

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра информатики и прикладной математики

ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к контрольной работе по дисциплине

«ТЕХНОЛОГИИ БАЗ ДАННЫХ И ЗНАНИЙ»

для студентов экономических специальностей

заочной формы обучения

Брест 2006

Методические указания содержат варианты заданий, вопросы и пояснения к выполнению заданий контрольной работы по курсу «Технологии баз данных и знаний».

Методические указания предназначены для использования студентами экономических специальностей заочной формы обучения.

Составители: Е. Н. Рубанова, старший преподаватель
И. Н. Аверина, старший преподаватель
И. М. Гучко, старший преподаватель

Введение

Курс «Технологии баз данных и знаний» является логическим продолжением курса «Компьютерные информационные технологии». В нем изучаются современные технологии баз данных, которые широко применяются для повышения эффективности во всех сферах бизнеса и экономической деятельности – производстве, торговле, маркетинге, финансах, бухгалтерском учете, менеджменте и др. Дается представление об исследованиях и разработках в области технологий баз знаний.

Цель курса – приобретение знаний о современных технологиях организации, хранения и обработки данных в базах и практических навыков по созданию баз данных и работе с ними при решении практических задач в своей предметной области, а также ознакомление с достижениями в области технологий баз знаний.

Предмет изучения – технология проектирования реляционных баз данных и ее реализация на основе базовых возможностей СУБД Microsoft Access.

Освоение курса «Технологии баз данных и знаний» предполагает получение знаний:

- о представлениях экономической информации в автоматизированных информационных системах;
- концепциях моделирования данных, принципах организации баз данных и их проектирования;
- назначении, архитектуре, функциональных возможностях современных систем управления базами данных (СУБД).

Кроме того, в данном курсе приобретаются практические навыки в среде СУБД:

- проектирования и создания баз данных;
- формирования запросов к базе данных;
- проектирования приложений базы данных – форм и отчетов;
- управления базой данных.

Знания и практические навыки, приобретенные в курсе, будут востребованы при изучении специальных дисциплин, выполнении научных, курсовых, дипломных работ, а также в будущей профессиональной деятельности.

Основной формой изучения курса «Технологии баз данных и знаний» для студентов-заочников является самостоятельная работа с компьютером и учебными пособиями. Список основных из них приводится в конце методических указаний.

Общие методические указания к выполнению контрольной работы

Выполнение заданий контрольной работы (создание базы данных, запросов, форм и отчетов) можно выполнять с использованием различных версий СУБД Microsoft Access.

Контрольная работа состоит из 4-ех заданий, при выполнении которых следует придерживаться указанных ниже требований. Работа, выполненная без их соблюдения, к защите не допускается и возвращается студенту на доработку.

1. **Номер варианта контрольной работы совпадает с последними двумя цифрами номера зачетной книжки (шифра).** Для последних цифр шифра (NN), превышающих число 27, следует определить номер варианта следующим образом: NN-10. Контрольная работа, выполненная не по своему варианту, возвращается студенту без проверки и к защите **не допускается!**
2. Контрольная работа должна быть выполнена и представлена на проверку в срок, предусмотренный учебным планом.
3. Контрольная работа должна быть выполнена на отдельных листах формата А4 и жестко скреплена. Страницы должны быть пронумерованы.
4. На титульном листе указывается название кафедры (кафедра информатики и прикладной математики), наименование дисциплины («Технологии баз данных и знаний»), факультет, номер курса, группа, фамилия, имя, отчество студента, номер зачетной книжки (шифр) и номер выполняемого варианта.
5. Выполненные задания нужно располагать в порядке своих номеров. Каждое задание нужно оформлять следующим образом:
 - *Текст задания для своего варианта.*
 - *Описание выполнения задания.*
 - *Результат выполнения задания.*
6. В конце работы нужно указать список используемой литературы, поставить подпись и дату выполнения работы.

В случае, если работа «не допущена к защите», необходимо выполнить работу над ошибками и представить исправленные задания на повторное рецензирование вместе с первым вариантом решения работы.

В случае необходимости студент может обращаться за консультациями к преподавателю кафедры, проверяющему контрольные работы в группе, лектору потока, либо к преподавателям, проводящим консультации студентов-заочников по графику, утвержденному на кафедре.

1. Содержание контрольной работы

1.1. Условие задания № 1

В задании № 1 необходимо ответить на теоретический вопрос согласно варианту. Варианты задания указаны в разделе 2 «Варианты заданий».

1.2. Условие задания № 2

Задание № 2 предусматривает создание реляционной базы данных, состоящей из трех взаимосвязанных таблиц. В задании требуется:

1.2.1. Создать три таблицы в соответствии с вариантом задания (указанных в разделе 2), присвоить полям подходящие типы данных, назначить им свойства;

1.2.2. Задать в таблицах ключевые поля. В главных таблицах (таблица 1 и таблица 2) для первого поля задать простой ключ, а в подчиненной таблице (таблица 3) – самостоятельно определить и задать простой или составной ключ;

1.2.3. Определить тип связи между таблицами и создать соответствующие связи;

1.2.4. Заполнить таблицы конкретными значениями данных, исходя из их смысла. Главные таблицы должны содержать не менее 10 записей, а подчиненная не менее 15 записей;

1.2.5. Описать выполнение задания, включив в описание структуру каждой из созданных таблиц в следующем виде:

Таблица1: Название таблицы

| Название поля | | Поле 1 | Поле 2 | Поле 3 | ... |
|---------------------------|-------------------------|--------|--------|--------|-----|
| Тип поля | | | | | |
| Основные свойства поля | Размер поля | | | | |
| | Формат поля | | | | |
| | Число десятичных знаков | | | | |
| | Маска ввода | | | | |
| | Подпись | | | | |
| | Значение по умолчанию | | | | |
| | Условие на значение | | | | |
| | Сообщение об ошибке | | | | |
| | Обязательное поле | | | | |
| | Индексированное поле | | | | |
| | Пустые строки | | | | |

1.2.6. Получить распечатки заполненных таблиц и распечатку схемы данных созданной базы данных.

