уничтожим все созданные нами в этой базе тренировочные поля. А для этого сначала выберем все ненужные поля щелкая на них мышкой, одновременно удерживая нажатой клавишу "Ctrl". (Такой прием работы называется "множественным выделением объектов") После того как все поля окажутся "выбранными" нажмем кнопку "Удалить". На вопросы Access о несоблюдении целостности базой ответим "Да".

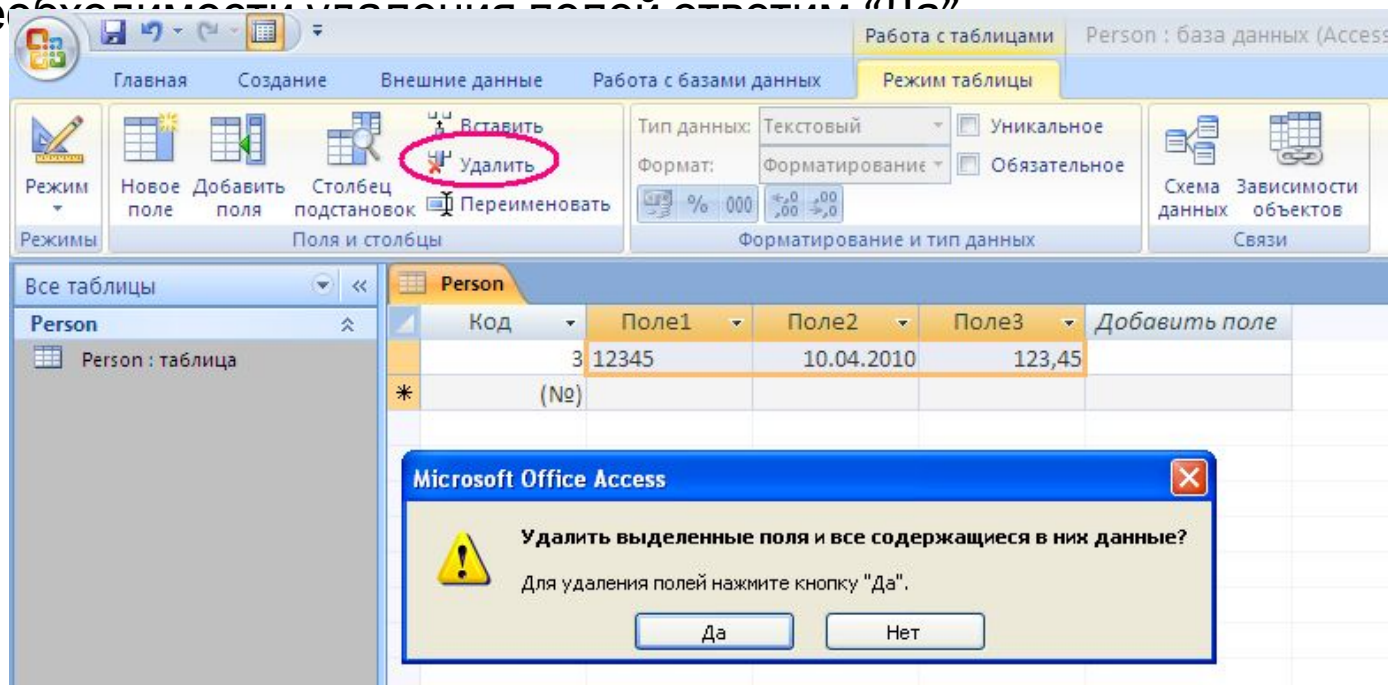
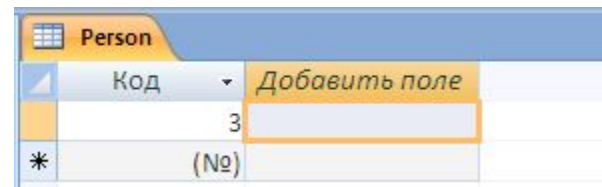


Рис.21. Удаление полей из базы данных

Итак мы вернулись к “пустой” таблице с единственным полем “Код”. Теперь на основе полученных знаний зададим следующий шаблон для генерации полей:

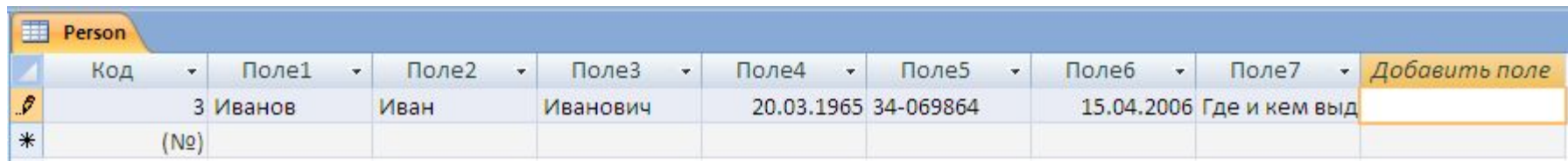
- Иванов
- Иван
- Иванович
- 20.03.1965
- 34-069864
- 15.04.2006
- Где и кем выдан

В результате получаем (рис.23)



Код	Добавить поле
3	
*	(№)

Рис.22. Исходная (пустая) таблица



Код	Поле1	Поле2	Поле3	Поле4	Поле5	Поле6	Поле7	Добавить поле
3	Иванов	Иван	Иванович	20.03.1965	34-069864	15.04.2006	Где и кем выд	
*	(№)							

Рис.23. Автоматически сгенерированные поля таблицы Person.

Путем проверки атрибутов полей убеждаемся, что поля 1-3,5 и 7 являются текстовыми, а 4 и 6 – типа “дата/время”. Настало время дать нашим полям осмысленные имена. Последовательно выбирая поля с 1 по 7 и нажимая кнопку “Переименовать” переименуем все поля так как полагано на рис.

24

Код	Фамилия	Имя	Отчество	Дата рождения	Паспорт	Паспорт дата выдачи	Паспорт кем выдан
3	Иванов	Иван	Иванович	20.03.1965	34-069864	15.04.2006	Где и кем выдан
*	(№)						

Рис.24. Переименованные поля таблицы Person.

Итак основная задача выполнена – таблица сформирована. Можно теперь заняться тонкой настройкой. Например, определим для даты рождения “Средний формат даты”, а для даты выдачи паспорта – “Длинный формат даты”. Результат- на рис. 25.

Код	Фамилия	Имя	Отчество	Дата рождения	Паспорт	Паспорт дата выдачи	Паспорт кем выдан
3	Иванов	Иван	Иванович	20-мар-1965	34-069864	15 апреля 2006 г.	Где и кем выдан
*	(№)						

Рис.25. Изменение форматов полей “Дата рождения” и “Паспорт дата выдачи” таблицы Person.

Теперь попробуем создать форму для ввода данных в нашу базу данных. В левой части выберем нашу базу Person. Вверху выберем ленту “Создание”.

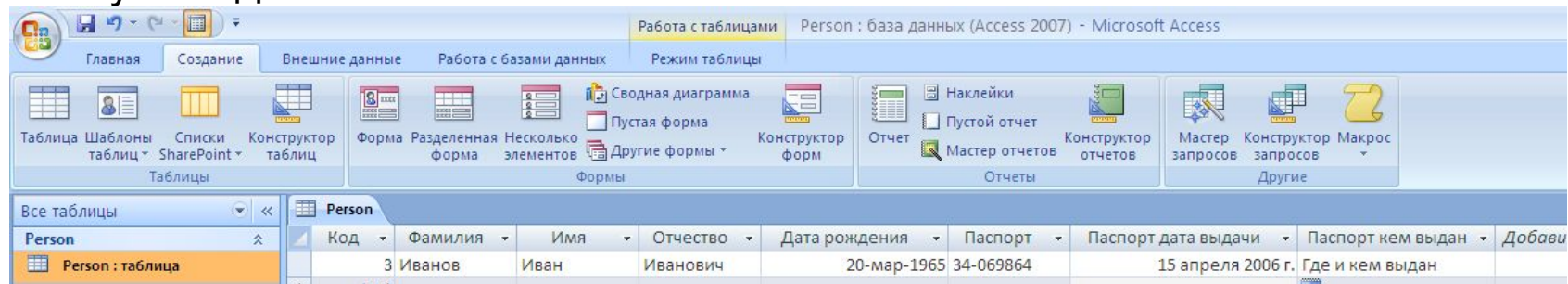


Рис.26. Лента “Создание”.

На панели “Формы” несколько возможных форм ввода данных. Выберем “Разделенную форму”. Просто нажмем эту кнопку (при выбранной таблице).

И, собственно, все. СУБД “ACCESS” автоматически создает вполне приемлемую форму для интерактивного ввода данных.



Person Person

## Person

+

Код: 3

Дата рождения: 20-мар-1965

Фамилия: Иванов

Паспорт: 34-069864

Имя: Иван

Паспорт дата выдачи: 15 апреля 2006 г.

Отчество: Иванович

Паспорт кем выдан: Где и кем выдан

Код	Фамилия	Имя	Отчество	Дата рождения	Паспорт	Паспорт дата выдачи	Паспорт кем выдан
3	Иванов	Иван	Иванович	20-мар-1965	34-069864	15 апреля 2006 г.	Где и кем выдан
* (№)							

Рис.27. Автоматически сгенерированная форма для ввода информации для ввода данных.

Конечно, эту форму можно и нужно подправить. Например, можно выделить поле “Код” и соответствующее этому полю надпись (удерживая “Ctrl”). После этого щелкнем правой кнопкой мыши и нажмем “свойства”.

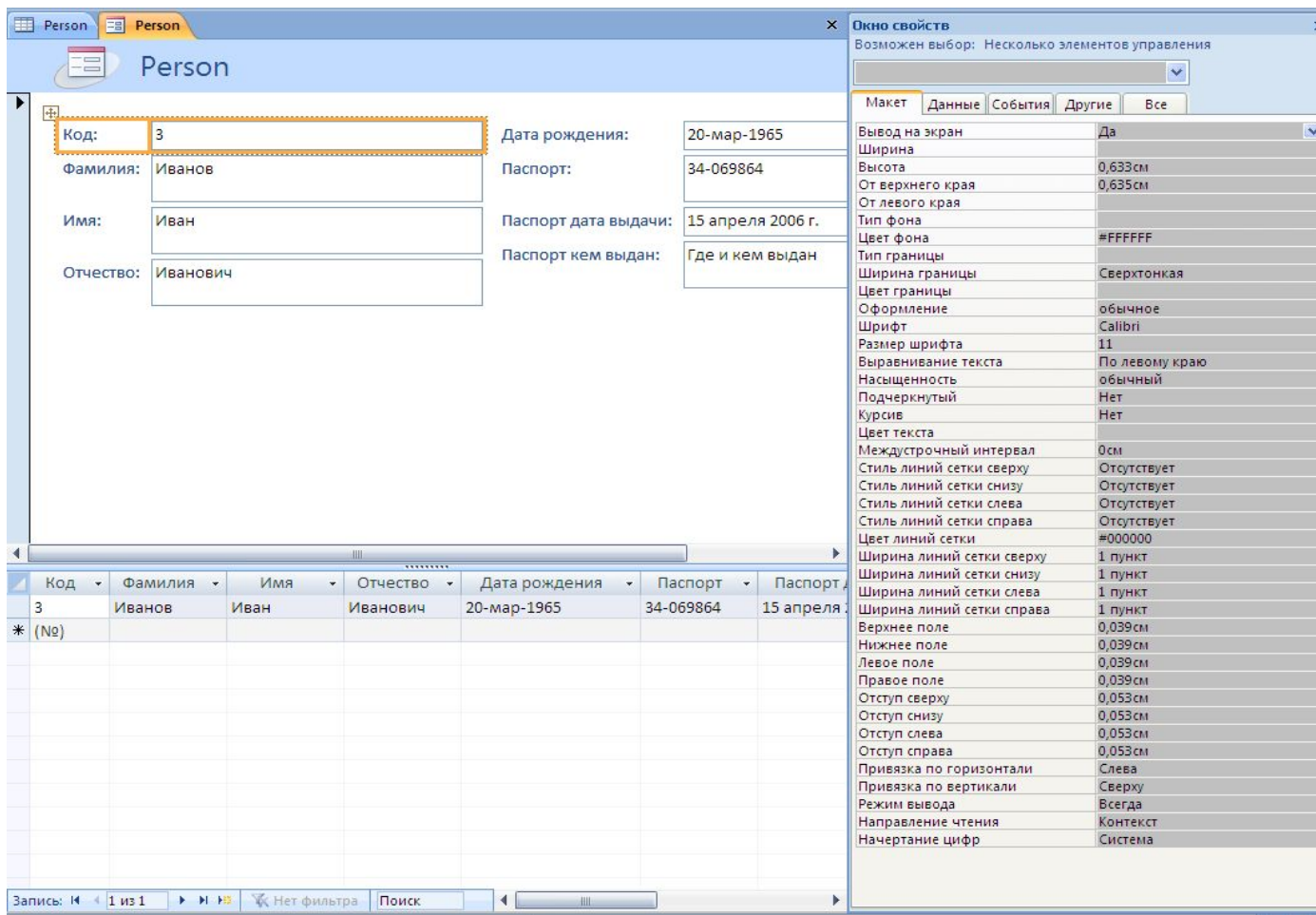


Рис.28. Изменение свойств полей карточки ввода данных.

Изменим параметр “Вывод на экран” с “Да” на “Нет”(Вкладка “Макет” панели “свойства”). Это поле типа “счетчик” и изменять нельзя – оно формируется автоматически. Соответственно, это поле не имеет смысла показывать на экран. Кроме того имеет смысл изменить название карточки. Пусть вместо Person будет написано “Личная карточка” Изменим параметр “подпись” для элемента Auto\_Title0 с “Person” на “Личная карточка” . Кроме того, выделим все элементы правого столбца и мышкой перетащим их вниз на уровень первого (показанного на экране) левого столбца. То, что получилось – на рис. 29.

The screenshot shows a software window titled 'Person' with a sub-tab 'Личная карточка'. The form contains the following fields:

Фамилия:	Иванов	Дата рождения:	20-мар-1965
Имя:	Иван	Паспорт:	34-069864
Отчество:	Иванович	Паспорт дата выдачи:	15 апреля 2006 г.
		Паспорт кем выдан:	123 о/м г. Москвы

Below the form is a table with the following data:

Код	Фамилия	Имя	Отчество	Дата рождения	Паспорт	Паспорт дата выдачи	Паспорт кем выдан
3	Иванов	Иван	Иванович	20-мар-1965	34-069864	15 апреля 2006 г.	123 о/м г. Москвы
* (№)							

At the bottom of the window, there is a status bar with the text 'Записи: 1 из 1' and a search field.

Отметим, что это карточка предназначена для ввода данных, поэтому вы сразу не увидите точно такую картинку. На рис. 29 результат ввода с клавиатуры данной карточки.

Рис.29. Окончательный вариант карточки ввода данных в базу.

Итак, с таблицей сотрудников компании мы, в основном, завершили работу. Теперь приступим к созданию таблицы телефонов.

Какие данные должна содержать эта таблица? Очевидно, как минимум, сам номер телефона и его владельца. Желательно указать также тип телефона (стационарный или мобильный), а также является данный телефон личным или служебным. Итак, мы получили следующие поля (рис 30):

ТЕЛЕФОНЫ КОМПАНИИ
Фамилия
Имя
Отчество
Телефон
Тип телефона
Принадлежность телефона

Рис. 30 Структура таблицы телефонов компании – начальный вариант

Приведем пример заполнения данными такой таблицы. (Это не таблица, подготовленная в ACCESS. Это просто исходные данные “на бумаге”)

ТЕЛЕФОНЫ КОМПАНИИ					
Фамилия	Имя	Отчество	Телефон	Тип тел.	Принадл.
Иванов	Иван	Иванович	+7(495)1234567	Стац.	личный
Иванов	Иван	Иванович	+7(905)7654321	Моб.	личный
Иванов	Иван	Иванович	+7(499)2345678	Служ.	компания
Иванов	Иван	Иванович	+7(903)5678901	Моб.	компания
Петров	Петр	Петрович	+7(903)2325465	Моб.	компания
Яблокова	Ольга	Владимиро вна	+7(495)5439612	Стац.	личный
Явлинский	Игорь	Петрович			

Рис.31. Заполненная таблица телефонов компании.

Итак, мы видим, что за Ивановым зарегистрировано 4 различных телефона, у Петрова и Яблоковой – по одному, а у Явлинского – вообще ни одного.

Возникает вопрос – зачем в таблице многократно повторять фамилию имя и отчество? (При условии наличия разработанной нами ранее таблицы персонала компании “Person”) Кроме того, предположим, что в компании работают 2 сотрудника с фамилией “Петров Петр Петрович” (полные тезки). Как мы будем различать их телефоны ?

Вспомним, что в структуре таблицы Person имеется поле “Код”, которое является уникальным. Поэтому мы можем использовать это поле для связи с таблицей Person, вместо комбинации полей “Фамилия”, “Имя” и “Отчество”. Так мы сразу избавляемся от этих проблем.

Итак, в соответствии со сказанным, структура нашей таблицы телефонов компании приобретает следующий вид:

ТЕЛЕФОНЫ КОМПАНИИ
Код сотрудника
Телефон
Тип телефона
Принадлежность телефона

Рис. 32. Структура таблицы телефонов компании

Создадим эту таблицу в ACCESS. Но, в отличии от таблицы Person, (которую мы создавали путем ввода образцов данных), воспользуемся другим инструментом ACCESS– “Конструктором” таблиц. Откроем ленту “Создание” и нажмем на кнопку “Конструктор таблиц”

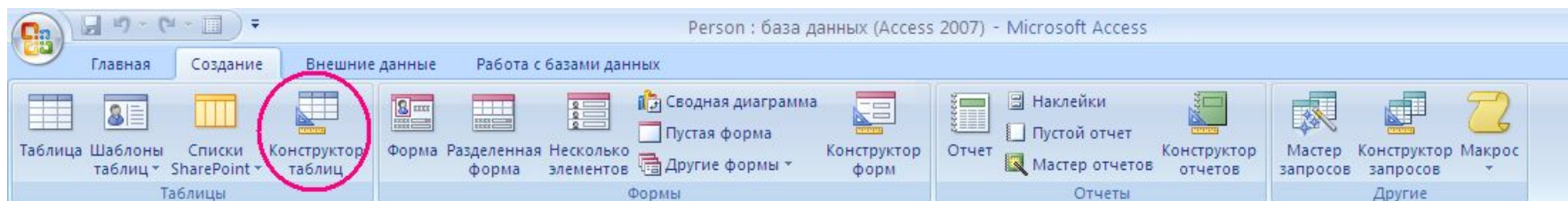


Рис. 33 Вызов “Конструктора таблиц”.