1.3. Условие задания № 3

1.3.1. Для созданной базы данных **сформулировать** и **создать** два запроса на выборку:

- записей, удовлетворяющих условию с логическим оператором **ИЛИ**;
- записей, удовлетворяющих условию с логическим оператором **И**.

Условия в созданных запросах должны быть параметрами !!!

1.3.2. Создать конкретные запросы, формулировки условий которых указаны в соответствующих вариантах заданий баз данных, а именно:

- запросы на вычисление, т.е. с использованием вычисляемых полей;
- итоговый запрос;
- перекрестный запрос.

1.3.3. Описать выполнение задания.

1.3.4. Распечатать (или представить схематично) бланк запроса QBE для каждого из запросов и результат выполнения каждого запроса.

1.4. Условие задания № 4

Задание № 4 предусматривает проектирование объектов БД – форм и отчетов в СУБД. В задании 4 требуется:

1.4.1. Создать форму для просмотра данных подчиненной таблицы, используя **Мастер форм**.

1.4.2. Используя режим конструктора, в созданную форму добавить вычисляемые поля, в которых необходимо разместить названия месяца и дня недели для даты, которая содержится в одном из полей подчиненной таблицы.

1.4.3. Сохранить созданную форму и вывести на печать для одной записи.

1.4.4. Создать отчет на основе подчиненной таблицы, используя **Мастер отчетов**. В отчете предусмотреть:

- группировку записей по полю, содержащему повторяющиеся значения;
- расчет итога для каждой группы записей;
- расчет общего итога по выбранному полю – либо минимального, либо максимального либо среднего, либо суммарного значения;
- в заголовок отчета включить ФИО студента и группу.

1.4.5. Отчет сохранить и вывести на печать первую страницу полученного отчета.

1.4.6. Создать управляющую форму, позволяющую:

- открывать созданную в пункте 1.4.1. форму;
- просматривать созданный в пункте 1.4.4. отчет.

1.4.7. Управляющую форму сохранить и вывести на печать.

1.4.8. Описать выполнение задания 4.

2. Варианты заданий

2.1. Варианты к заданию № 1

1. Понятие базы данных (БД). Виды моделей организации данных.
2. Системы управления базами данных (СУБД). Реляционные СУБД. Правила построения реляционных баз данных.
3. Общая характеристика СУБД MS Access. Способы создания баз данных в MS Access. Порядок создания файла новой БД.
4. Объекты MS Access. Их назначение и особенности. Общие принципы создания.
5. Таблица как объект БД. Понятие записи и поля. Способы создания таблиц в MS Access. Режим Конструктора таблиц. Создание и изменение структуры таблицы.
6. Понятие ключевого поля в таблицах. Виды ключевых полей. Назначение и использование.
7. Типы данных, используемых в MS Access, их краткая характеристика.
8. Свойства полей: *размер поля, формат поля*. Примеры.
9. Свойства полей: *условие на значение и значение по умолчанию*. Примеры.
10. Свойства полей: *маска ввода*. Коды для создания масок. Примеры.
11. Схема данных. Определение связей между таблицами. Изменение существующих связей. Целостность данных.
12. Работа в Режиме таблицы. Модификация макета таблицы. Сортировка данных.
13. Запрос как объект БД. Типы и возможности запросов. Способы создания запросов в MS Access.
14. Режим конструктора запроса. Вид бланка QBE. Способы задания полей. Выполнение и сохранение запроса. Просмотр результатов.
15. Запрос-выборка. Назначение. Виды. Способы создания. Задание условия отбора в бланке QBE. Комбинация нескольких условий отбора.
16. Задание условий отбора для поля *Дата / время*.
17. Использование операторов BETWEEN, IN, LIKE. Примеры.
18. Выражения. Построитель выражений. Полное имя поля в выражении.
19. Функции *Даты/времени*.
20. Логическая функция **Iif()**: назначение, синтаксис и примеры использования.
21. Вычисляемое поле. Назначение. Правила и способы создания. Свойства вычисляемого поля. Примеры.
22. Создание запросов с группировкой. Установки Групповых операций. Назначение группировки. Выбор записей, формирующих группы. Примеры.
23. Запрос с параметром. Назначение. Правила создания. Примеры.
24. Перекрестный запрос. Назначение. Правила и способы создания.
25. Запросы-действия. Виды и назначение. Особенности создания и выполнения.
26. Форма как объект БД. Назначение. Режимы просмотра. Виды форм. Способы проектирования форм в MS Access. Формы, создаваемые мастером.
27. Отчет как объект БД. Отличия отчетов от форм. Просмотр и печать отчетов.