Автоматически открывается новая таблица в режиме конструктора (рис. 34)

The image shows the 'Table Design' view in Microsoft Access. The table is named 'Таблица1'. It has three columns: 'Имя поля' (Field Name), 'Тип данных' (Data Type), and 'Описание' (Description). The 'Имя поля' column is highlighted in orange. The table is currently empty.

Имя поля	Тип данных	Описание
----------	------------	----------

Рис. 34. Создание таблицы в режиме конструктора

В качестве первого поля введем уникальный ключ для каждой строки создаваемой таблицы. Назовем его “Код телефона”. Тип данных – “Счетчик”. Введем эту информацию (рис. 35). Кроме того, мы решили, что данное поле будет ключевым. Это наше решение необходимо “сообщить” СУБД. Для этого откроем ленту “Конструктор” и нажмем на кнопку “Ключевое поле” (Красный кружок на рисунке). Обратим внимание, что в результате наших действий рядом с нашим полем появился маленький рисунок ключа (Синий кружок) Этим символом СУБД информирует нас о ключевых полях таблиц.

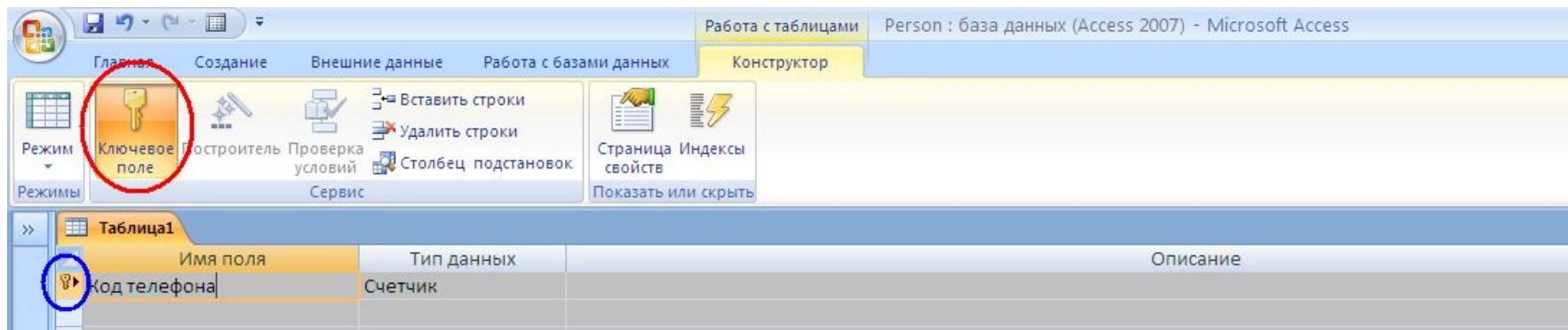


Рис. 35. Создание ключевого поля в режиме конструктора



Теперь посмотрим на нижнюю часть экрана. Здесь отображаются свойства текущего (выбранного в верхней части экрана) поля. Если мы раскроем свойство “размер поля”, то увидим, что у нас 2 возможности – “длинное целое” и “Код репликации”. Возможность “Код репликации” не следует использовать, если только мы не хотим создать запись “уникальную вообще” (а не только в пределах нашей таблицы). Это достаточно длинное (и, соответственно, достаточно медленно обрабатываемое) значение. Имеет смысл, только, если наша база данных будет “распределенной”, т.е. различные записи базы данных могут создаваться и храниться на различных компьютерах. Например, в различных филиалах компании (в разных городах)

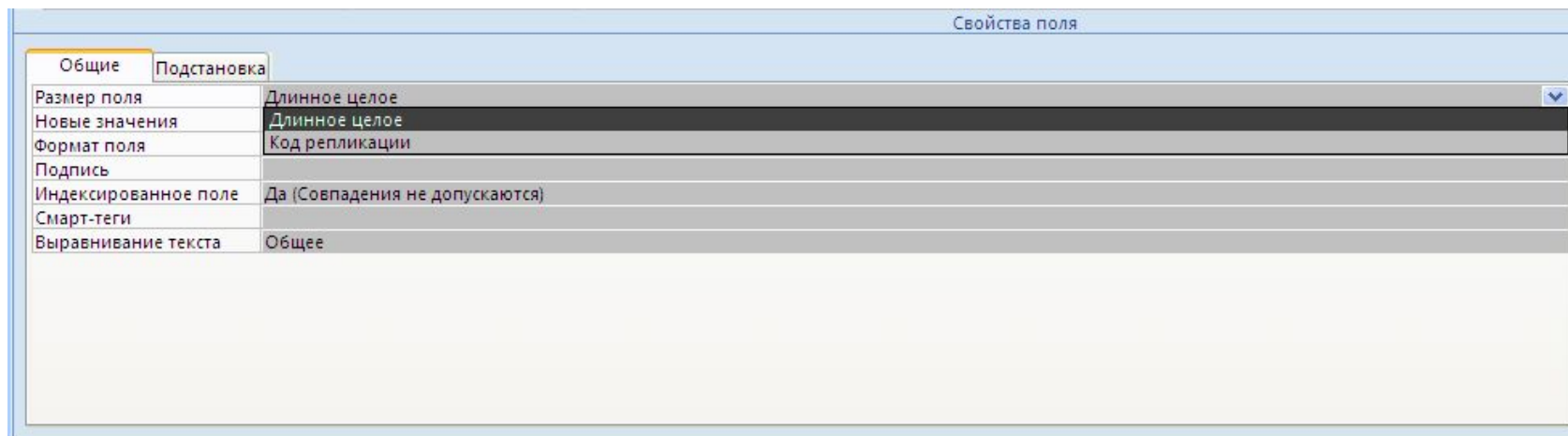


Рис. 36. Свойство “Размер поля” для поля типа “Счетчик”.

Обратим внимание еще на одно свойство поля типа “Счетчик”. Свойство называется “Новые значения”.

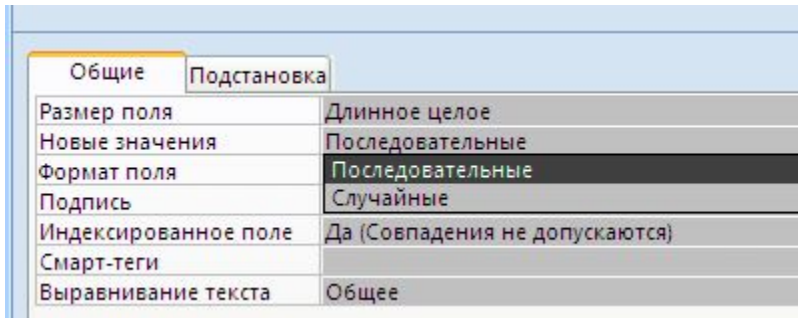


Рис. 37. Свойство “Новые значения” для поля типа “Счетчик”.

Как видно на рисунке, поле “Счетчик” может генерировать новые значения либо путем последовательного увеличения на 1 последнего значения, либо в режиме генератора случайных чисел.

Если у нас нет каких либо особых причин использовать режим генератора случайных чисел, всегда следует оставлять режим последовательных чисел, так как этот режим работает быстрее.

Примечание. Поскольку поле типа “Счетчик” автоматически генерирует последовательность автоматически возрастающих на 1 значений возникает соблазн использовать поле этого типа как “номер по порядку”. Этого делать нельзя. Причина – глобальный (в пределах таблицы) характер генерации значений. Например, если у вас будут три записи с автоматически сгенерированными кодами “1”, “2” и “3” и затем вы удалите

запись с кодом “2”, то запись с кодом “3” не станет автоматически записью с кодом “2”. Говоря другими словами перенумерации записей не произойдет – останется “дырка”. Новая (вновь создаваемая) запись также не получит код “2” (Несмотря на то что это значение “свободно”). Код будет “4”. Это справедливо, в том числе, и для последней записи тоже. Нельзя удалить последнюю запись “3” и тут же ввести ее заново, ожидая вновь значение “3”. Нет, даже в этом случае новым значением будет “4”.

Теперь перейдем к созданию поля “Код сотрудника” (рис. 38)

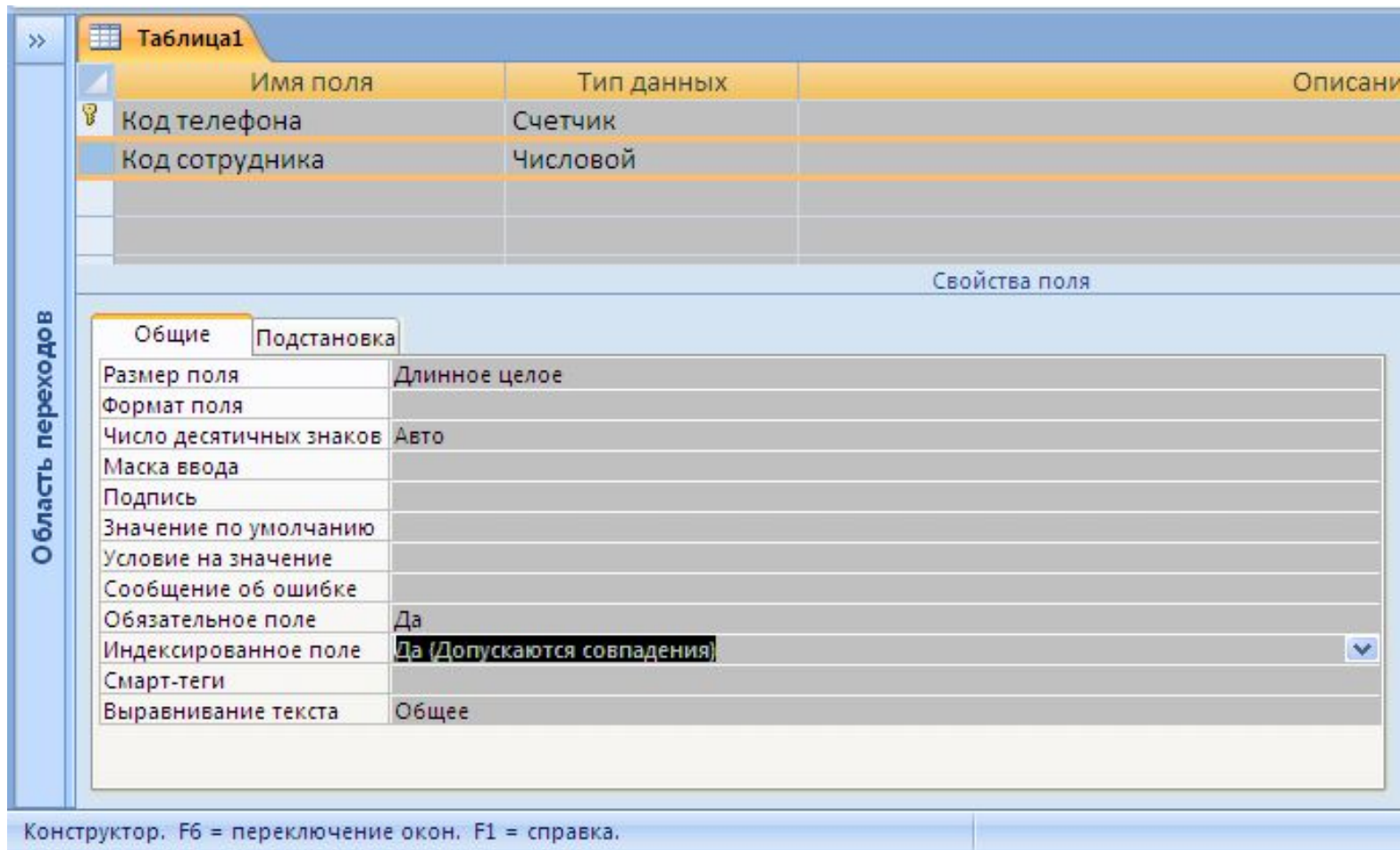


Рис. 38. Создание поля “Код сотрудника”.

Это поле тоже “Код”, но мы определили тип его данных не на как “Счетчик”, а как “Числовой” и размер поля как “Длинное целое”. Почему?

В данном случае мы не хотим, чтобы СУБД “генерировала” для нас новый код. Наша задача – сделать ссылку на запись в уже существующей таблице “Person”.

Говорят, что запись из одной таблицы ссылается на запись из другой таблицы в случае, если значения ключевых полей обеих таблиц совпадают. В качестве “ключевых” может использоваться как одно поле, так и набор из нескольких полей. Однако, для того чтобы ссылка работала, количество полей, их последовательность и типы данных должны совпадать.

Поле “Код” базы Person – это “счетчик”. Но как мы видели ранее, с технической точки зрения “Счетчик” – это “Длинное целое” число. Именно поэтому мы и выбрали этот тип данных для нашего поля “Код сотрудника”.

Обратим также внимание, что данное поле мы сделали “Обязательным” и “Индексированным – совпадения допускаются”.

Сделав это поле обязательным, мы тем самым определили правило о том, что у нас не может быть телефонов, не закрепленных ни за каким сотрудником. (Т.е. поле “Код сотрудника” обязано содержать значение одного из кодов из таблицы “Person”, что эквивалентно тому, что каждый телефон, занесенный в таблицу телефонов, должен быть закреплен за одним из сотрудников

Если же мы посчитаем, что это не так (т.е. компания может иметь телефоны, которые еще ни за кем не закреплены) то этот атрибут следует сбросить.

Определив поле как “индексированное” мы тем самым существенно убыстрили процесс поиска в таблице, но несколько замедлили время на вставку новых записей(так как индексы нужно обслуживать). Разрешив совпадения мы, тем самым, определили правило о том, что один и тот же сотрудник может иметь более одного телефона.

Теперь поговорим о поле “Телефон”. В принципе, телефон – это просто число. И никто не запрещает нам его хранить как число. Однако, по установившимся традициям и для удобства телефонный номер принято разделять на его составные части. Кроме того, довольно часто в номере телефона пишут специальные символы, указывающие на выполнение специальных действий или просто для удобства. Например, знак “+” в начале номера означает международный формат записи телефона, “\*” – необходимость перевода телефона в тональный режим, скобками выделяют код региона или код оператора связи.

Кроме того, математических операций над номерами телефонов обычно никто не производит. Поэтому будем хранить телефон как набор стандартных фрагментов и в текстовом виде.

Имя поля	Тип данных
Код телефона	Счетчик
Код сотрудника	Числовой
Тел код страны	Текстовый
Код региона	Текстовый
Телефон	Текстовый
Доп	Текстовый
Тип телефона	Текстовый
Принадлежность телефона	Текстовый

Общие	Подстановка
Размер поля	7
Формат поля	
Маска ввода	0000000
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Да
Пустые строки	Нет
Индексированное поле	Нет
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Рис. 39. Поля “Телефон”

Для каждого из полей “Тел код страны”, “Код региона” и “Телефон” определяем, что поле является обязательным и запрещаем пустые поля.

Для этого последовательно перебирая указанные поля устанавливаем для каждого из них значения свойства “Обязательное поле” в “да” и значение свойства “Пустые строки” в “нет”.

Дополнительно, для поля “Телефон” определим свойство “Маска ввода” в виде строки из 7 нулей. Это значит, что в это поле могут вводиться только цифры, причем ввод всех 7 цифр обязателен.

Все это, в совокупности, вынуждает пользователя вводить телефон полностью, причем во всех остальных полях, кроме поля “Телефон”, могут быть и иные символы (Например, скобки в поле “Код региона”) В поле “Телефон” – только цифры. В поле “Доп” может быть вообще ничего не введено.

Сохраним нашу работу и назовем таблицу “Phones”.

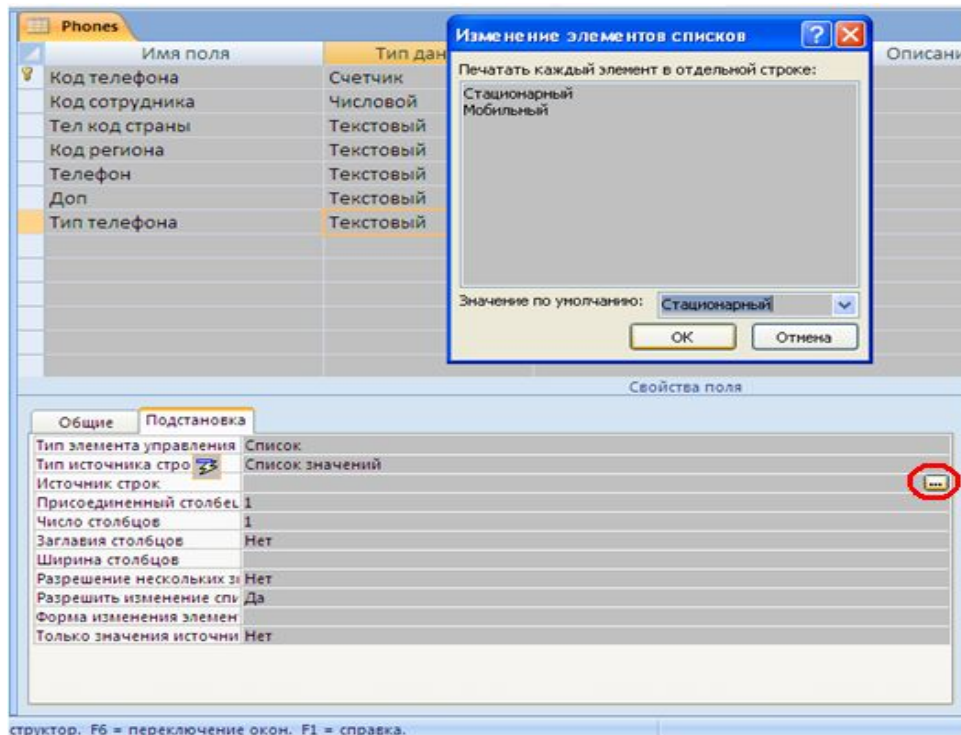


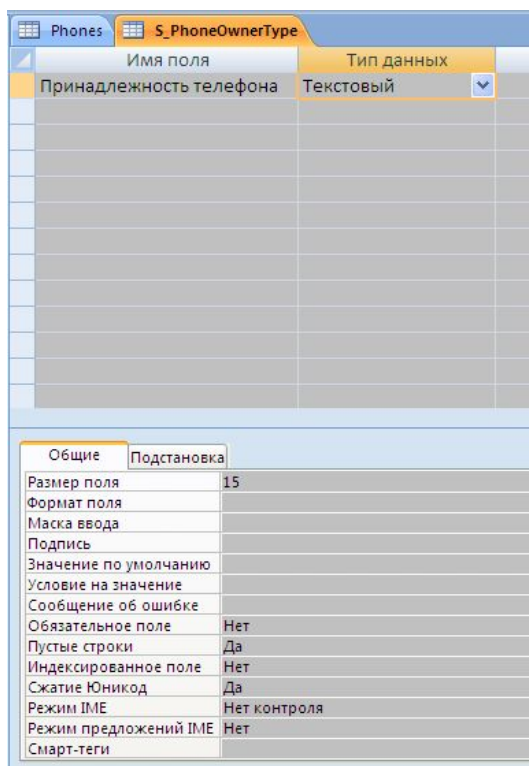
Рис. 40. Поле “Тип телефона”

Т.е. последовательно установим для свойства “тип элемента управления” – значение “Список” и для свойства “Тип источника строк” – значение “Список значений”. Перейдя на свойство “Источник строк” нажмем на кнопку “...” (помечена на рис 40 красным кружком). Это приведет к открытию дополнительного окна “Изменение элементов списков” в котором мы и введем значения “Стационарный” и “Мобильный”. По умолчанию поставим значение “Стационарный”.

Создадим теперь поле “Тип телефона”. Определим поле как текстовое длиной в 15 символов. Обратим теперь внимание на вкладку “Подстановка” в свойствах поля. Данная вкладка позволяет создавать список значений из которых пользователь сможет выбрать только одно значение. Выберем самый простой вариант подстановки – из заранее predeterminedенного здесь (при создании поля) списка текстовых значений.



Аналогичным образом будем создавать поле “Принадлежность телефона”. Однако, теперь список возможных значений определим не через прямой текстовый список, а через отдельно созданную таблицу. Для этого, не закрывая конструктор таблицы Phones параллельно запустим еще один “Конструктор создания таблиц” и в новой таблице определим единственное поле “Принадлежность телефона” как текстовое поле длиной 15 символов. (Рис. 41) Сохраним таблицу как S\_PhoneOwnerType



Откроем только что созданную вспомогательную таблицу в обычном (табличном) режиме и введем нужные значения подстановки (Рис. 42) Закроем вспомогательную таблицу и вернемся в создаваемую нами таблицу Phones.

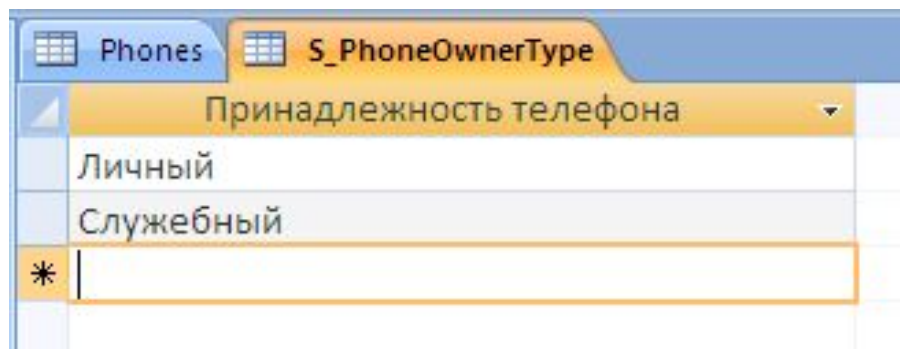


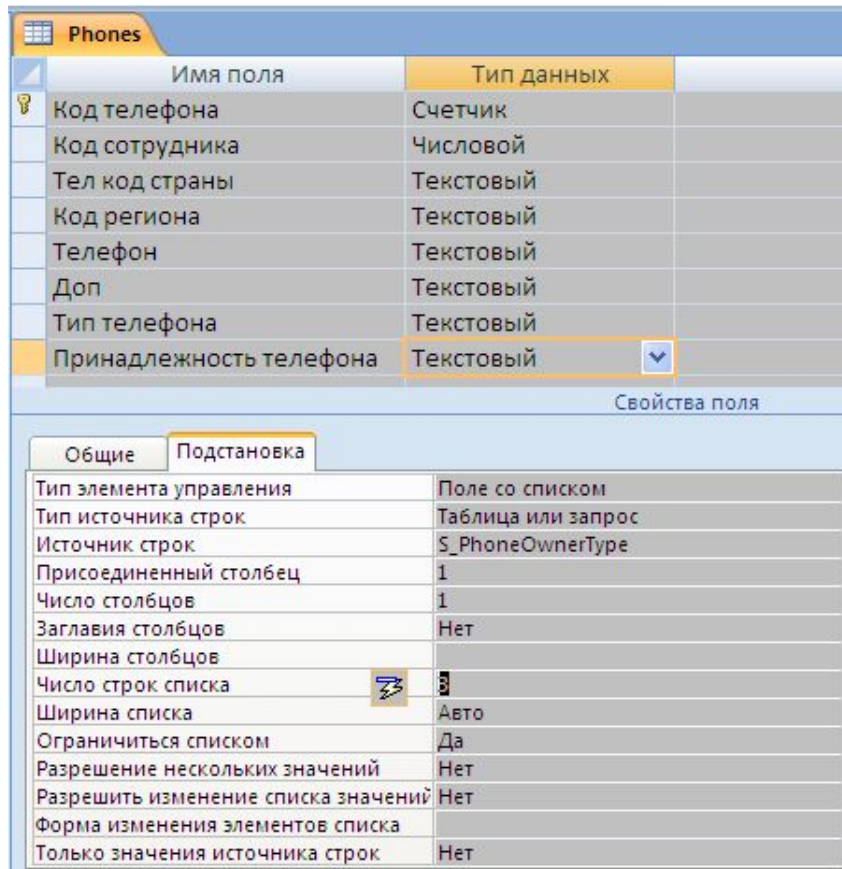
Рис. 42. Значения вспомогательной таблицы S\_PhoneOwnerType

Рис. 41. Вспомогательная таблица S\_PhoneOwnerType

Создадим поле “Принадлежность телефона” как текстовое с длиной в 15 символов. Снова перейдем на вкладку “Подстановка”. Определим на этой вкладке свойства следующим образом:

“Тип элемента управления” – “Поле со списком”.

“Тип источника строк” – “Таблица или запрос”.



Перейдем на свойство “Источник строк”. Нажмем на стрелку, которая появится в этом поле (справа). В раскрывшемся перечне таблиц выберем нашу вспомогательную таблицу S\_PhoneOwnerType.

Все прочие атрибуты поля определим как показано на рис. 43, тем самым запретив прямой ввод информации в данное поле. (Возможен только выбор предварительно определенных в S\_PhoneOwnerType значений).

Рис. 43. Определение поля принадлежность телефона

На этом формирование таблицы “Phones” завершено. Сохраним работу. Теперь определим связи между уже созданными таблицами. Для этого на ленте “Работа с базами данных” щелкнем на кнопке “Схема данных”

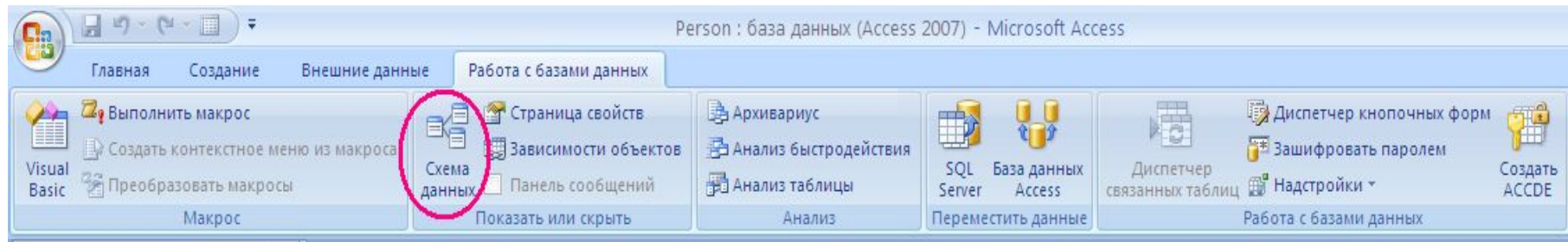
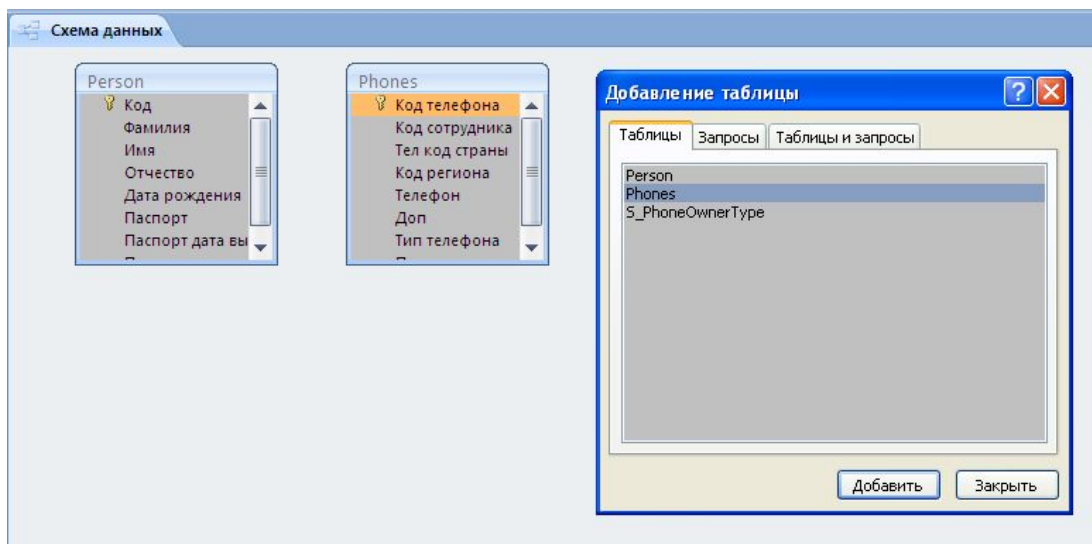


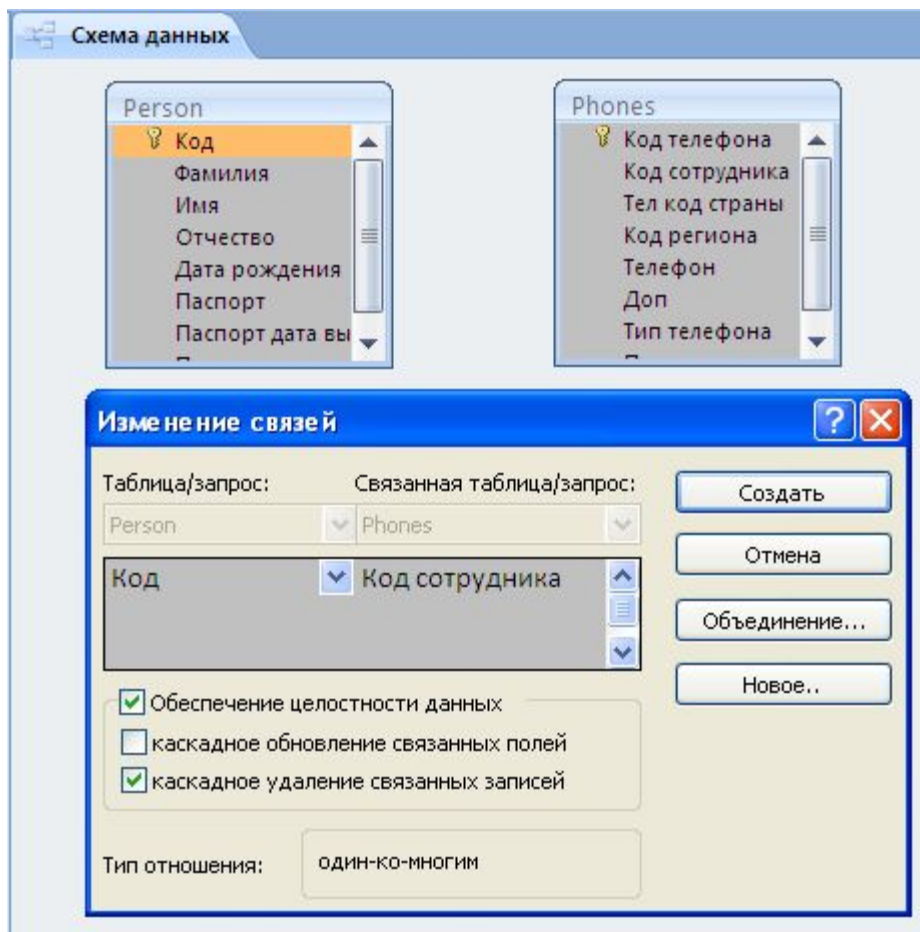
Рис. 44. Кнопка открытия страницы со схемой данных



В появившемся окне “Добавление таблицы” последовательно выберем таблицы “Person” и “Phones”, нажимая после выбора каждой таблицы кнопку “Добавить”. По окончании – кнопку “Закреть”

Рис. 45. Вставка таблиц в схему данных

В базе “Person” выберем поле “Код” левой кнопкой мыши. Затем, не отпуская кнопку мыши, “перетащим” это поле на поле “Код сотрудника” таблицы “Phones” и здесь отпустим кнопку мыши. Появится диалог изменения связей (рис.46)



Убедимся, что мы создаем связь между необходимыми нам полями Person.[Код] и Phones.[Код сотрудника]. Выберем опции “Обеспечение целостности данных” и “каскадное удаление связанных записей”. Тем самым мы разрешаем СУБД контролировать целостность данной связи и разрешаем при удалении сотрудника из списка сотрудников автоматически удалить всю информацию о связанных с ним телефонах.

Рис. 46. Изменение связей

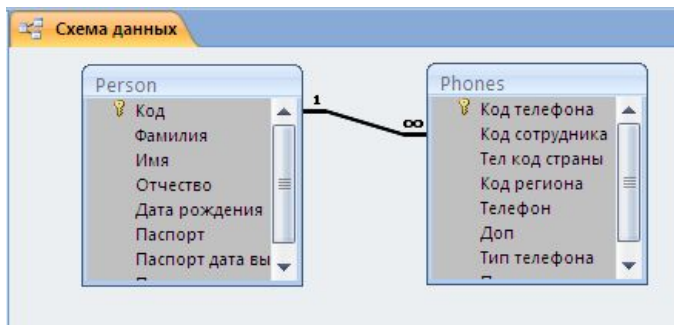


Рис. 47. Отображение связи на схеме данных

Имя поля	Тип данных
Код телефона	Счетчик
Код сотрудника	Числовой
Тел код страны	Текстовый
Код региона	Текстовый
Телефон	Текстовый
Доп	Текстовый
Тип телефона	Текстовый
Принадлежность телефона	Текстовый

Свойства поля	
Общие	Подстановка
Тип элемента управления	Список
Тип источника строк	Таблица или запрос
Источник строк	Person
Присоединенный столбец	1
Число столбцов	4
Заглавия столбцов	Да
Ширина столбцов	0см;8см;6см;6см
Разрешение нескольких значений	Нет
Разрешить изменение списка	Нет
Форма изменения элементов	
Только значения источника	Нет

Рис. 48. Определение подстановки для поля “Код сотрудника”

Построенная нами связь схематически отобразилась на схеме данных (Рис 47)

В соответствии с построенной нами связью определим подстановочные данные для поля “Код сотрудника” из таблицы Phones (рис. 48) Обратите внимание - число столбцов -4 , т.е сам код, фамилия, имя и отчество, но заданием нулевой ширины для первого столбца (самого кода) мы его скрываем (0 см;8 см; 6 см;6 см) Остальные значения определяют ширину столбцов в сантиметрах. Вводить “см” не обязательно – ACCESS подставит это сокращение сам. Можно ввести 0;8;6;6 . Значение может быть дробным, например 4,5

Теперь воспользовавшись стандартной карточкой для ввода данных сгенерируем карточку для ввода телефонов сотрудников. Откорректировав автоматически сгенерированную форму получаем:

The screenshot shows a web browser window titled "Phone\_Form" with a page header "Телефоны сотрудников". The form contains the following fields:

Сотрудник:	<table border="1"><thead><tr><th>Фамилия</th><th>Имя</th><th>Отчество</th></tr></thead><tbody><tr><td>Иванов</td><td>Иван</td><td>Иванович</td></tr><tr><td>Петров</td><td>Петр</td><td>Петровия</td></tr></tbody></table>	Фамилия	Имя	Отчество	Иванов	Иван	Иванович	Петров	Петр	Петровия
Фамилия	Имя	Отчество								
Иванов	Иван	Иванович								
Петров	Петр	Петровия								
Тел код страны:	+7									
Код региона:	905									
Телефон:	7654321									
Доп:										
Тип телефона:	<table border="1"><tbody><tr><td>Стационарный</td></tr><tr><td>Мобильный</td></tr></tbody></table>	Стационарный	Мобильный							
Стационарный										
Мобильный										
Принадлежность телефона:	<table border="1"><tbody><tr><td>Личный</td></tr></tbody></table>	Личный								
Личный										

Рис. 49. Форма для ввода телефонов сотрудников

Обращаем внимание – это форма для ввода данных, а не для их показа. Т.е. данные либо выбираются (“Сотрудник”, “Тип телефона”, “Принадлежность телефона”), либо вводятся с клавиатуры оператором базы.

Теперь перейдем к созданию таблицы “Телефонные разговоры сотрудников” Эту таблицу будем создавать как и Phones в режиме конструктора.

Телефонные разговоры
Код телефона
Направление (исходящий или входящий)
Дата и время начала разговора
Продолжительность разговора
Стоимость разговора
Ответный телефон

Рис. 50. Структура таблицы “Телефонные разговоры”

Итак, вновь выбираем ленту “Создание” и вновь нажимаем на кнопку “Конструктор таблиц”. В пустой (новой) таблице создадим новое поле типа “Счетчик” и назовем его “Код записи”. (Рис. 51) Данное поле будет уникально идентифицировать каждый отдельно взятый телефонный разговор. Это поле – ключевое для данной таблицы.