2.2. Варианты к заданиям №№ 2, 3, 4

ВАРИАНТ 1

База данных «БИБЛИОТЕКА»

| ТАБЛИЦА 1 «КНИГИ» | ТАБЛИЦА 2 «ЧИТАТЕЛИ» | ТАБЛИЦА 3 «УЧЕТ ВЫДАЧИ КНИГ» |
|------------------------------|---------------------------------|---|
| • НОМЕР КНИГИ | • НОМЕР ЧИТАТЕЛЯ | • НОМЕР ПО ПОРЯДКУ |
| • НАИМЕНОВАНИЕ | • Ф И О ЧИТАТЕЛЯ | • КОД ЧИТАТЕЛЯ |
| • АВТОР | • АДРЕС | • КОД КНИГИ |
| • ЦЕНА | • ТЕЛЕФОН | • ДАТА ВЫДАЧИ КНИГИ |
| • ГОД ИЗДАНИЯ | • № ПАСПОРТА | • ДАТА ВОЗВРАТА |

Запросы с вычислением полей:

- Вычислить «возраст» каждой книги в годах, а также в месяцах.
- Подсчитать, сколько дней находилась на руках выданная книга. Код книги вводится с клавиатуры.

Итоговый запрос:

Подсчитать количество книг, которое брал каждый читатель. Список должен содержать фамилию читателя и число книг.

Перекрестный запрос:

Подсчитать по каждой дате количество выдач каждому читателю и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать наибольшее количество выдач за день.

ВАРИАНТ 2

База данных «АВТОПЕРЕВОЗКИ»

| ТАБЛИЦА 1 «АВТОМОБИЛИ» | ТАБЛИЦА 2 «ВОДИТЕЛИ» | ТАБЛИЦА 3 «ПЕРЕВОЗКИ» |
|---|---|---|
| • КОД МАШИНЫ | • КОД ВОДИТЕЛЯ | • НОМЕР ЗАКАЗА |
| • МАРКА АВТОМОБИЛЯ | • ФИО ВОДИТЕЛЯ | • ДАТА ЗАКАЗА |
| • РЕГИСТРАЦИОННЫЙ НОМЕР ГАИ | • НОМЕР ВОДИТЕЛЬСКОГО УДОСТОВЕРЕНИЯ. | • КОД МАШИНЫ |
| • ДАТА ТЕХ.ОСМОТРА | • КАТЕГОРИЯ ВОДИТЕЛЯ | • КОД ВОДИТЕЛЯ |
| • ВИД ТОПЛИВА | • АДРЕС | • ВРЕМЯ, ЗАТРАЧЕННОЕ НА ВЫП. ЗАКАЗА (ч.) |
| • НОРМА РАСХОДА ТОПЛИВА НА 1 КМ ПРОБЕГА (л) | • ДАТА ПОСТУПЛЕНИЯ НА РАБОТУ | • ПРОБЕГ (км) |

Запросы с вычислением полей:

- Вычислить стаж каждого водителя в годах на текущую дату, а также день недели поступления на работу.
- Рассчитать, каким будет расход топлива по выполненному заказу, номер которого вводится с клавиатуры.

Итоговый запрос:

Подсчитать общее количество заказов, пробег и затраченное время каждым водителем. Список должен содержать фамилию и категорию водителя.

Перекрестный запрос:

Подсчитать суммарный пробег каждого водителя на разных машинах и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общее количество выполненных водителем заказов.

ВАРИАНТ 3

База данных «ОСНОВНЫЕ СРЕДСТВА»

| ТАБЛИЦА 1 «ВИДЫ ОСНОВНЫХ СРЕДСТВ» | ТАБЛИЦА 2 «МАТЕРИАЛЬНООТВЕТСТВЕННЫЕ ЛИЦА (МОЛ)» | ТАБЛИЦА 3 «ОСНОВНЫЕ СРЕДСТВА (ОС)» |
|--------------------------------------|--|---------------------------------------|
| • КОД ВИДА ОС | • КОД МОЛ | • ИНВЕНТАРНЫЙ НОМЕР |
| • НАИМЕНОВАНИЕ ВИДА ОС | • ФАМИЛИЯ ИМЯ ОТЧЕСТВО | • НАЗВАНИЕ ОС |
| • СРОК ПОЛЕЗНОГО ИСПОЛЬЗОВАНИЯ (лет) | • НАИМЕНОВАНИЕ ОТДЕЛА | • КОД ВИДА ОС |
| • ГОД. НОРМА ИЗНОСА (%) | • ТЕЛЕФОН | • КОД МОЛ |
| | | • БАЛАНСОВАЯ СТОИМОСТЬ |
| | | • ДАТА ВВОДА В ЭКСПЛУАТАЦИЮ |

Запросы с вычислением полей:

- Для каждого объекта ОС подсчитать на текущий момент срок полезного использования в месяцах и днях.
- Получить для каждого ОС дату окончания срока полезного использования, исходя из даты ввода в эксплуатацию и срока полезного использования.

Итоговый запрос:

Сколько на предприятии основных средств по каждому виду ОС и какова их общая балансовая стоимость?

Перекрестный запрос:

Подсчитать по каждому МОЛ общее количество закрепленных объектов ОС в разрезе каждого вида и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общую балансовую стоимость всех ОС, закрепленных за каждым МОЛ.

ВАРИАНТ 4

База данных «ОПЛАТА УСЛУГ ТЕЛЕФОННОЙ СВЯЗИ»

| ТАБЛИЦА 1 «КАТЕГОРИИ УСЛУГ» | ТАБЛИЦА 2 «ТЕЛЕФОНЫ» | ТАБЛИЦА 3 «ОПЛАТА ЗА УСЛУГИ» |
|--------------------------------|-----------------------------|--------------------------------------|
| • КОД КАТЕГОРИИ | • НОМЕР ТЕЛЕФОНА | • КОД РАЗГОВОРА |
| • НАИМЕНОВАНИЕ | • ФИО ВЛАДЕЛЬЦА | • ДАТА РАЗГОВОРА |
| • СТОИМОСТЬ 1 мин | • АДРЕС | • НОМЕР ТЕЛЕФОНА |
| | • АБОНПЛАТА | • КОД КАТЕГОРИИ |
| | • ДОЛГ ЗА ПРЕДЫДУЩИЙ ПЕРИОД | • ПРОДОЛЖИТЕЛЬНОСТЬ РАЗГОВОРА (мин.) |

Запросы с вычислением полей:

- Для тех абонентов, у которых долг за предыдущий месяц превышает размер двукратной абонплаты, выдать сообщение – телефон отключить!
- Вычислить стоимость каждого разговора в зависимости от продолжительности и категории услуги.