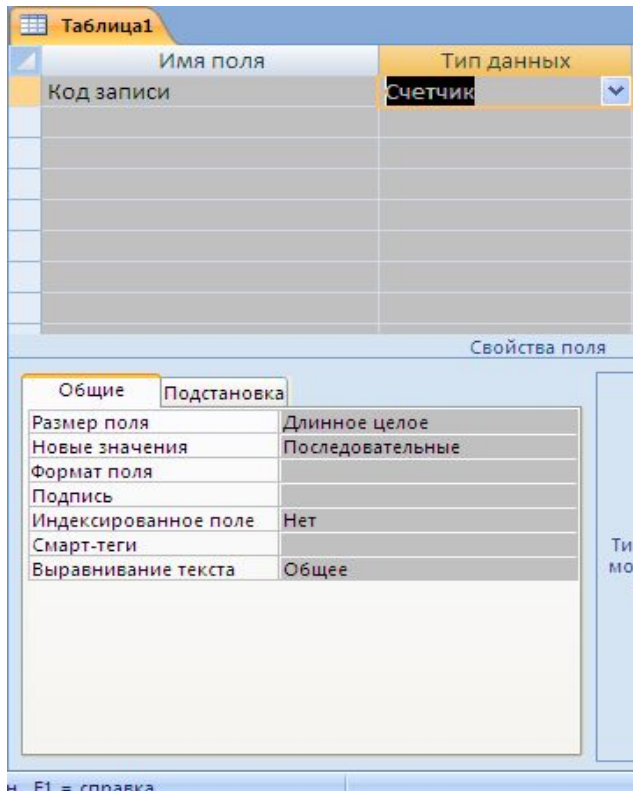


Рис. 51. Ключевое поле “Код записи”

Нажмем на кнопку “Сохранить” и сохраним создаваемую нами таблицу под именем “Phone\_Rings”.

Теперь создадим поле “Код телефона”. Данное поле будет внешним ключом для таблицы “Phones” (т.е. по этому полю мы будем строить “связь” между записью о телефоне в таблице “Phones” и теми разговорами в которых этот телефон будет участвовать, информация о которых будет храниться в создаваемой нами таблице “Phone\_Rings”). Поскольку тип данных для поля “Код телефона” в “Phones” – “Счетчик”, то, соответственно, тип данных для поля “Код телефона” в “Phone\_Rings” должен быть “числовым”, а “размер” – “длинное целое”. (Рис. 52)





Рис. 52. Внешний ключ “Код телефона”

Следующее поле – “Направление звонка”. Поле может иметь только два значения “Входящий” и “Исходящий”. Соответственно тип данных для данного поля – текстовый, размер – 15 символов. Это обычное подстановочное поле. Следовательно, на вкладке “Подстановка” определяем свойства: “Тип элемента управления” – “Поле со списком”, “Тип источника строк” – “Список значений”, “Источник строк” – “Исходящий”; “Входящий” (См. рис. 53)



Рис. 53. Поле “Направление”

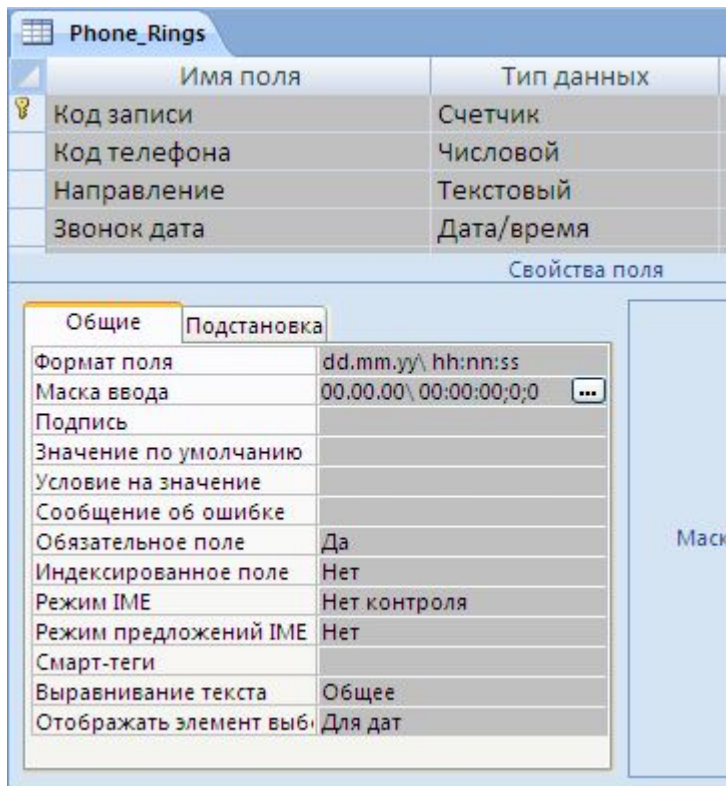


Рис. 54. Поле “Звонок дата”

Следующее поле – “Звонок дата”. Как понятно из наименования поля тип данных будет “Дата/Время”. Свойство “Формат поля” определим как “dd.mm.yy\ hh:nn:ss” , т.е две цифры числа, две- месяца, две – года. Далее часы, потом минуты и секунды, разделенные двоеточием.

Маска ввода - 00.00.00\ 00:00:00;0;0

Эта маска состоит из 3-х секций, разделенных точкой с запятой

Первая секция – это 00.00.00\ 00:00:00; Секция определяет, что для ввода даты числовой формат, вторая секция 0; обозначает, что литералы (точки) также будут передаваться на обработку (а не только высвечиваться на экране); Третья секция – заключительный 0 означает, что в качестве символов-заполнителей экранной маски будут использоваться нули.

Следующее поле – “Продолжительность звонка” Ничем не примечательное числовое поле. Если мы будем вести учет с точностью до минут – можем “обрезать” разрядность после запятой. Если хотим хранить точнее – можем оставить.

**Внимание** – мы определили это поле как “Числовое” (а не дата/время). Мы можем хранить здесь продолжительность звонков более точно, чем с точностью до минут. Но это не будут секунды! Это будут десятые (или даже сотые) доли минуты.

Время не может быть отражено как 1мин 99 секунд – в минуте только 60 секунд, но мы можем хранить значение 1,99 сотых долей минуты.

В нашем примере – ограничимся учетом с точностью до минут. Т.е. свойство “Число десятичных знаков” равно нулю. (Рис 55)

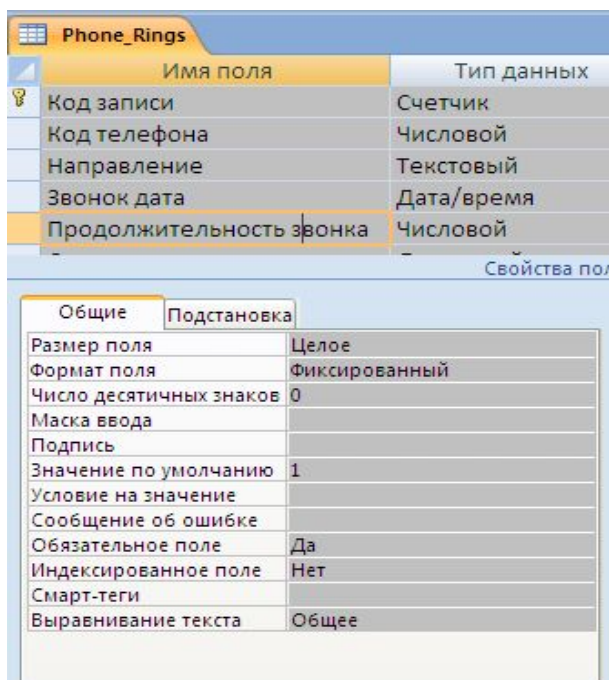


Рис. 55. Поле “Продолжительность звонка”

Следующее поле – “Стоимость разговора”. Это деньги – соответственно, выбор типа данных очевиден – “Денежный”. (Рис 56)  
 Ну и последнее поле – “Телефон ответный” – обычное текстовое поле размером в 50 символов. (Рис 57)

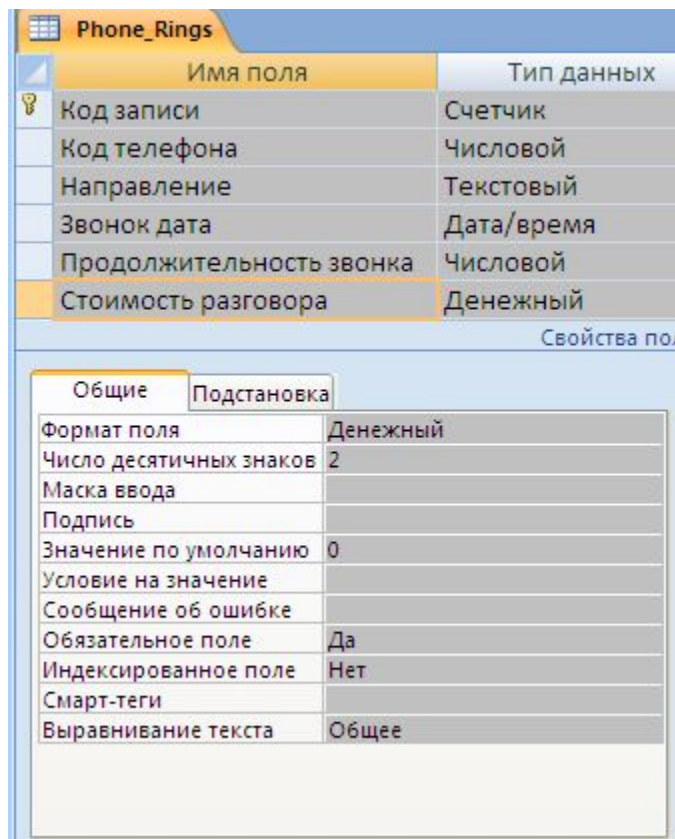


Рис. 56. Поле “Стоимость разговора”

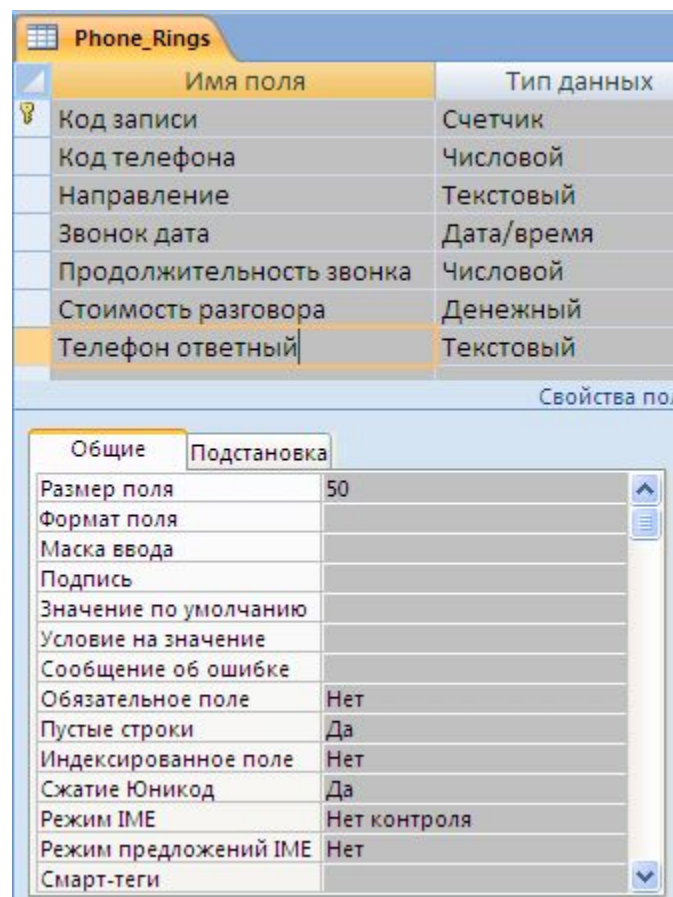


Рис. 57. Поле “Телефон ответный”

Откроем схему данных, добавим таблицу “Phone\_Rings” и “перетащив” “Код телефона” из таблицы “Phones” ( при нажатой левой кнопке мыши) на поле “Код телефона” в таблице “Phone\_Rings” и отпустив там мышью, мы построим новую связь. Определим ее свойства (рис.58). Нажав “Создать” получаем результат (рис. 59)

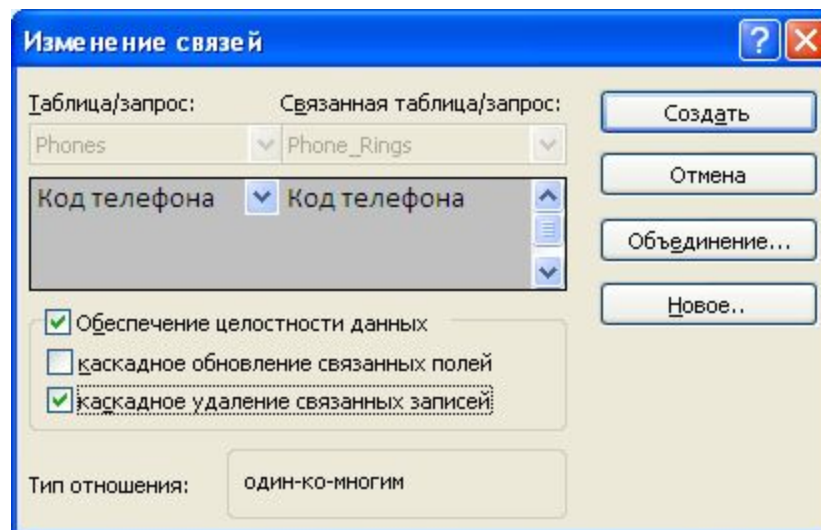


Рис. 58. Создание связи Phones – Phone\_Rings

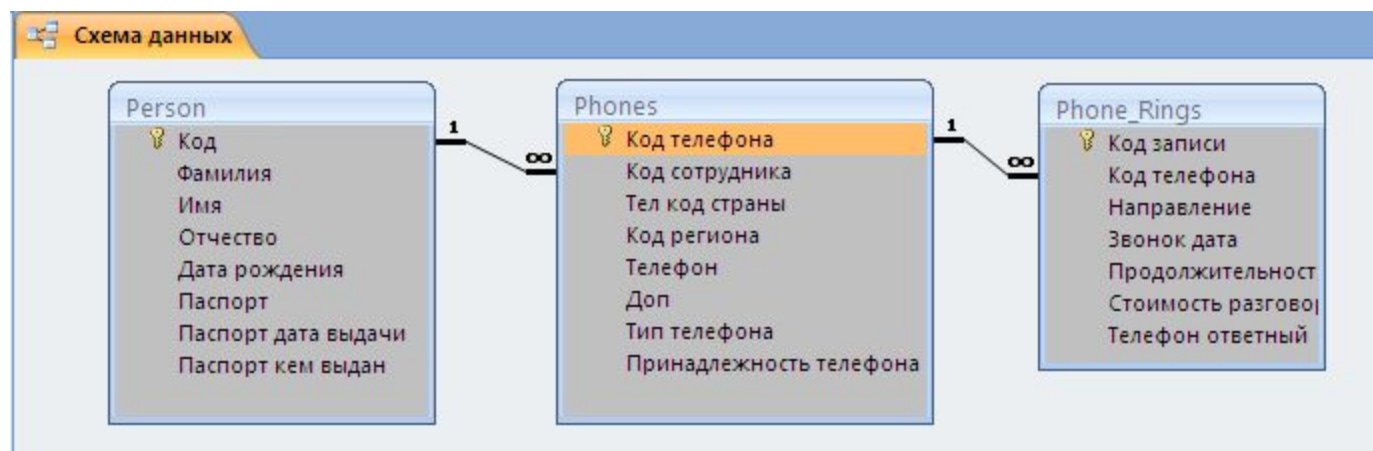
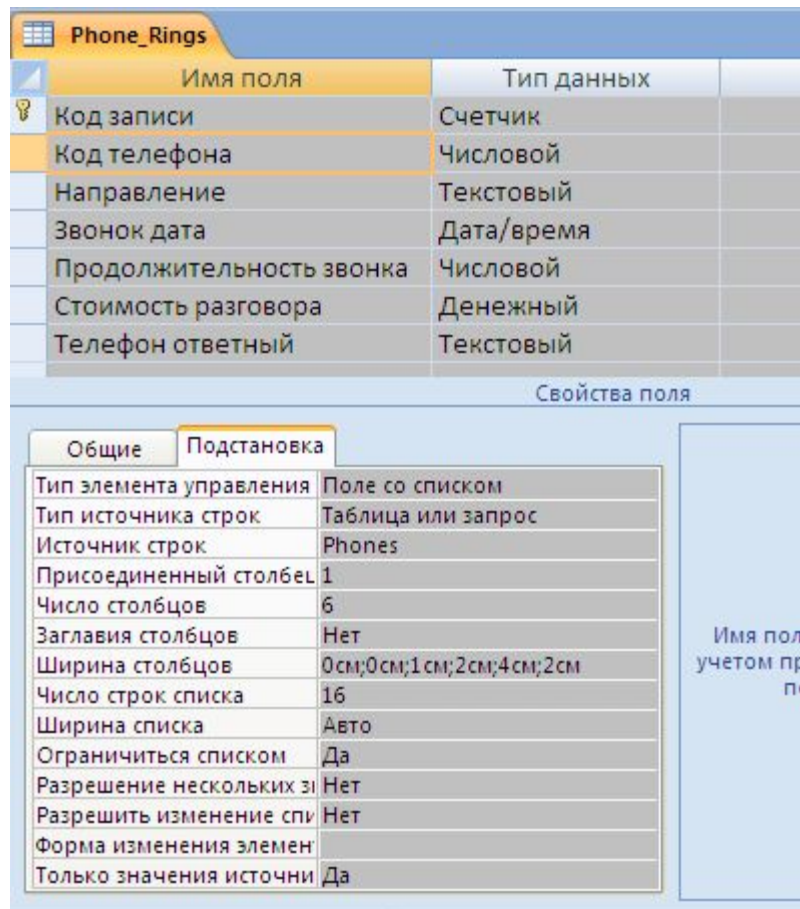


Рис. 59. Схема данных

Теперь, на основании построенной связи вернемся к описанию нашего внешнего ключа – поля “Код телефона” и определим для него “Подстановку” из таблицы “Phones”



Свойство “Тип элемента управления” определим как “Поле со списком”, свойство “Тип источника строк” – “Таблица или запрос”. В качестве источника строк определим таблицу “Phones”. Присоединенный столбец (т.е. значение которого сохраняется в Phone\_Rings – 1) Число столбцов таблицы Phones для показа - 6, из которых два первых (“Код телефона” и “Код сотрудника”) от пользователя скрываем, определив их ширину как 0. Отсюда свойство “Ширина столбцов” – “0см;0см;1см;2см;4см;2см”

Рис. 60. Определение “подстановки” для поля “Код телефона” таблицы “Phone\_Rings”.