Итоговый запрос:

Сколько всего разговоров и какова максимальная и минимальная продолжительность разговоров по каждому номеру телефона?

Перекрестный запрос:

Подсчитать по каждому телефону суммарную продолжительность разговоров в зависимости от категории разговора и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общую продолжительность разговоров по каждому телефону.

ВАРИАНТ 5

База данных «РЕАЛИЗАЦИЯ ТОВАРА»

| ТАБЛИЦА 1 «ТОВАРЫ» | ТАБЛИЦА 2 «ПОКУПАТЕЛИ» | ТАБЛИЦА 3 «ЗАКАЗЫ» |
|-------------------------|---------------------------|-----------------------|
| • КОД ТОВАРА | • КОД ПОКУПАТЕЛЯ | • НОМЕР ЗАКАЗА |
| • НАИМЕНОВАНИЕ ТОВАРА | • НАЗВАНИЕ ФИРМЫ | • ДАТА ЗАКАЗА |
| • ЕДИНИЦА ИЗМЕРЕНИЯ | • АДРЕС | • КОД ПОКУПАТЕЛЯ |
| • ЦЕНА ЗА ЕД. (руб) | • ТЕЛЕФОН/ФАКС | • КОД ТОВАРА |
| • ДАТА ИЗГОТОВЛЕНИЯ | • БАНК | • КОЛИЧЕСТВО ТОВАРА |
| • СРОК ГОДНОСТИ (в мес) | • РАСЧЕТНЫЙ СЧЕТ | • СУММА ПРЕДОПЛАТЫ |
| • ДАТА ПОСТУПЛЕНИЯ | | • ДАТА ПРЕДОПЛАТЫ |

Запросы с вычислением полей:

- Для каждого товара вычислить «возраст» товара в днях, месяцах и годах на текущую дату.
- Для товаров, которые заказаны в количестве больше чем 100, начислить скидку к цене в размере 2% от цены, больше чем 300 – 5%.

Итоговый запрос:

Сколько всего заказов сделал каждый покупатель и на какую стоимость?

Перекрестный запрос:

Подсчитать для каждого покупателя сумму предоплаты по каждому заказу и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общую сумму предоплаты по всем заказам.

ВАРИАНТ 6

База данных «ВИДЕОПРОКАТ»

| ТАБЛИЦА 1 «КЛИЕНТЫ» | ТАБЛИЦА 2 «ВИДЕОФИЛЬМЫ» | ТАБЛИЦА 3 «УЧЕТ ВЫДАЧИ ФИЛЬМОВ» |
|------------------------|-----------------------------|---------------------------------------|
| • НОМЕР КЛИЕНТА | • НОМЕР ВИДЕО | • № ПО ПОРЯДКУ |
| • ФИО КЛИЕНТА | • НАЗВАНИЕ ФИЛЬМА | • НОМЕР КЛИЕНТА |
| • АДРЕС | • РЕЖИССЕР | • НОМЕР ВИДЕО |
| • ТЕЛЕФОН | • ТИП ФИЛЬМА | • ДАТА ВЫДАЧИ |
| • № ПАСПОРТА | • ДЛИТЕЛЬНОСТЬ ФИЛЬМА (мин) | • ПРИЗНАК ВОЗВРАТА |
| | | • ДАТА ВОЗВРАТА |

Запросы с вычислением полей:

- Подсчитать, на сколько дней была выдана на прокат кассета клиенту.
- На основе предыдущего запроса вычислить стоимость проката кассеты для клиента, если стоимость проката за 1 день вводится с клавиатуры как параметр.

Итоговый запрос:

Подсчитать, сколько кассет взял каждый клиент, и сколько времени он потратил на просмотр всех взятых фильмов.

Перекрестный запрос:

Подсчитать для каждого типа фильма суммарное количество кассет, выданных каждому отдельному клиенту, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести максимальное количество выданных кассет каждого типа.

ВАРИАНТ 7

База данных «СЕКРЕТАРЬ»

| ТАБЛИЦА 1 «СОТРУДНИКИ» | ТАБЛИЦА 2 «СПИСОК ПОРУЧЕНИЙ» | ТАБЛИЦА 3 «РАСПРЕДЕЛЕНИЕ ПОРУЧЕНИЙ» |
|---------------------------|-----------------------------------|---|
| • КОД СОТРУДНИКА | • КОД ПОРУЧЕНИЯ | • НОМЕР П/П |
| • ФИО СОТРУДНИКА | • ОПИСАНИЕ ПОРУЧЕНИЯ | • КОД СОТРУДНИКА |
| • ДОМАШНИЙ АДРЕС | • МАХ. ВРЕМЯ ВЫПОЛНЕНИЯ (час.) | • КОД ПОРУЧЕНИЯ |
| • ТЕЛЕФОН | • УРОВЕНЬ СЛОЖНОСТИ | • ДАТА ВЫДАЧИ ПОРУЧЕНИЯ |
| • ДОЛЖНОСТЬ | | • ДАТА ВЫПОЛНЕНИЯ ПОРУЧЕНИЯ |
| | | • РАСХОДЫ НА ИСПОЛНЕНИЕ |

Запросы с вычислением полей:

- Подсчитать фактическое время выполнения каждого поручения в днях.
- Для сотрудника, который превысил время выполнения поручения более чем на час, назначить штраф – 2 у.е., более чем на 3 часа – 5 у.е.