Закроем таблицу “Phone\_Rings”. Выберем ее в качестве текущей в области переходов. На ленте “Создание” на вкладке “Формы” нажмем на кнопку “Несколько элементов”. Тем самым мы создадим третий стандартный тип формы для ввода данных

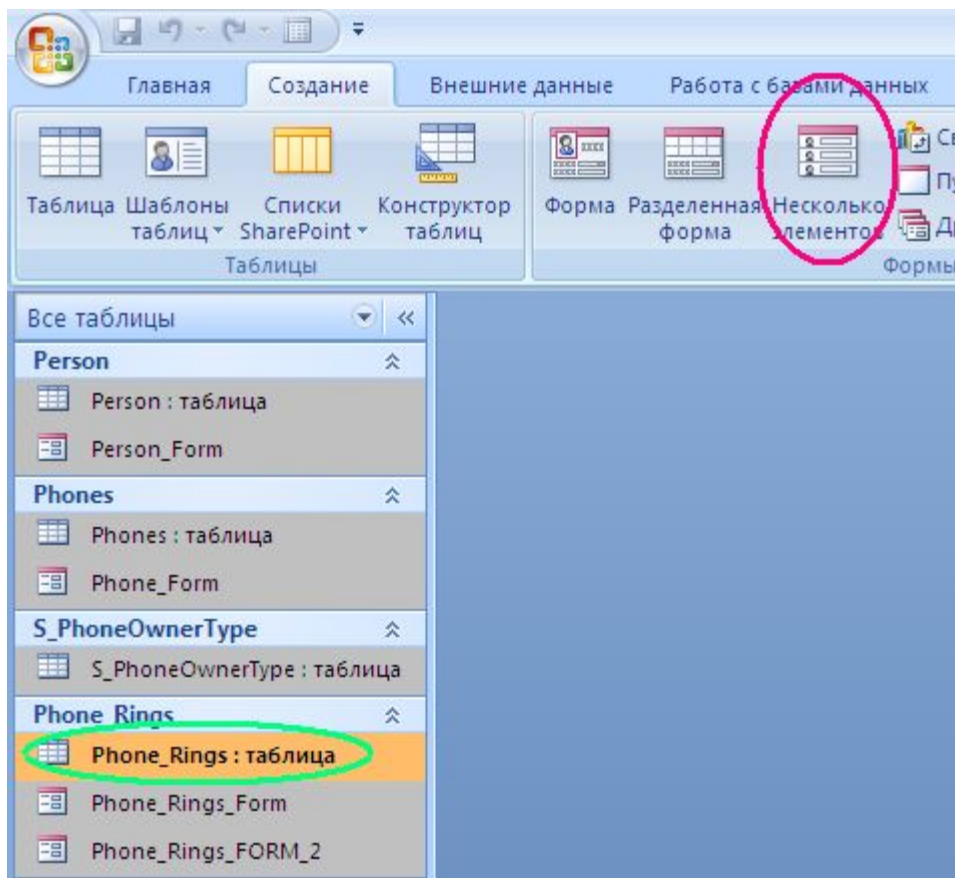
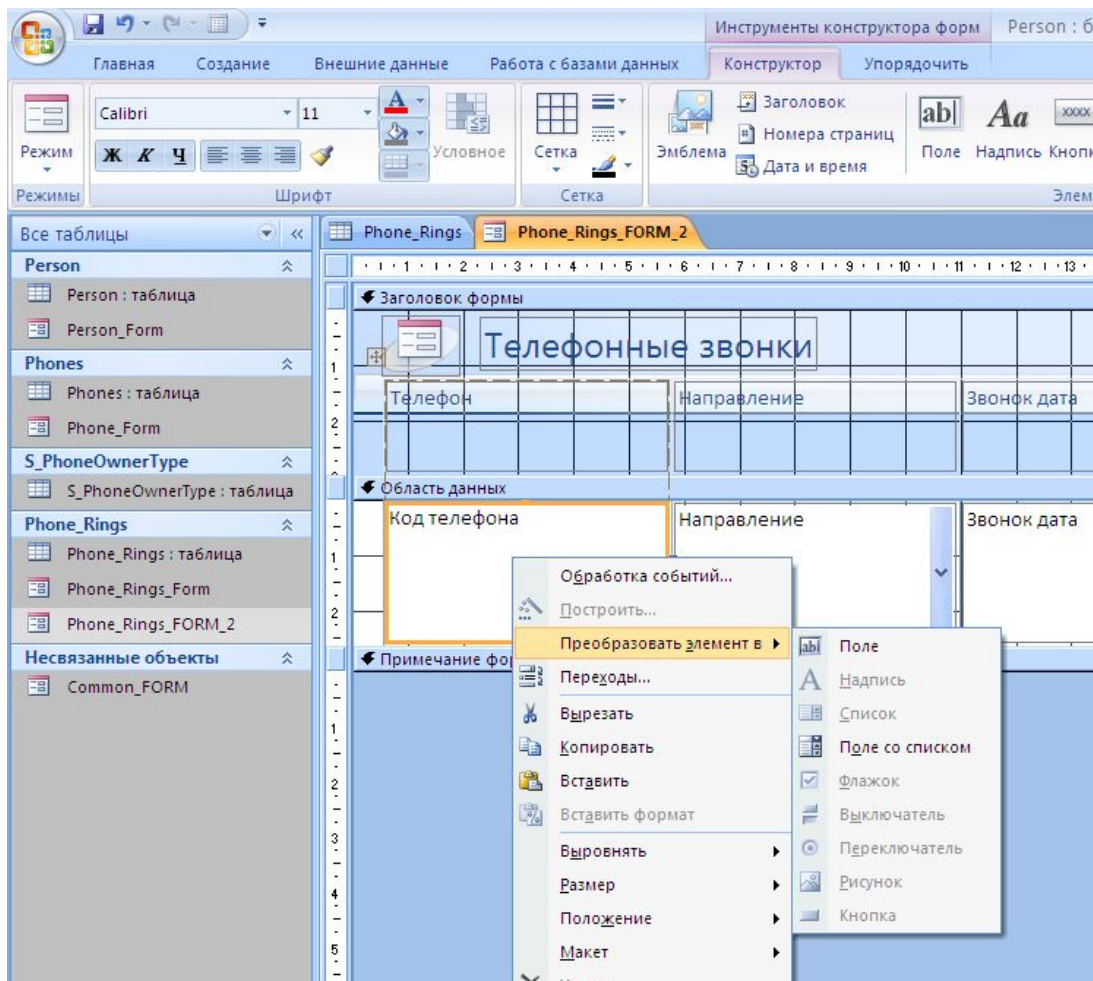


Рис. 61. Выбор формы ввода данных

Как и ранее слегка откорректируем автоматически построенную форму ввода. Во –первых, уберем оттуда автоматически генерируемое поле “Код записи “ (Просто выделив и удалив соответствующие элементы). Как всегда изменим стандартный заголовок .



Новый заголовок – “Телефонные звонки”. Кроме того, изменим заголовок столбца. Вместо “Код телефона” будет “Телефон”. Ну и для элемента тела столбца “Телефон” выполним преобразование отображаемого элемента на “Список”. Для этого щелкнем правой кнопкой мыши на элементе, выберем “Преобразовать элемент в” и далее выберем “Список”. Сохраним полученное под именем Phone\_Rings\_Form\_2.

Рис. 62. Преобразование элемента формы.



Phone\_Rings\_FORM\_2

### Телефонные звонки

Код телефона	Направление	Звонок дата	Продолжит. звонка	Стоимость разговора	Телефон ответный
+7 495 1234567 +7 905 7654321	Исходящий	05.04.10 14:30:15	2	3,00р.	+7(495) 5675241
+7 495 1234567 +7 905 7654321	Исходящий	12.04.10 10:02:10	1	1,00р.	+7(903)5431209
+7 495 1234567 +7 905 7654321	Входящий	07.04.10 12:37:40	2	0,00р.	+7(495)1234567
+7 495 1234567 +7 905 7654321	Исходящий		1	0,00р.	

Рис. 62. Работающая форма Phone\_Rings\_Form\_2.

Приступим к созданию простого запроса. Найдем все телефонные звонки, которые совершил сотрудник Иванов. Для этого на ленте “Создание” нажмем кнопку “Конструктор запросов” (рис. 63)

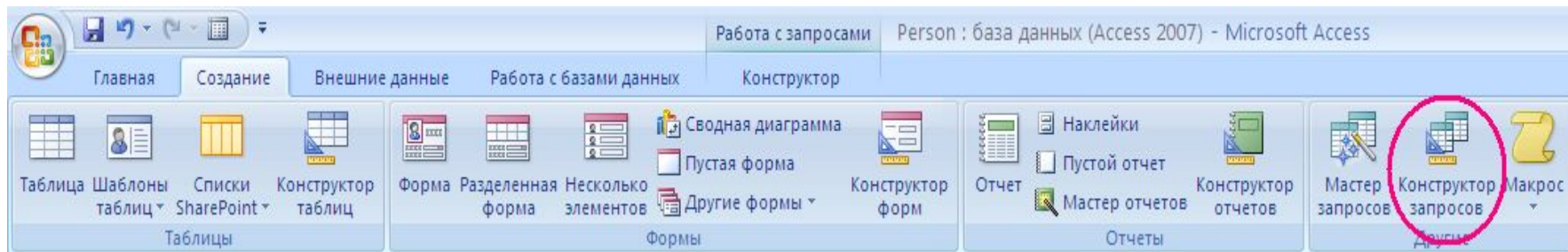
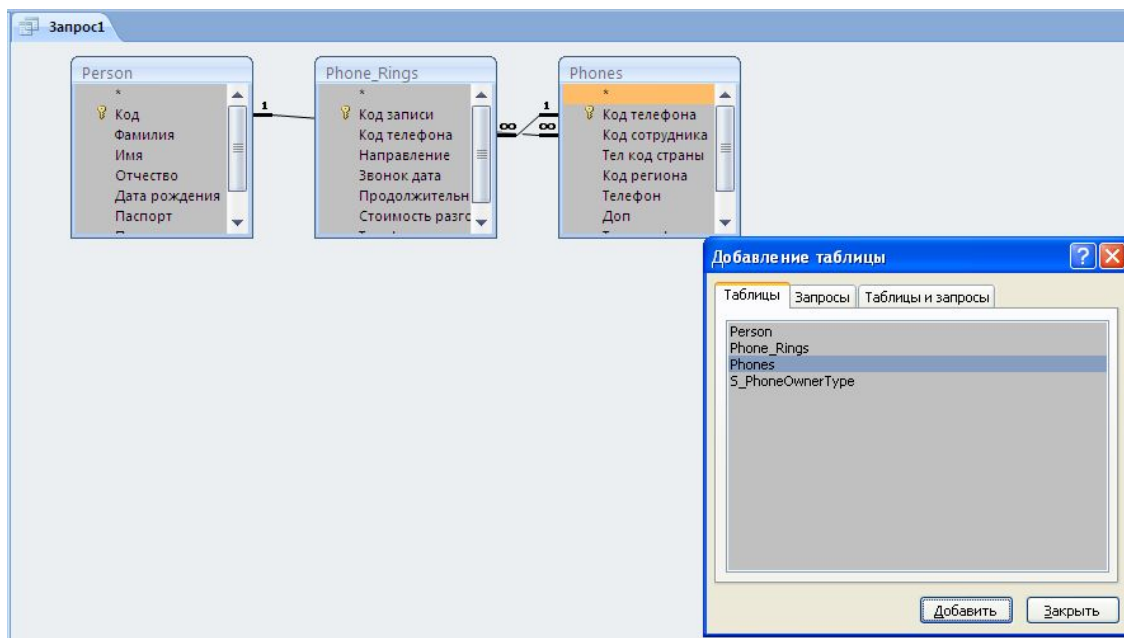


Рис. 63. Вызов “Конструктора запросов”



Откроется пустой запрос и диалоговое окно добавления таблиц. Последовательно : выбираем таблицу Person и нажимаем “Добавить”, затем Phone\_Rings и снова “Добавить” и, наконец , Phones - “Добавить” – “Закреть”

Рис. 64. Добавление таблиц в запрос

Расположим таблицы так, чтобы были видны все связи и все поля таблиц. Выбрав позицию поля (т.е. щелкнув мышкой на нужном пустом поле макета заголовка таблицы внизу экрана) сделаем двойной клик на требуемом поле в схеме запроса. Имя таблицы и самого поля будут отображены в выбранной нами позиции макета таблицы. Выберем таким образом все необходимые нам поля:

- Person.Фамилия
- Person.Имя
- Person.Отчество
- Phones.[Тел код страны]
- Phones.[Код региона]
- Phones.[Телефон]
- Phones.[Доп]
- Phones.[Тип телефона]
- Phones.[Принадлежность телефона]
- Phone\_Rings .[Направление]
- Phone\_Rings .[Звонок дата]
- Phone\_Rings .[Продолжительность звонка]
- Phone\_Rings .[Стоимость разговора]
- Phone\_Rings .[Телефон ответный]

Результат показан на рис 65.

The screenshot shows the Microsoft Access 2007 interface. The title bar reads 'Person : база данных (Access 2007) - Microsoft A...'. The ribbon is set to 'Конструктор' (Design) under the 'Работа с запросами' (Queries) group. The main area displays a query design grid for 'Запрос1' with three tables: 'Person', 'Phones', and 'Phone\_Rings'. The 'Person' table has fields: Код, Фамилия, Имя, Отчество, Дата рождения, Паспорт, Паспорт дата выдачи, Паспорт кем выдан. The 'Phones' table has fields: Код телефона, Код сотрудника, Тел код страны, Код региона, Телефон, Доп, Тип телефона, Принадлежность телефона. The 'Phone\_Rings' table has fields: Код записи, Код телефона, Направление, Звонок дата, Продолжительность звонка, Стоимость разговора, Телефон ответный. The design grid below shows the following configuration:

Поле:	Фамилия	Имя	Отчество	Тел код страны	Код региона	Телефон	Доп	Тип
Имя таблицы:	Person	Person	Person	Phones	Phones	Phones	Phones	Ph
Сортировка:								
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Условие отбора:								
или:								

Рис. 65. Определение запрашиваемых полей в запросе.

Теперь в свойство “условие отбора” для поля Person.Фамилия в печатаем “Иванов”. (Можно без кавычек – ACCESS автоматически вставит кавычки)

The screenshot displays the Microsoft Access Query Design view for a query named 'Запрос1'. It shows three tables: 'Person', 'Phones', and 'Phone\_Rings'. The 'Person' table is linked to 'Phones' with a 1-to-many relationship, and 'Phones' is linked to 'Phone\_Rings' with a 1-to-many relationship. The design grid below the tables is as follows:

Поле:	Фамилия	Имя	Отчество	Тел код страны	Код региона	Телефон	Доп	Тип
Имя таблицы:	Person	Person	Person	Phones	Phones	Phones	Phones	Ph
Сортировка:								
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Условие отбора или:	Иванов							

Рис. 66. Определение условия запроса

Теперь запустим запрос на выполнение. Для этого нажмем на кнопку “Выполнить” ленты “Конструктор”

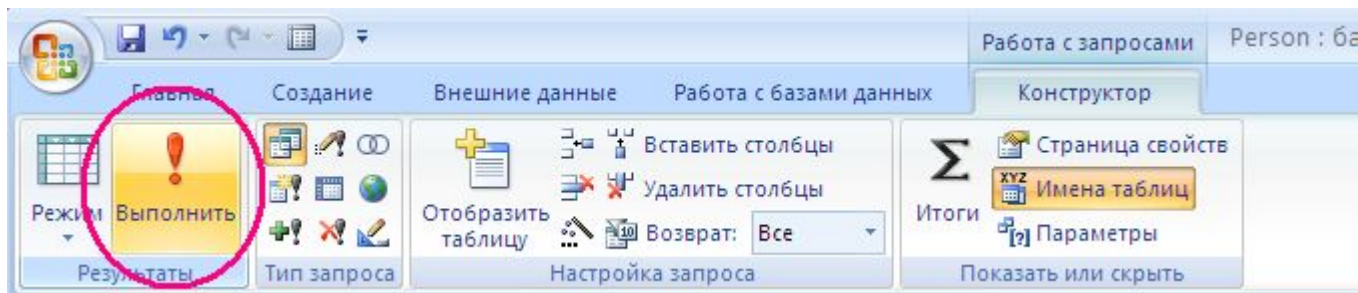


Рис. 67. Запуск запросы на исполнение – кнопка “Выполнить”

Результат выполнения запроса на рис. 68.

The image shows a screenshot of a query result table in Microsoft Access. The table has a title bar that says 'Запрос1' (Query1). The table has 10 columns: 'Фамилия' (Surname), 'Имя' (Name), 'Отчество' (Patronymic), 'Тел код стр' (Phone area code), 'Код регион' (Region code), 'Телефон' (Phone number), 'Доп' (Additional), 'Тип телефо' (Phone type), 'Принадлеж' (Ownership), and 'Направлен' (Direction). The first row is highlighted in orange and contains the following data: 'Иванов', 'Иван', 'Иванович', '+7', '495', '1234567', an empty cell, 'Стационарный' (Landline), 'Личный' (Personal), and 'Исходящий' (Outgoing). The second row contains: 'Иванов', 'Иван', 'Иванович', '+7', '905', '7654321', an empty cell, 'Мобильный' (Mobile), 'Личный' (Personal), and 'Исходящий' (Outgoing). The third row contains: 'Иванов', 'Иван', 'Иванович', '+7', '905', '7654321', an empty cell, 'Мобильный' (Mobile), 'Личный' (Personal), and 'Входящий' (Incoming). There is a small asterisk in the bottom left corner of the table area.

Рис. 68. Результат выполнения запроса

Сохраним запрос под именем “Звонки Иванова” и закроем запрос.

Теперь создадим отчет. Для этого на ленте “Создание” нажмем кнопку “Мастер отчетов”

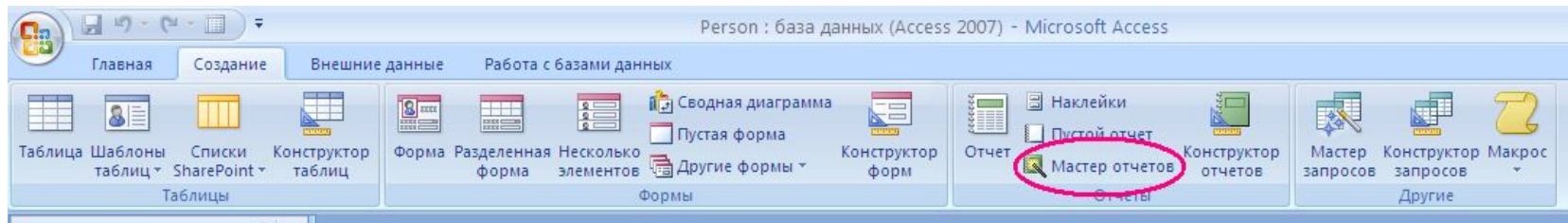
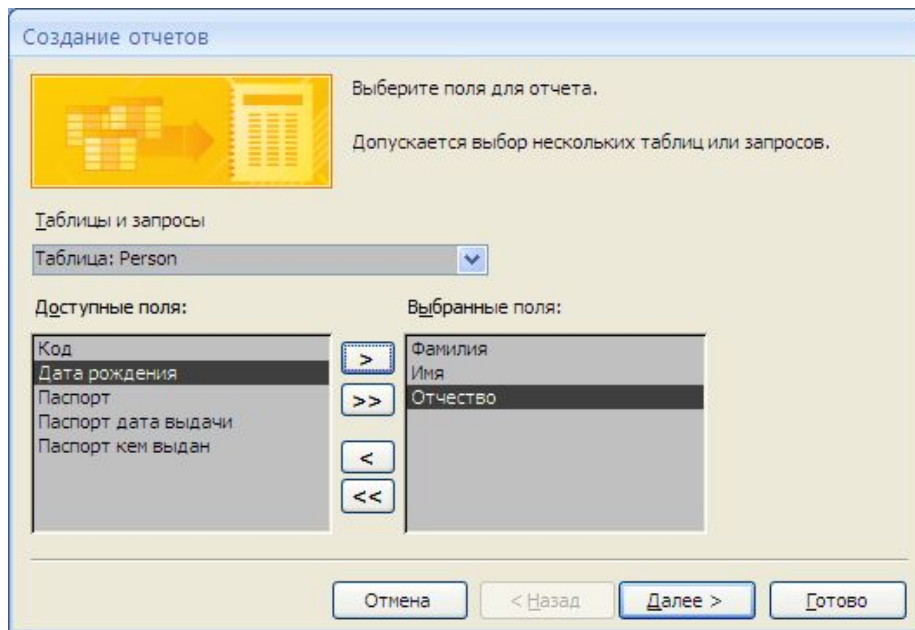


Рис. 69. Вызов мастера отчетов

Данный мастер автоматически проведет нас по всем этапам создания стандартного отчета.




Сначала в выборе “Таблицы и запросы” выберем таблицу Person, а затем, нажимая на кнопку  переместим поля “Фамилия”, “Имя” и “Отчество” из списка “Доступные поля” в список “Выбранные поля”. (рис 70.)

Рис. 70. Выбор полей из Person

Затем изменим выбор в “Таблицы и запросы” на таблицу “Phones” и добавим к выбранным полям “Код региона” и “Телефон” (рис 71)

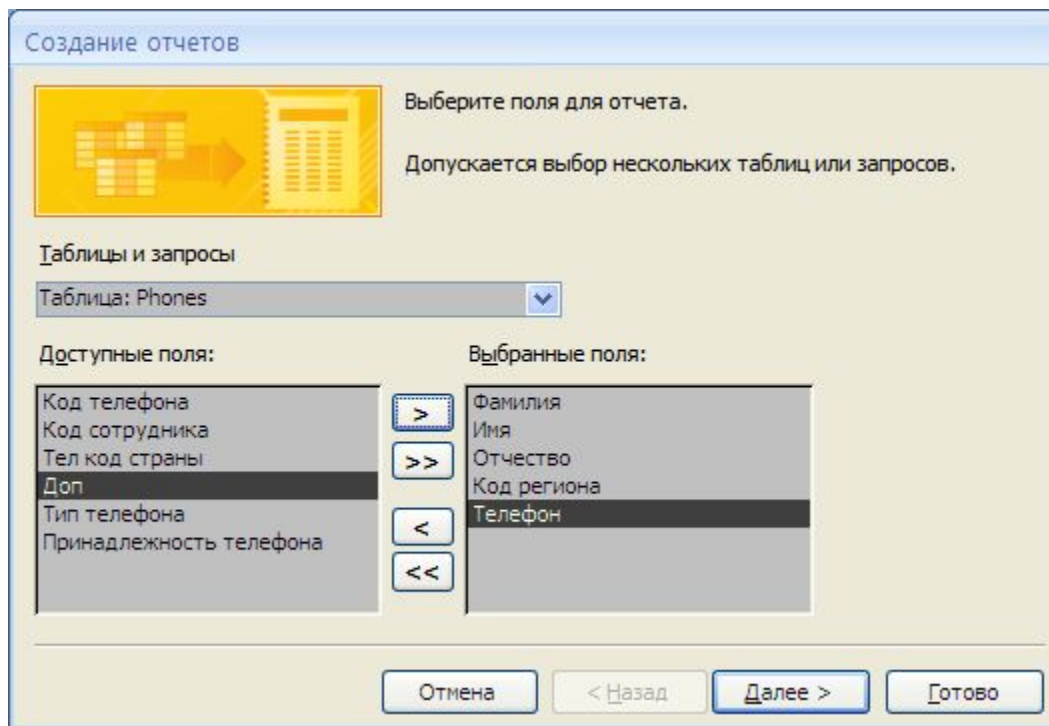
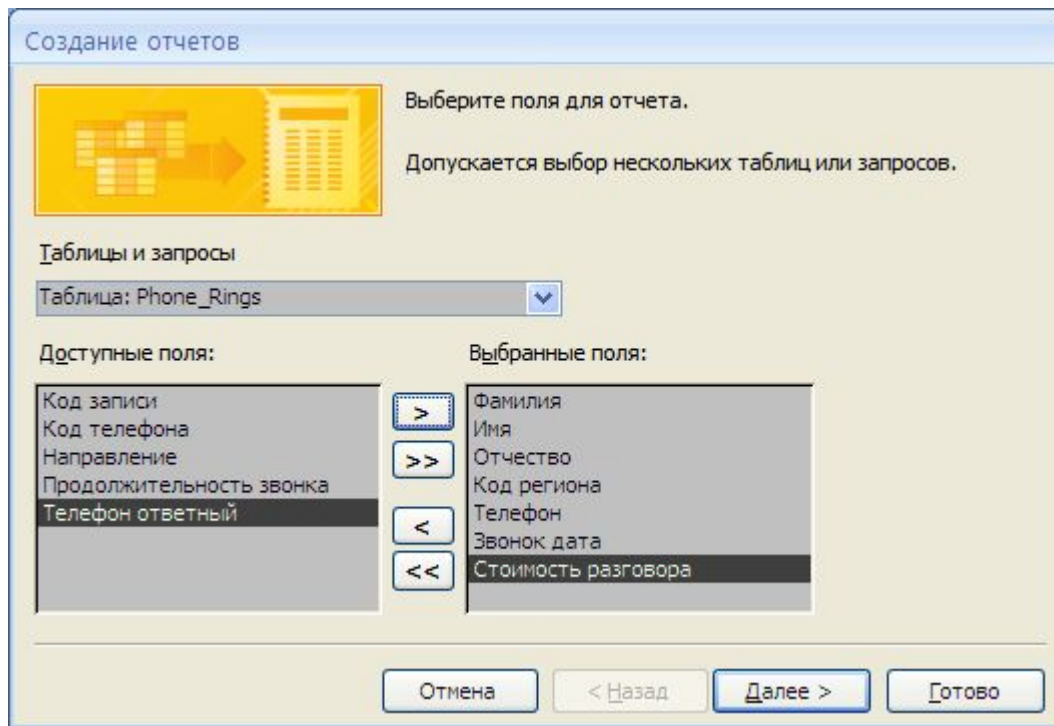


Рис. 71. Выбор полей из Phones.

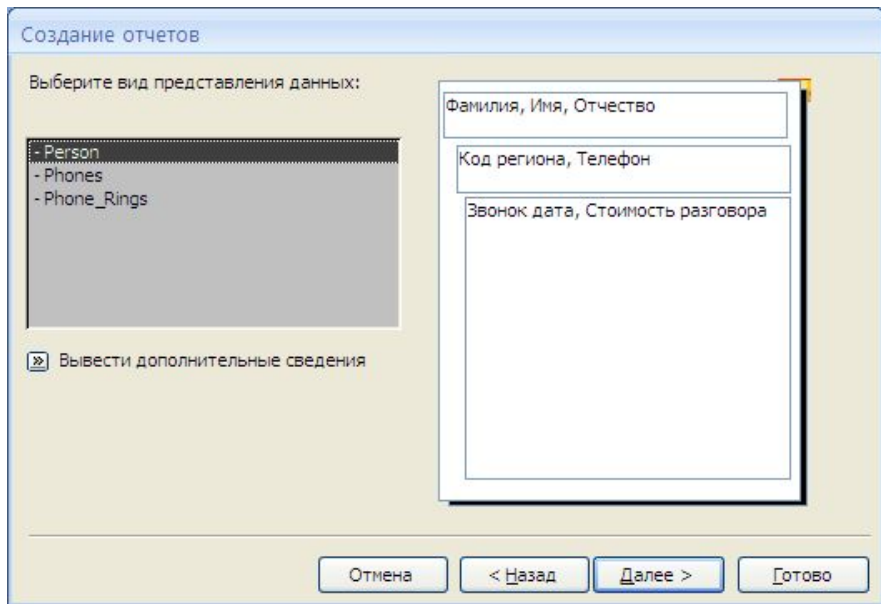


Снова сменим таблицу – на этот раз на Phone\_Rings и добавим поля “Звонок дата” и “Стоимость разговора”. (Рис 72)

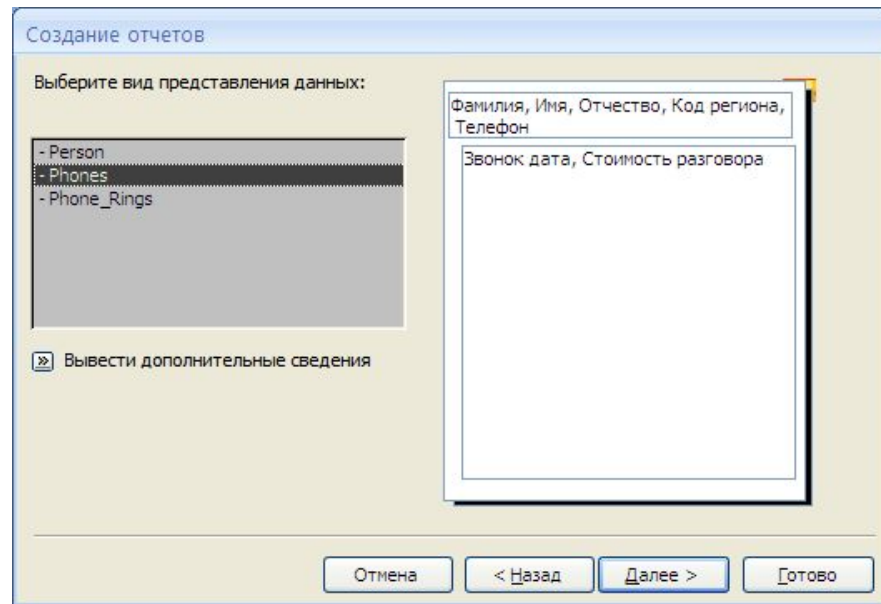


Теперь все необходимые поля выбраны. Нажмем кнопку “Далее”.

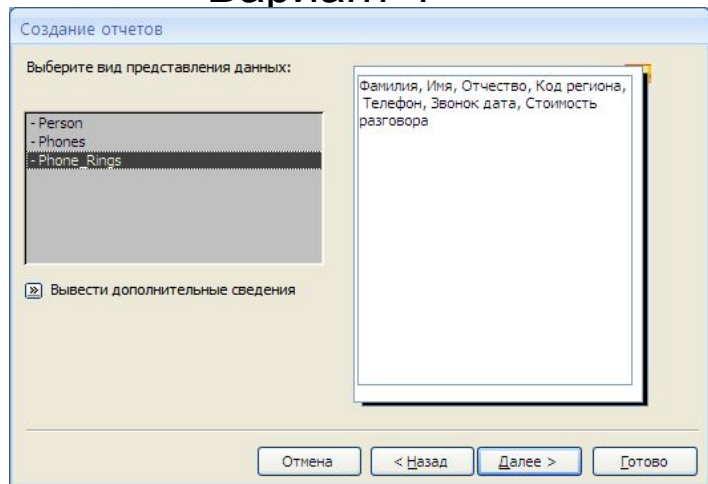
Рис. 72. Выбор полей из Phone\_Rings



Вариант 1



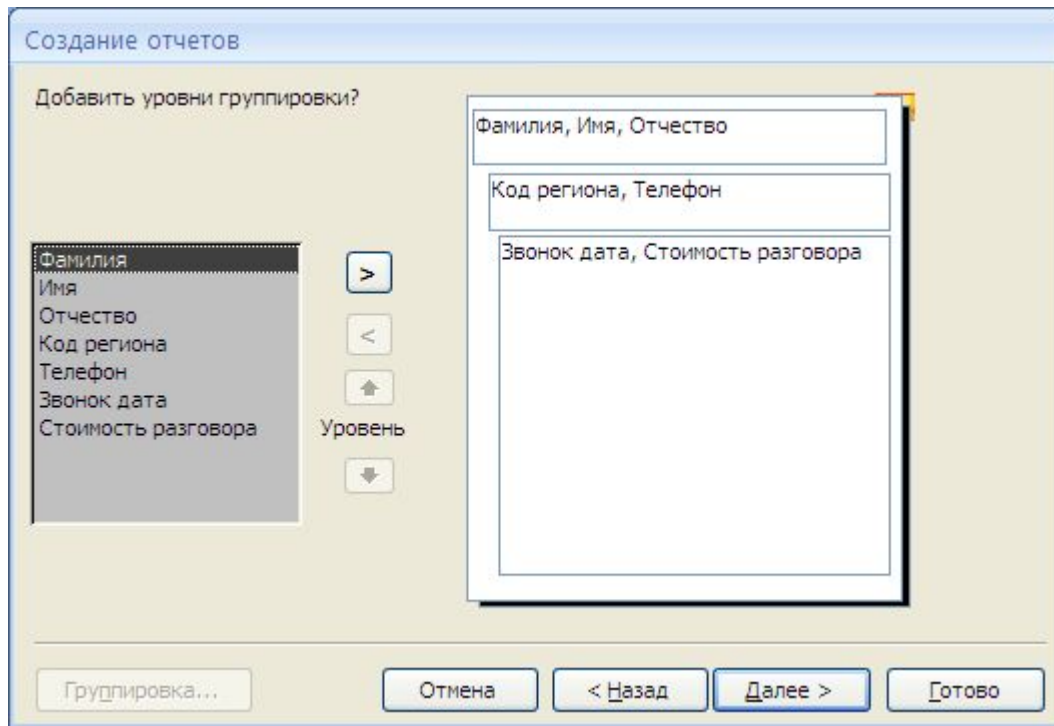
Вариант 2



Вариант 3

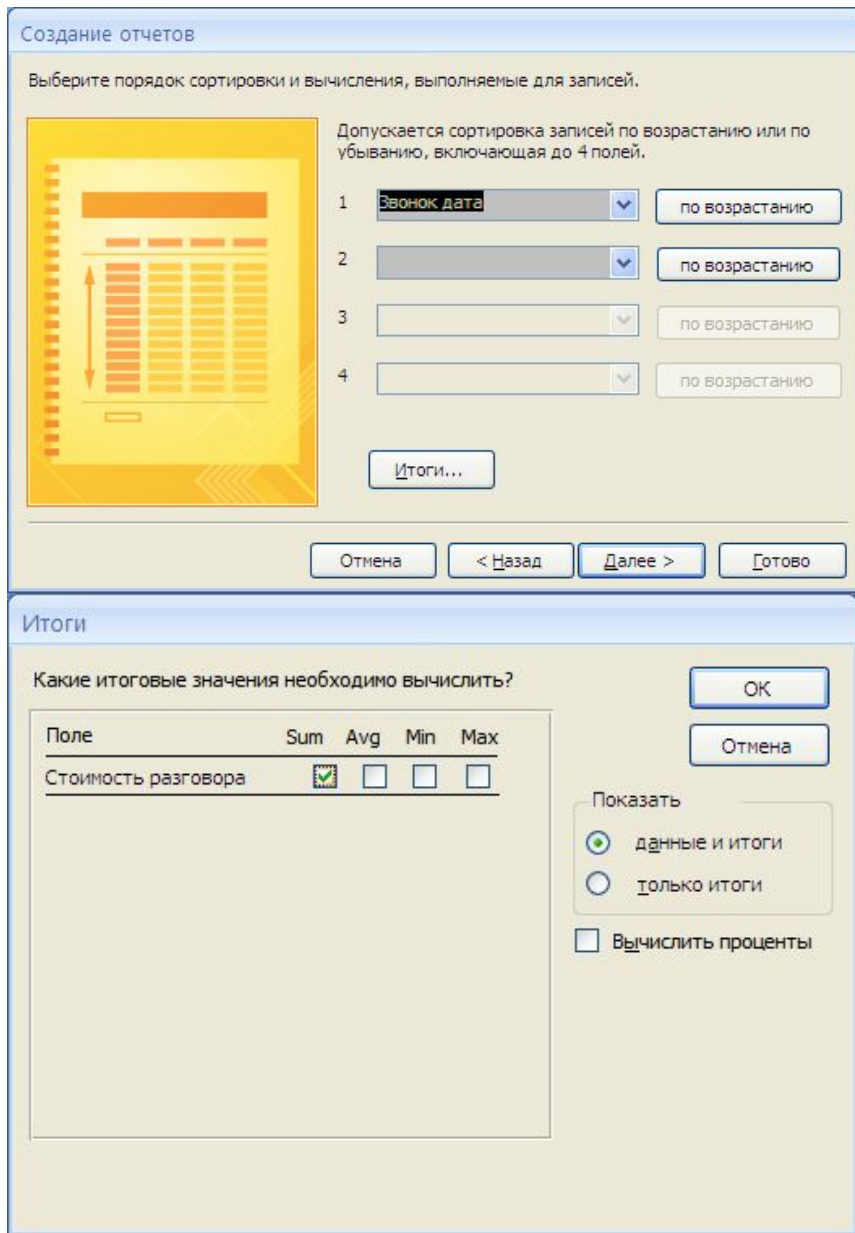
У нас есть выбор одного из трех вариантов группировки данных в отчете. В варианте 1 сначала по ФИО, затем по коду региона и телефону. В варианте 2 сразу по ФИО, коду региона и телефону. одновременно Вариант 3 – это фактически просто список всех звонков без группировки. Выберем вариант 1. и нажмем кнопку “Далее”

Рис. 73. Выбор варианта представления данных



На данном этапе можно добавить или уменьшить количество группировок в отчете, их распределение по уровням. Не будем ничего менять – просто нажмем кнопку “Далее”.