Итоговый запрос:

Для каждого сотрудника подсчитать количество выполненных поручений и общее фактически затраченное время на их исполнение.

Перекрестный запрос:

Подсчитать по каждому сотруднику суммарное количество выполненных поручений за каждый день определенного месяца и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общее количество выполненных поручений каждым сотрудником за этот месяц.

ВАРИАНТ 8

База данных «ПЛАТНАЯ ОТРАБОТКА»

| ТАБЛИЦА 1 «СТУДЕНТЫ» | ТАБЛИЦА 2 «ДИСЦИПЛИНЫ» | ТАБЛИЦА 3 «ОТРАБОТКА» |
|-------------------------|--|--------------------------|
| • ШИФР СТУДЕНТА | • КОД ДИСЦИПЛИНЫ | • НОМЕР ВЕДОМОСТИ |
| • Ф И О СТУДЕНТА | • НАИМЕНОВАНИЕ | • ШИФР СТУДЕНТА |
| • ДАТА РОЖДЕНИЯ | • КОЛИЧЕСТВО ЧАСОВ В СЕМЕСТРЕ | • КОД ДИСЦИПЛИНЫ |
| • ФАКУЛЬТЕТ | • ФОРМА КОНТРОЛЯ | • ДАТА ОТРАБОТКИ |
| • СПЕЦИАЛЬНОСТЬ | • СТОИМОСТЬ 1 ЧАСА ОТРАБОТКИ (руб.) | • ОТРАБОТАНО ЧАСОВ |
| • НОМЕР ГРУППЫ | | • № КВИТАНЦИИ ОБ ОПЛАТЕ |
| • АДРЕС | | • ДАТА ОПЛАТЫ |

Запросы с вычислением полей:

- Вычислить возраст каждого студента в годах, а также отдельно и в месяцах, и днях.
- Подсчитать стоимость отработки по каждой ведомости.

Итоговый запрос:

Подсчитать количество разных дисциплин, отработанных каждым студентом, а также уплаченную за отработку сумму.

Перекрестный запрос:

Подсчитать количество отработанных часов студентами по каждой дисциплине в разрезе дат определенного месяца и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общее количество отработанных часов по каждой дисциплине за весь месяц.

ВАРИАНТ 9

База данных «ТОВАРООБОРОТЫ»

| ТАБЛИЦА 1 «МАГАЗИНЫ» | ТАБЛИЦА 2 «ТОВАРЫ» | ТАБЛИЦА 3 «ПОСТАВКИ» |
|-------------------------|------------------------|-------------------------|
| • НОМЕР МАГАЗИНА | • КОД ТОВАРА | • НОМЕР ТТН*** |
| • НАЗВАНИЕ МАГАЗИНА | • НАИМЕНОВАНИЕ ТОВАРА | • ДАТА ТТН |
| • АДРЕС | • ЕДИНИЦА ИЗМЕРЕНИЯ | • НОМЕР МАГАЗИНА |
| • Ф И О ДИРЕКТОРА | • ДАТА ИЗГОТОВЛЕНИЯ | • КОД ТОВАРА |
| • ТЕЛЕФОН/ФАКС | • СРОК ХРАНЕНИЯ (дней) | • КОЛИЧЕСТВО |
| | | • ЦЕНА ЗА 1 ЕД. (руб.) |

*** – товарно-транспортная накладная

Запросы с вычислением полей:

- Вычислить «возраст» (на текущую дату) каждого товара в днях. В запрос включить наименование товара и дату изготовления.
- Для товаров с истекшим сроком хранения выдать сообщение – «заменить», а для товаров, у которых срок хранения истекает через 5 дней, – «срочно реализовать».

Итоговый запрос:

Получить статистическую информацию по каждому товару за весь период: общий и средний объем поставок; общую стоимость; максимальную и минимальную по количеству поставку.

Перекрестный запрос:

Подсчитать суммарное количество каждого товара, поставленного в каждый магазин, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую сумму поставок по каждому товару во все магазины.

ВАРИАНТ 10 База данных «КВАРТПЛАТА»

| ТАБЛИЦА 1 «КВАРТИРЫ» | ТАБЛИЦА 2 «ВИДЫ УСЛУГ» | ТАБЛИЦА 3 «ОПЛАТА» |
|----------------------------|---------------------------|--|
| • КОД КВАРТИРЫ | • КОД УСЛУГИ | • НОМЕР РАСЧЕТА |
| • АДРЕС | • НАИМЕНОВАНИЕ УСЛУГИ | • ДАТА РАСЧЕТА |
| • Ф. И. О. ВЛАДЕЛЬЦА | • ЕД. ИЗМЕРЕНИЯ | • КОД УСЛУГИ |
| • ПЛОЩАДЬ КВАРТИРЫ (м.кв.) | • ТАРИФ (руб.) | • КОД КВАРТИРЫ |
| • КОЛИЧЕСТВО ЖИЛЬЦОВ | | • КОЛ-ВО ПОТРЕБЛЕННОЙ УСЛУГИ |
| • КОЛИЧЕСТВО КОМНАТ | | • ДОЛГ ИЛИ ПЕРЕПЛАТА ЗА ПРЕДЫДУЩИЙ МЕСЯЦ |
| • ЛЬГОТЫ (%) | | |

Запросы с вычислением полей:

- Подсчитать стоимость услуги в рублях с учетом тарифа и потребленного количества услуги.
- На основе предыдущего запроса подсчитать размер льготы (в руб.) как % от стоимости услуги.

Итоговый запрос:

Вывести список количества однокомнатных, двухкомнатных и т.д. квартир с количеством проживающих в них жильцов.

Перекрестный запрос:

Подсчитать общее количество потребленных услуг каждого вида в каждой квартире и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую сумму долга или переплаты по каждой квартире.

ВАРИАНТ 11 База данных «РЕМОНТ КВАТИР»

| ТАБЛИЦА 1 «МАТЕРИАЛЫ» | ТАБЛИЦА 2 «ВИДЫ РАБОТ» | ТАБЛИЦА 3 «ВЫПОЛНЕНИЕ РЕМОНТА» |
|-------------------------------|-----------------------------|--|
| • КОД МАТЕРИАЛА | • КОД РАБОТЫ | • НОМЕР КВИТАНЦИИ |
| • НАИМЕНОВАНИЕ | • НАИМЕНОВАНИЕ | • ФАМИЛИЯ ЗАКАЗЧИКА |
| • СТРАНА ИЗГОТОВИТЕЛЬ | • ОПИСАНИЕ | • КОД МАТЕРИАЛА |
| • ЕД. ИЗМЕРЕНИЯ | • ЦЕНА 1 ЧАСА РАБОТЫ (руб.) | • КОД РАБОТЫ |
| • ЦЕНА ЕДИНИЦЫ (руб.) | | • КОЛ-ВО ЗАТРАЧЕННОГО МАТЕРИАЛА |
| • ДАТА ИЗГОТОВЛЕНИЯ МАТЕРИАЛА | | • ВРЕМЯ, ЗАТРАЧЕННОЕ НА ВЫП. РАБОТЫ (час.) |
| • СРОК ГОДНОСТИ (мес.) | | • ФАМИЛИЯ ИСПОЛНИТЕЛЯ |
| | | • ДАТА ВЫП. РЕМОНТА |

Запросы с вычислением полей:

- Для каждой квитанции подсчитать стоимость материалов за выполненный объем работ и стоимость исполнения ремонтных работ.
- Для материалов, срок годности которых истек, выдать сообщение - «материал не годен к употреблению», для тех, у которых срок реализации истекает через 20 дней, – «срочно реализовать».

Итоговый запрос:

Сколько потребовалось разных видов материалов для выполнения каждого вида работы.

Перекрестный запрос:

Подсчитать общее количество отработанного времени каждым исполнителем в каждом месяце и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общее количество затраченного материала за каждый месяц.

ВАРИАНТ 12

База данных «МУЗЕЙ»

| <u>ТАБЛИЦА 1</u> <u>«МУЗЕЙ»</u> | <u>ТАБЛИЦА 2</u> <u>«ХУДОЖНИКИ»</u> | <u>ТАБЛИЦА 3</u> <u>«КАРТИНЫ»</u> |
|------------------------------------|--|--------------------------------------|
| • КОД МУЗЕЯ | • КОД ХУДОЖНИКА | • КОД КАРТИНЫ |
| • НАЗВАНИЕ | • ИМЯ | • НАЗВАНИЕ |
| • АДРЕС (город) | • ДАТА РОЖДЕНИЯ | • КОД ХУДОЖНИКА |
| • ФИО ДИРЕКТОРА | • СТИЛЬ В ИСКУССТВЕ | • КОД МУЗЕЯ |
| • ТЕЛЕФОН | • МЕСТО ПРОЖИВАНИЯ | • СТОИМОСТЬ КАРТИНЫ (y.e.) |
| • ЦЕНА ВХОДНОГО БИЛЕТА (y.e.) | | • ДАТА ПРИОБРЕТЕНИЯ КАРТИНЫ |

Запросы с вычислением полей:

- Для каждого музея вычислить цену входного билета в рублях, если курс 1 у.е. задается с клавиатуры в виде параметра.
- Для каждого художника определить день недели, в который он родился, и вывести текстовое название дня недели в отдельном поле.

Итоговый запрос:

Сколько художников XVIII века (дата рождения которых попадает на этот период) представлено в каждом музее и каким количеством картин?

Перекрестный запрос:

Подсчитать общее количество картин каждого художника, представленных в каждом музее, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую стоимость картин каждого художника.

ВАРИАНТ 13
База данных «БОЛЬНИЦА»

| ТАБЛИЦА 1 «ОТДЕЛЕНИЯ» | ТАБЛИЦА 2 «ВРАЧИ» | ТАБЛИЦА 3 «БОЛЬНЫЕ» |
|--|------------------------------------|--------------------------------------|
| • КОД ОТДЕЛЕНИЯ | • КОД ВРАЧА | • № РЕГИСТРАЦИИ |
| • НАЗВАНИЕ ОТДЕЛЕНИЯ | • Ф.И.О. ВРАЧА | • Ф И О БОЛЬНОГО |
| • ЗАВ. ОТДЕЛЕНИЕМ | • СПЕЦИАЛИЗАЦИЯ ВРАЧА | • ДАТА РОЖДЕНИЯ |
| • КОЛИЧЕСТВО МЕСТ | • КВАЛИФИКАЦИЯ | • ДАТА ПОСТУПЛЕНИЯ |
| | • ДАТА РОЖДЕНИЯ | • КОД ОТДЕЛЕНИЯ |
| | | • КОД ВРАЧА |
| | | • ДИАГНОЗ |
| | | • ДАТА ВЫПИСКИ |

***... может быть пустым

Запросы с вычислением полей:

- Подсчитать для каждого больного длительность прохождения лечения в больнице в днях на дату выписки, а если больной не выписан – на текущую дату.
- Определить возраст каждого больного и для больных старше 60 лет вывести сообщение - «Пенсионный возраст».