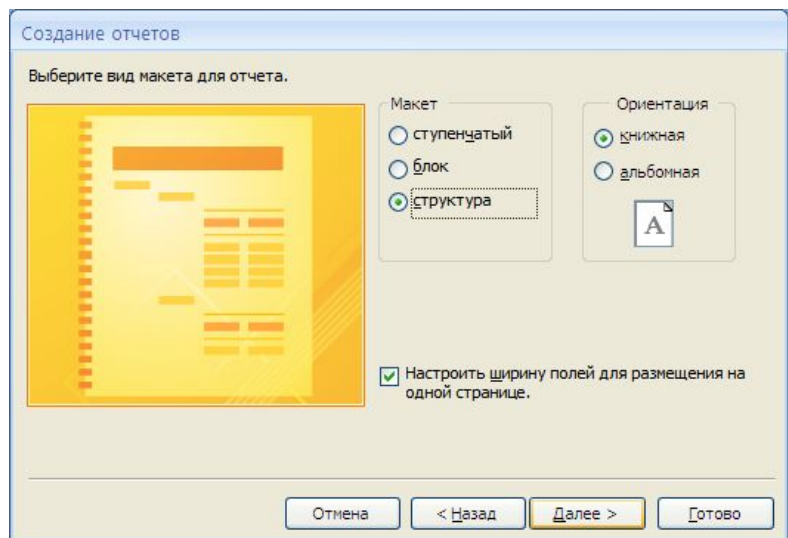
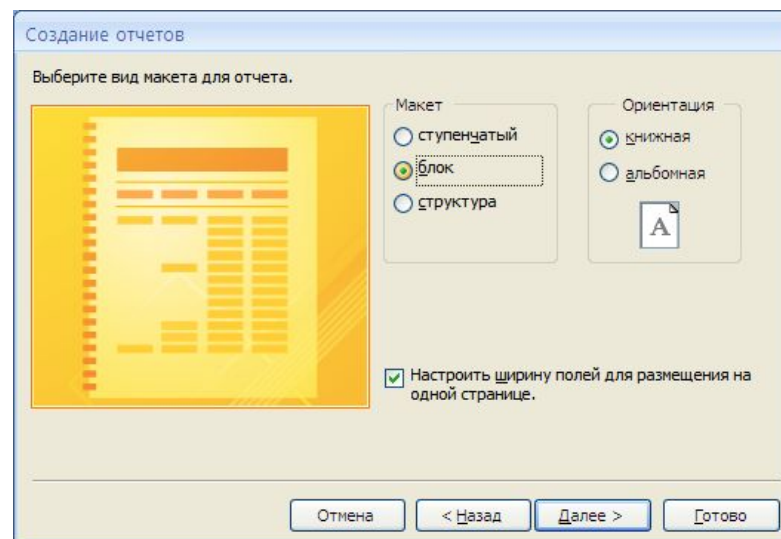
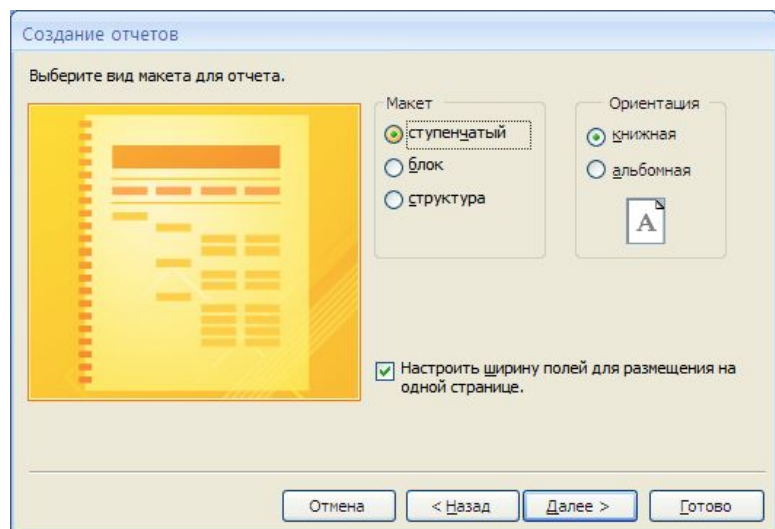
Рис. 74. Выбор варианта представления данных



На следующем этапе нам предлагается определить порядок сортировки записей. Выберем на первом уровне сортировки полк “Звонок дата” и определим, что сортировка будет по возрастанию. Нажмем кнопку “Итоги”. В дополнительном окне нам предлагается выбрать одну или несколько функций расчета итоговых значений. Нас интересует общая стоимость разговоров. Соответственно, выберем “Sum”. Нажмем “OK”. Окно выбора итогов закроется. В окне “Создание отчетов” нажмем кнопку “Далее”.

Рис. 75. Сортировка и подведение итогов

На следующем этапе нам предлагается выбрать внешний вид нашего отчета, а также определить ориентацию бумаги при выводе отчета на печать.

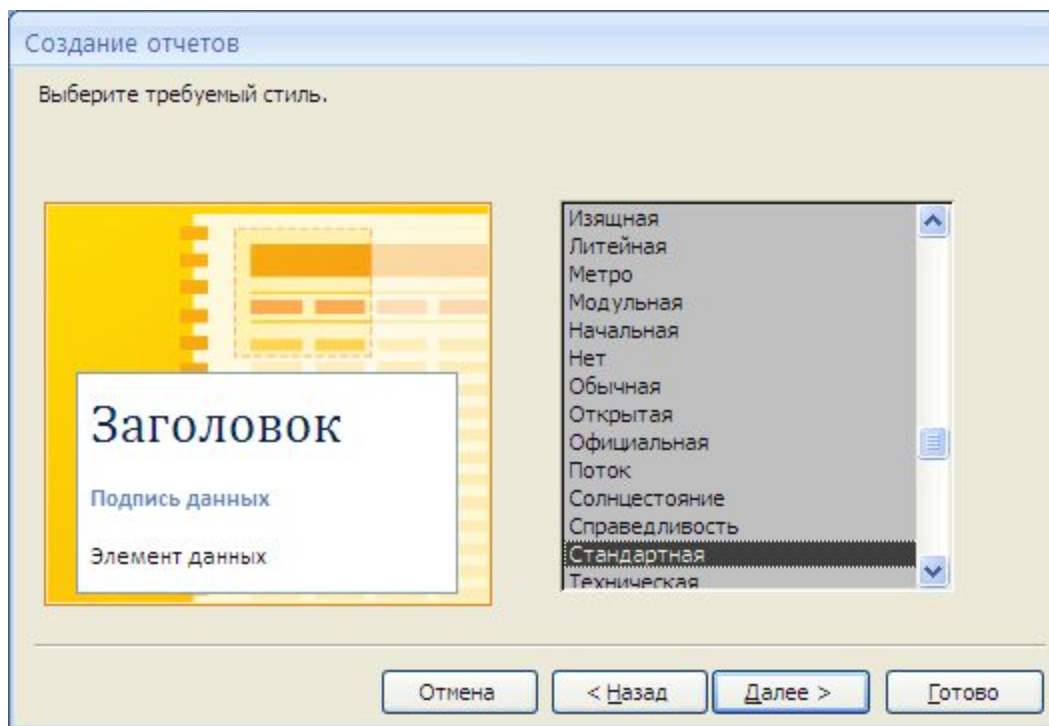


(Обратите внимание – стилизованные изображения отчетов меняются при выборе вариантов макета, показывая нам, каков, в общих чертах, будет внешний вид отчета при распечатке на бумагу)

Выберем вариант “Блок” и нажмем кнопку “Далее”.

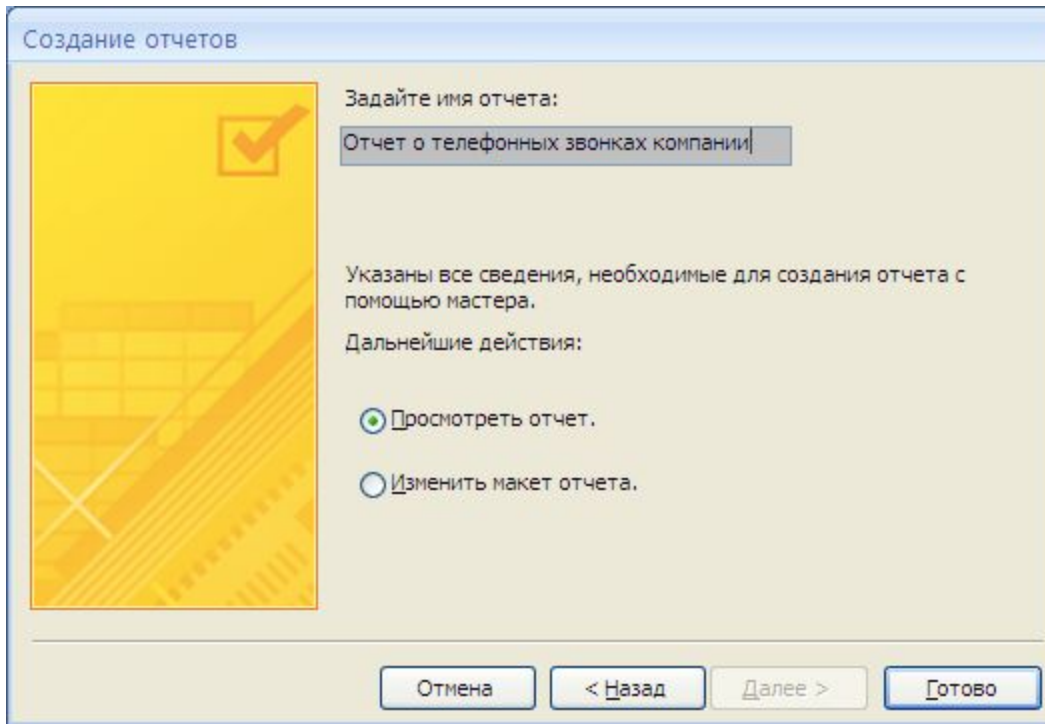
Рис. 76. Варианты распечатки отчета

Далее, нам предлагается выбрать стиль оформления отчета.



Стилей много.  
Оставим стандартный и  
нажмем кнопку “Далее”.

Рис. 77. Выбор стиля отчета.



На данном этапе нам предлагается задать имя отчета. Пусть это будет “Отчет о телефонных звонках компании”.

Оставим выбор “Просмотреть отчет” и нажмем на кнопку “Готово”.

Рис. 78. Задание наименования отчета.

## Отчет о телефонных звонках компании

Фамилия	Иванов
Имя	Иван
Отчество	Иванович
Код региона	495
Телефон	1234567
	Звонок дата    Стоимость разговора
	#####                    3,00р.
Итого для 'Код телефона' = 4 (1 запись)	
Sum	3,00р.
Код региона	905
Телефон	7654321
	Звонок дата    Стоимость разговора
	#####                    0,00р.
	#####                    1,00р.
Итого для 'Код телефона' = 5 (2 записей)	
Sum	1,00р.
Итого для 'Код' = 3 (3 записей)	
Sum	4,00р.
<b>ИТОГО</b>	<b>4,00р.</b>

Смотрим, что получилось. Все вроде бы хорошо – но дата и время звонков не вывелись. (Заполнение поля знаком # означает, что для вывода значения недостаточно места).

Закроем отчет. Выберем его на закладке переходов и перейдем в режим конструктора.

Рис. 79. Отчет.



Отчет о телефонных звонках компании

# Отчет о телефонных звонках компании

Верхний колонтитул

Заголовок группы 'Код'

Фамилия	Фамилия
Имя	Имя
Отчество	Отчество

Заголовок группы 'Код телефона'

Код региона	Код регион
Телефон	Телефон
Звонок дата	Стоимость разговора

Область данных

Звонок дата	Стоимость разговора
-------------	---------------------

Примечание группы 'Код телефона'

=Итого для " & "Код телефона" + " & " & [Код телефона] & " (" & Count(*) & " " & If(Count(*)=1,"запись","записей") & "	
Sum	=Sum([Стоимость разговора])

Примечание группы 'Код'

=Итого для " & "Код" = " & " & [Код] & " (" & Count(*) & " " & If(Count(*)=1,"запись","записей") & "	
Sum	=Sum([Стоимость разговора])

Нижний колонтитул

Примечание отчета

Итого

Окно свойств

Возможен выбор: Надпись

Звонок дата\_Надпись

Макет | Данные | События | Другие | Все

Подпись	Звонок дата
Вывод на экран	Да
Ширина	3,492см
Высота	0,503см
От верхнего края	1,799см
От левого края	1,915см
Тип фона	Прозрачный
Цвет фона	#FFFFFF
Тип границы	Отсутствует
Ширина границы	Сверхтонкая
Цвет границы	#C0504D
Оформление	обычное
Шрифт	Calibri
Размер шрифта	9
Выравнивание текста	По левому краю
Насыщенность	полужирный
Подчеркнутый	Нет
Курсив	Нет
Цвет текста	#5C83B4
Междустрочный интервал	0см
Адрес гиперссылки	
Дополнительный адрес	
Стиль линий сетки сверху	Отсутствует
Стиль линий сетки снизу	Отсутствует
Стиль линий сетки слева	Отсутствует
Стиль линий сетки справа	Отсутствует
Цвет линий сетки	#000000
Ширина линий сетки сверху	1 пункт
Ширина линий сетки снизу	1 пункт
Ширина линий сетки слева	1 пункт
Ширина линий сетки справа	1 пункт
Верхнее поле	0см
Нижнее поле	0см

Рис. 80. Отчет в режиме конструктора.

## Отчет о телефонных звонках компании

Фамилия            Иванов  
Имя                Иван  
Отчество         Иванович

Код региона        495  
Телефон            1234567

Звонок	дата	Стоимость разговора
--------	------	---------------------

	05.04.10 14:30:15	3,00р.
--	-------------------	--------

Итоги для 'Код телефона' = 4 (1 запись)

Sum		3,00р.
-----	--	--------

Код региона        905  
Телефон            7654321

Звонок	дата	Стоимость разговора
--------	------	---------------------

	07.04.10 12:37:40	0,00р.
--	-------------------	--------

	12.04.10 10:02:10	1,00р.
--	-------------------	--------

Итоги для 'Код телефона' = 5 (2 записей)

Sum		1,00р.
-----	--	--------

Итоги для 'Код' = 3 (3 записей)

Sum		4,00р.
-----	--	--------

ИТОГО		4,00р.
-------	--	--------

18 апреля 2010 г.

Растянем влево рамку для вывода поля звонок дата так, чтобы туда поместились все данные о дате и времени звонка. Кроме того, зададим для этого поля прижим “влево”.

(Вкладка “Свойства” ⇨ “Макет” ⇨ “Выравнивание текста” ⇨ “По левому краю”)

Сохраним отчет и запустим его на выполнение. (Рис 81). Уже лучше. Дата и время звонка появились. Но имеются записи типа “Итоги для 'Код телефона' = 4 (1 запись)” Эти записи нам совсем не нужны. Кроме того, заменим заголовок “Sum” внутреннего уровня сортировки на “Итого по телефону”, а заголовок “Sum” второго уровня – на “Итого по сотруднику”. Для этого снова закроем отчет и откроем его в режиме конструктора.

Рис. 81. Отчет .

Отчет о телефонных звонках компании												
Заголовок отчета												
Отчет о телефонных звонках компании												
Верхний колонтитул												
Заголовок группы 'Код'												
Фамилия		Фамилия										
Имя		Имя										
Отчество		Отчество										
Заголовок группы 'Код телефона'												
Код региона		Код регион										
Телефон		Телефон										
Звонок дата		Стоимость разговора										
Область данных												
Звонок дата		Стоимость разговора										
Примечание группы 'Код телефона'												
Итого для " & "Код телефона" & " & " & {Код телефона} & " (" & Count(*) & " " & If(Count(*)=1;"запись";"записей") & "!"												
Sum		=Sum([Стоимость разг										
Примечание группы 'Код'												
Итого для " & "Код" & " & " & {Код} & " (" & Count(*) & " " & If(Count(*)=1;"запись";"записей") & "!"												
Sum		=Sum([Стоимость разг										
Нижний колонтитул												
=Now()											="Стр." & [Page] & " из " & [Pages]	
Примечание отчета												
ИТОГО		=Sum([Стоимость разг										

Рис. 82. Удаление ненужных итоговых строк.