Итоговый запрос:

Подсчитать среднюю продолжительность лечения и количество больных у каждого врача.

Перекрестный запрос:

Посчитать количество поступивших больных в каждое отделение за каждый день и вывести результат в виде перекрестной таблицы. В итоговом столбце для каждого отделения подсчитать общее количество больных.

ВАРИАНТ 14
База данных «ТУРФИРМА»

| ТАБЛИЦА 1 «КЛИЕНТЫ» | ТАБЛИЦА 2 «МАРШРУТЫ» | ТАБЛИЦА 3 «УЧЕТ ПУТЕВОК» |
|--------------------------------------|---------------------------------------|---|
| • КОД КЛИЕНТА | • КОД МАРШРУТА | • № ПУТЕВКИ |
| • Ф И О КЛИЕНТА | • МАРШРУТ | • КОД МАРШРУТА |
| • АДРЕС | • ВИД ТРАНСПОРТА | • КОД КЛИЕНТА |
| • ТЕЛЕФОН | • ДЛИТЕЛЬНОСТЬ (сутки) | • ДАТА ОТПРАВЛЕНИЯ |
| • ПАСПОРТНЫЕ ДАННЫЕ | • СТОИМОСТЬ ПРОЕЗДА (у.е.) | • СУММА ПРЕДОПЛАТЫ (у.е.) |
| • СКИДКА (%) | • СТОИМОСТЬ ПРОЖИВАНИЯ (у.е.) | |
| | • СТОИМОСТЬ ПИТАНИЯ (у.е.) | |
| | • СТОИМОСТЬ СТРАХОВКИ (у.е.) | |

Запросы с вычислением полей:

- Вычислить для каждого маршрута полную стоимость в у.е. и сумму, которую необходимо внести клиенту после предоплаты.
- На основании предыдущего запроса рассчитать размер скидки от полной стоимости для каждого клиента в рублях. Курс 1 у.е. вводится с клавиатуры.

Итоговый запрос:

Подсчитать количество проданных путевок и их общую стоимость по каждому маршруту.

Перекрестный запрос:

Подсчитать общее количество выданных путевок по каждому маршруту за каждый день определенного месяца и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую стоимость (в у.е.) реализованных путевок за каждый день в этом месяце.

ВАРИАНТ 15

База данных «МУЗЫКАЛЬНАЯ КОЛЛЕКЦИЯ»

| ТАБЛИЦА 1 «ИСПОЛНИТЕЛИ» | ТАБЛИЦА 2 «ДИСКИ» | ТАБЛИЦА 3 «СВЕДЕНИЯ О ПЕСНЯХ» |
|----------------------------|----------------------|----------------------------------|
| • КОД ИСПОЛНИТЕЛЯ | • КОД ДИСКА | • КОД ПЕСНИ |
| • ИМЯ ИСПОЛНИТЕЛЯ | • НАЗВАНИЕ ДИСКА | • НАЗВАНИЕ ПЕСНИ |
| • МЕСТО РОЖДЕНИЯ | • ДАТА ЗАПИСИ ДИСКА | • КОД ИСПОЛНИТЕЛЯ |
| • МЕСТО ПРОЖИВАНИЯ | • СТУДИЯ ЗАПИСИ | • КОД ДИСКА |
| • ДАТА РОЖДЕНИЯ | • СТИЛЬ ЗАПИСИ | • ДЛИТЕЛЬНОСТЬ ПЕСНИ |
| | | • ДАТА ЗАПИСИ ПЕСНИ |

Запросы с вычислением полей:

- Подсчитать возраст (в годах) исполнителя, имя которого вводится с клавиатуры.
- Определить день недели, в который родился исполнитель, и вывести текстовое название дня недели в отдельном поле.

Итоговый запрос:

Подсчитать количество песен каждого исполнителя на разных дисках. Вывести в запросе название и год записи диска.

Перекрестный запрос:

Подсчитать суммарную длительность песен у каждого исполнителя в каждом альбоме и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать общую длительность песен во всех альбомах для каждого исполнителя.

ВАРИАНТ 16

База данных «ЗАРПЛАТА»

| ТАБЛИЦА 1 «СПИСОК СОТРУДНИКОВ» | ТАБЛИЦА 2 «ТАРИФНАЯ СЕТКА» | ТАБЛИЦА 3 «НАЧИСЛЕНИЕ ЗАРПЛАТЫ» |
|--------------------------------|----------------------------|---------------------------------|
| • ТАБЕЛЬНЫЙ НОМЕР | • НОМЕР РАЗРЯДА | • НОМЕР ВЕДОМОСТИ |
| • Ф И О СОТРУДНИКА | • КОЭФФИЦИЕНТ | • ДАТА ВЕДОМОСТИ |
| • ОТДЕЛ | • ПОСЛЕДНЯЯ ДАТА ИЗМЕНЕНИЯ | • ТАБЕЛЬНЫЙ НОМЕР |
| • ДАТА ПОСТУПЛЕНИЯ НА РАБОТУ | | • НОМЕР РАЗРЯДА |
| • ПЕРСОНАЛЬНАЯ НАДБАВКА (%) | | • ОТРАБОТАНО ЧАСОВ |

Запросы с вычислением полей:

- Посчитать оклад и размер персональной надбавки (в % от оклада) для каждого сотрудника. Оклад рассчитывается как коэффициент от ставки 1-го разряда, которая вводится с клавиатуры.
- Подсчитать стаж каждого сотрудника в годах на текущую дату.