Выберем эти итоговые элементы и после того, как они станут выбранными удалим их. (Просто нажав клавишу "Delete" на клавиатуре)

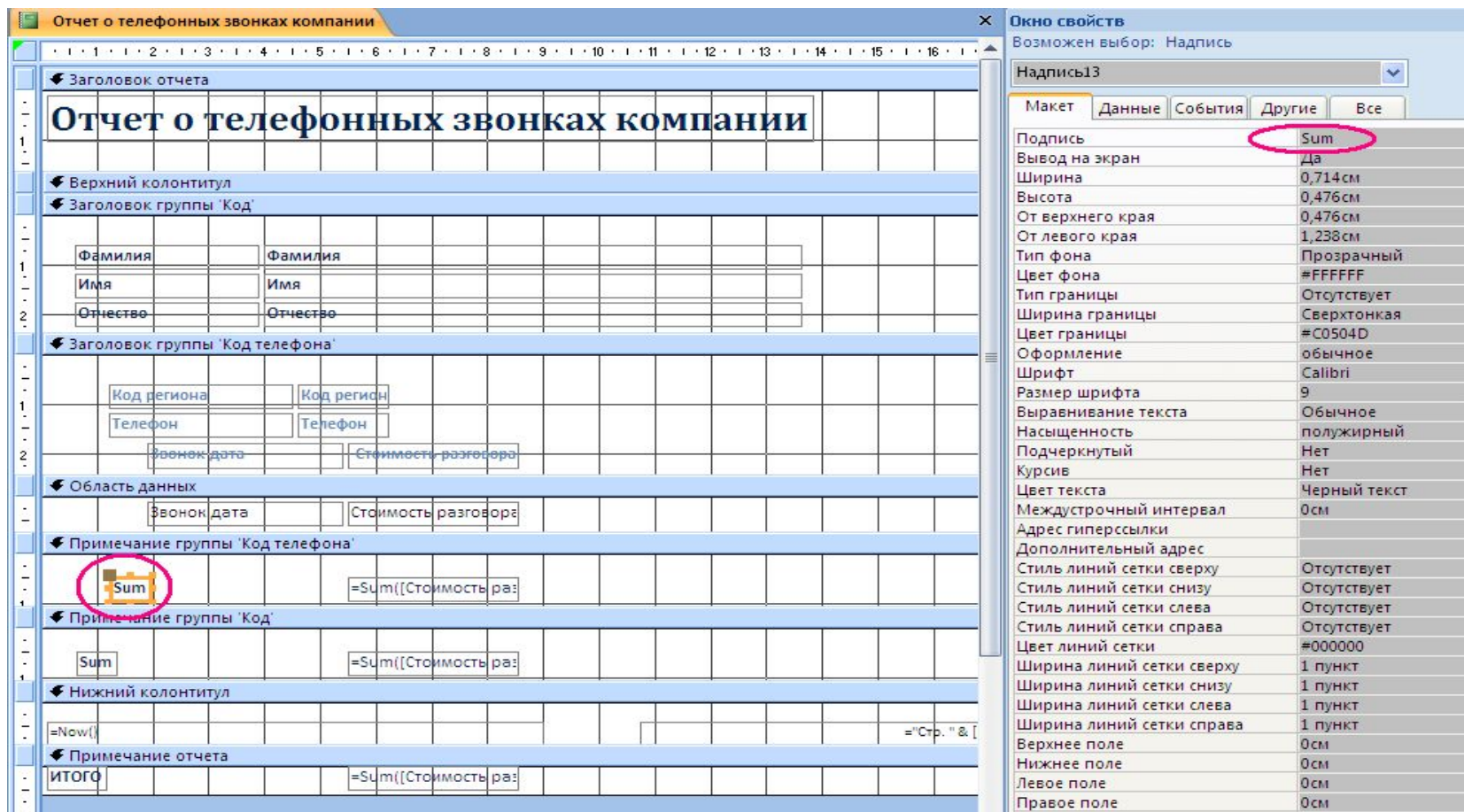


Рис. 83. Корректировка надписи Sum внутренней группировки звонков (по “Коду телефона”)

Найдем поле “Sum” внутренней группировки и изменим подпись на “Итого по телефону” и растянем соответствующую ячейку вправо. (Чтобы надпись уместилась)

Отчет о телефонных звонках компании

Окно свойств  
Возможен выбор: Надпись

Надпись7

Макет Данные События **Другие** Все

Подпись	Итого по сотруднику
Вывод на экран	Да
Ширина	4,365см
Высота	0,476см
От верхнего края	0,476см
От левого края	0,635см
Тип фона	Прозрачный
Цвет фона	#FFFFFF
Тип границы	Отсутствует
Ширина границы	Сверхтонкая
Цвет границы	#C0504D
Оформление	обычное
Шрифт	Calibri
Размер шрифта	9
Выравнивание текста	Обычное
Насыщенность	полужирный
Подчеркнутый	Нет
Курсив	Нет
Цвет текста	Черный текст
Междустрочный интервал	0см
Адрес гиперссылки	
Дополнительный адрес	
Стиль линий сетки сверху	Отсутствует
Стиль линий сетки снизу	Отсутствует
Стиль линий сетки слева	Отсутствует
Стиль линий сетки справа	Отсутствует
Цвет линий сетки	#000000
Ширина линий сетки сверху	1 пункт
Ширина линий сетки снизу	1 пункт
Ширина линий сетки слева	1 пункт
Ширина линий сетки справа	1 пункт
Верхнее поле	0см
Нижнее поле	0см
Левое поле	0см
Правое поле	0см

Заголовок отчета

# Отчет о телефонных звонках компании

Верхний колонтитул

Заголовок группы 'Код'

Фамилия	Фамилия
Имя	Имя
Отчество	Отчество

Заголовок группы 'Код телефона'

Код региона	Код регион
Телефон	Телефон
Дата звонка	Стоимость разговора

Область данных

Звонок дата	Стоимость разговора
-------------	---------------------

Примечание группы 'Код телефона'

Итого по телефону	=Sum([Стоимость разг
-------------------	----------------------

Примечание группы 'Код'

Итого по сотруднику	=Sum([Стоимость разг
---------------------	----------------------

Нижний колонтитул

=Now()	"Стр." & [
--------	------------

Примечание отчета

ИТОГО	=Sum([Стоимость разг
-------	----------------------

Рис. 85. Корректировка надписи Sum группировки звонков (по “Коду” сотрудника) и надписи “Звонок дата”

## Отчет о телефонных звонках компании

Фамилия	Иванов	
Имя	Иван	
Отчество	Иванович	
Код региона	495	
Телефон	1234567	
Дата звонка	Стоимость разговора	
	05.04.10 14:30:15	3,00р.
Итого по телефону	3,00р.	
Код региона	905	
Телефон	7654321	
Дата звонка	Стоимость разговора	
	07.04.10 12:37:40	0,00р.
	12.04.10 10:02:10	1,00р.
Итого по телефону	1,00р.	
Итого по сотруднику	4,00р.	
ИТОГО	4,00р.	

18 апреля 2010 г.

Аналогичным образом откорректируем надпись “Sum” для среднего уровня группировки (По “Коду” сотрудника)

Ну и заодно, придадим более удобочитаемый вид надписи “Звонок дата” на более привычную надпись “Дата звонка” (Рис. 85). Сохраним и закроем отчет.

Запустим его на исполнение. Результат – на рис. 86.

Признаем его удовлетворительным и закончим на этом работу с данным отчетом.

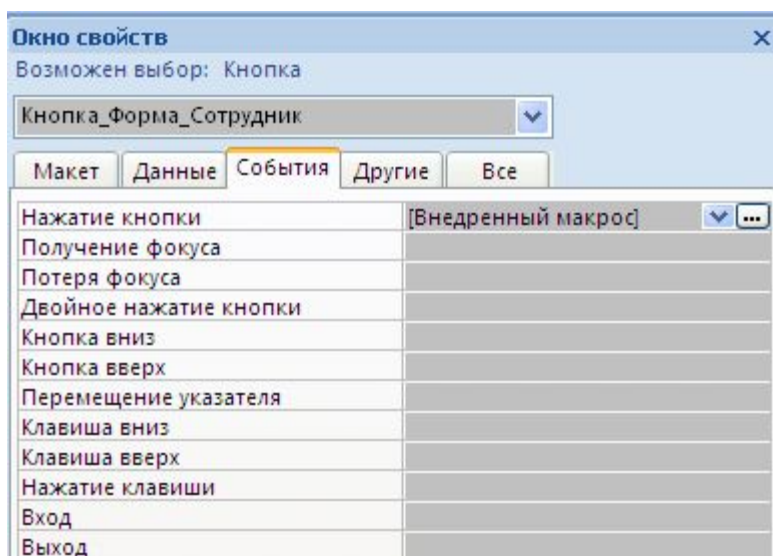
Рис. 86. Отчет о телефонных звонках. Окончательный вариант.

Теперь объединим все наши формы так, чтобы их можно было вызвать из одного окна. Для этого создадим новую (пустую) форму и назовем ее COMMON\_FORM.

Наверху разместим надпись, в которой объявим что это есть “БАЗА ДАННЫХ ОТСЛЕЖИВАНИЯ ТЕЛЕФОННЫХ ЗВОНКОВ КОМПАНИИ”

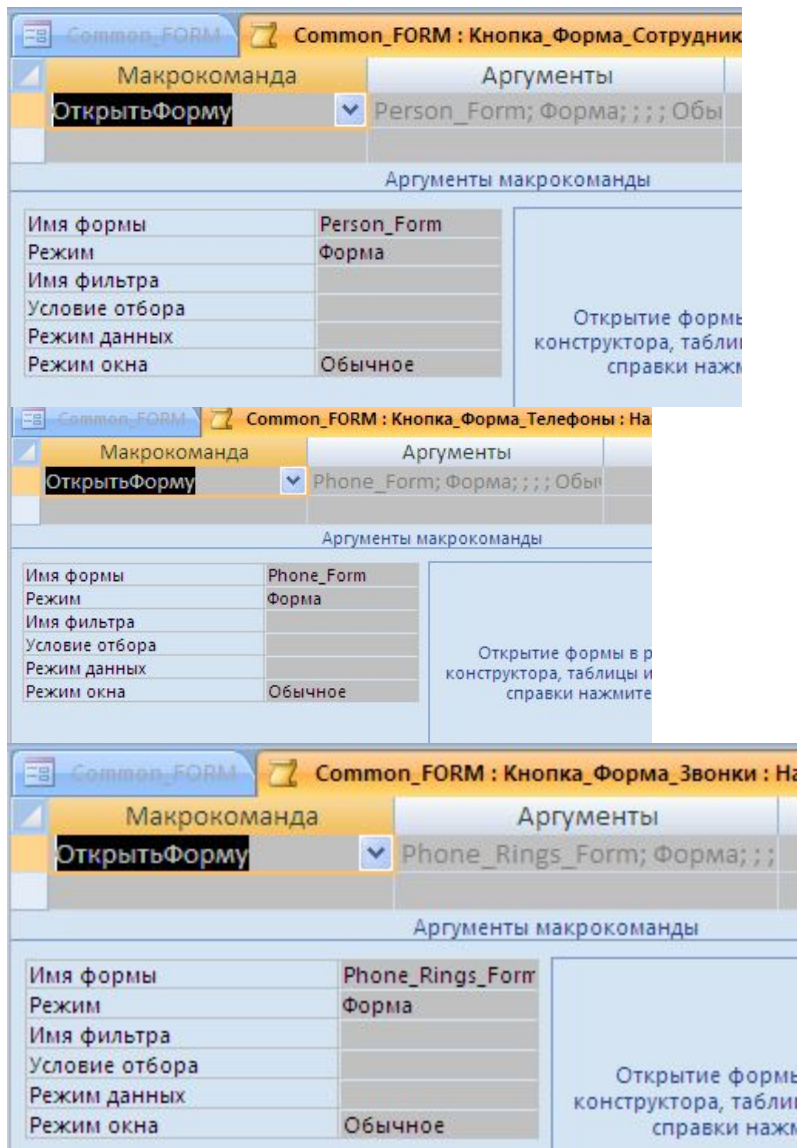
Ниже разместим 5 кнопок и напишем на них “СОТРУДНИКИ”, “ТЕЛЕФОНЫ”, “ЗВОНКИ”, “Запрос:Звонки Иванова” и “Отчет о звонках”.

Откроем страницу свойств кнопки и перейдем на закладку “События”



При выборе события “Нажатие кнопки”. При нажатии на стрелку можно выбрать тип реакции. Выбираем “Внедренный макрос”. Нажимаем на кнопку рядом.

Рис. 87. Выбор свойства для события “Нажатие кнопки”



Выбираем макрокоманду “Открыть форму” В качестве аргументов задаем выбор нужной формы.

Для кнопки “Сотрудники” – Person\_Form.

Для кнопки “Телефоны” – Phone\_Form.

Для кнопки “Звонки” - Phone\_Rings\_Form.

Для всех трех форм:

Режим – “Форма”.

Режим окна – “Обычное”.

(См. рис. 88)

Рис. 88. Определение параметров Макроса для кнопок открытия форм.



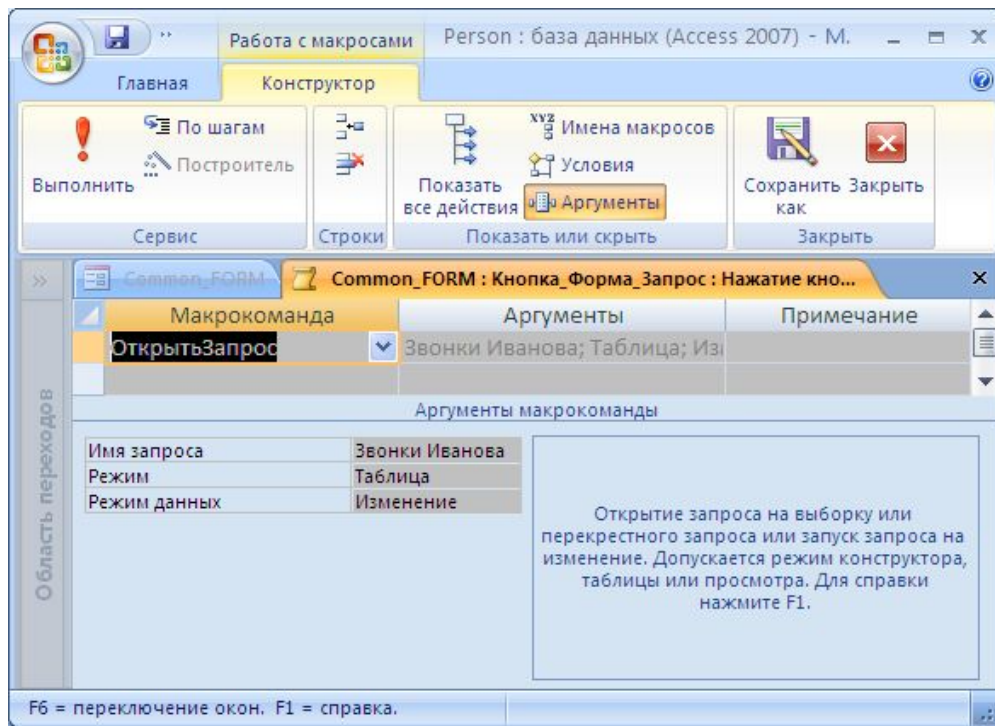


Рис. 89. Определение параметров макроса для кнопки открытия запроса “Звонки Иванова”

Для кнопки “Запрос: Звонки Иванова” определяем следующие свойства:

Макрокоманда – “Открыть запрос”

Имя запроса – “Звонки Иванова”

Режим – “Таблица”

Режим данных – “Изменение” ( Если не хотим, чтобы пользователь мог изменять данные через этот запрос – тогда ставим режим “Только чтение”)

Теперь перейдем к нашей последней кнопке – “Отчет о звонках телефонной компании”

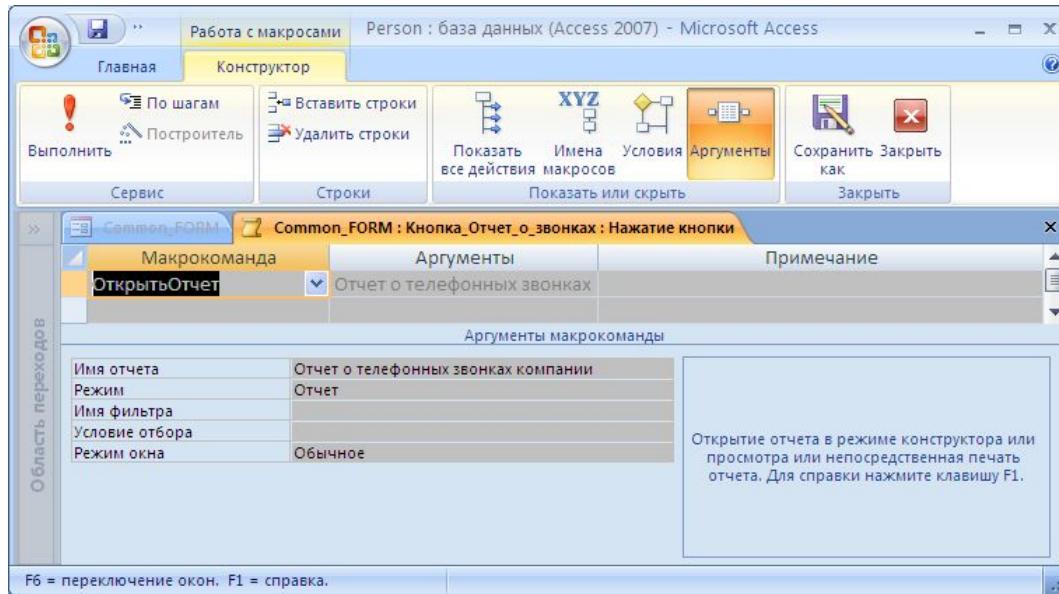



Рис. 90. Определение параметров макроса для кнопки открытия отчета о телефонных звонках компании.

Для кнопки “Отчет: Звонки компании” определяем следующие свойства:

Макрокоманда – “Открыть отчет”

Имя отчета – “Отчет о телефонных звонках компании”

Режим – “Отчет”

Чисто в декоративных целях определим в качестве формы какую-нибудь картинку. (Свойства формы ⇨ Макет ⇨ Рисунок ⇨  Далее выбираем файл с рисунком.)



Все сохраняем и закрываем, а потом вызываем “Common\_Form” на исполнение двойным кликом на этом объекте в области переходов. На рис.91 – результат. Кликнув на каждую из кнопок убеждаемся, что все формы / запрос / отчет вызываются.

Рис. 91. Форма вызова других форм.