Итоговый запрос:

Подсчитать сумму к выдаче по каждой ведомости с учетом оклада и надбавки за определенный месяц.

Перекрестный запрос:

Посчитать по каждому сотруднику суммарное количество отработанных часов за каждый месяц и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общий заработок каждого сотрудника.

ВАРИАНТ 17

База данных «ВИННЫЙ ПОГРЕБ»

| ТАБЛИЦА 1 «СПИСОК ВИН» | ТАБЛИЦА 2 «ТИПЫ ВИН» | ТАБЛИЦА 3 «СВЕДЕНИЯ О ПОКУПКЕ ВИН» |
|---------------------------|-------------------------|---------------------------------------|
| • КОД ВИНА | • КОД ТИПА ВИНА | • КОД ПОКУПКИ ВИНА |
| • НАИМЕНОВАНИЕ ВИНА | • НАИМЕНОВАНИЕ | • ДАТА ПОКУПКИ |
| • ПРОИЗВОДИТЕЛЬ | • ЦВЕТ | • КОД ВИНА |
| • ЦЕНА БУТЫЛКИ (руб.) | • КРЕПОСТЬ | • КОД ТИПА ВИНА |
| • ОБЪЕМ БУТЫЛКИ (литр) | | • КОЛИЧЕСТВО БУТЫЛОК |
| • ГОД УРОЖАЯ | | |

Запросы с вычислением полей:

- Подсчитать «возраст» каждого вина. Если «возраст» больше 10 лет, то вывести в отдельном поле сообщение «Выдержка более 10 лет», если больше 5 лет – «Выдержка от 5 до 10 лет».
- Подсчитать стоимость каждой покупки в у.е. Курс 1 у.е. вводится с клавиатуры.

Итоговый запрос:

Подсчитать общее количество купленных бутылок и общий объем вина определенного типа. Тип вина вводится с клавиатуры.

Перекрестный запрос:

Подсчитать общее количества купленных бутылок каждого типа вина в разрезе дат и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести максимальное число купленных бутылок по каждой дате.

ВАРИАНТ 18
База данных «ПЕРЕВОЗКА ПассаЖИРов»

| ТАБЛИЦА 1 «АВТОБУСЫ» | ТАБЛИЦА 2 «ВОДИТЕЛИ» | ТАБЛИЦА 3 «РЕИСЫ» |
|---------------------------------|---------------------------------|------------------------------|
| • НОМЕР АВТОБУСА | • КОД ВОДИТЕЛЯ | • НОМЕР РЕЙСА |
| • МАРКА | • Ф И О ВОДИТЕЛЯ | • ПУНКТ НАЗНАЧЕНИЯ |
| • КОЛИЧЕСТВО МЕСТ | • АДРЕС | • НОМЕР АВТОБУСА |
| • ТЕХНИЧЕСКОЕ СОСТОЯНИЕ | • ТЕЛЕФОН | • КОД ВОДИТЕЛЯ |
| | • КАТЕГОРИЯ | • ДАТА ОТПРАВЛЕНИЯ |
| | • ГОД РОЖДЕНИЯ | • СТОИМОСТЬ 1-го БИЛЕТА |
| | | • КОЛ-ВО ПРОДАННЫХ БИЛЕТОВ |

Запросы с вычислением полей:

- Подсчитать возраст каждого из водителей определенной категории. Категория водителя вводится с клавиатуры.
- Подсчитать по каждому рейсу стоимость всех проданных билетов.

Итоговый запрос:

Подсчитать общую стоимость всех проданных билетов в каждый отдельный пункт назначения.

Перекрестный запрос:

Создать запрос, выводящий количество проданных билетов в разные пункты назначения в разрезе дат определенного месяца, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести количество рейсов в каждый пункт назначения.

ВАРИАНТ 19
База данных «ОПЕРАЦИИ С БАНКАМИ»

| ТАБЛИЦА 1 «БАНКИ» | ТАБЛИЦА 2 «КЛИЕНТЫ» | ТАБЛИЦА 3 «ВИДЫ ОПЕРАЦИЙ» |
|------------------------------|--------------------------------|--------------------------------------|
| • КОД БАНКА | • КОД КЛИЕНТА | • КОД ОПЕРАЦИИ |
| • НАЗВАНИЕ | • Ф И О КЛИЕНТА | • ДАТА ОПЕРАЦИИ |
| • УЧРЕДИТЕЛЬ | • АДРЕС | • КОД БАНКА |
| • УСТАВНЫЙ ФОНД | • ТЕЛЕФОН | • КОД КЛИЕНТА |
| • АДРЕС | • ФАКС | • ВИД РАСЧЕТА |
| • ГОДОВОЙ % | | • СУММА ВКЛАДА |

Запросы с вычислением полей:

- Подсчитать стоимость услуг определенного банка, как 0,1% от суммы вклада. Код банка вводится с клавиатуры.
- Определить день операции (название дня недели) и вывести его в отдельном поле.

Итоговый запрос:

Подсчитать количество вкладчиков и общую сумму вкладов в каждом банке за определенный год.

Перекрестный запрос:

Подсчитать общую сумму вкладов в каждом банке по каждому клиенту и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести максимальное значение вклада по каждому банку.

ВАРИАНТ 20 База данных «РЕМОНТ АВТОМОБИЛЕЙ»

| ТАБЛИЦА 1 «ОБСЛУЖИВАЮЩИЙ ПЕРСОНАЛ» | ТАБЛИЦА 2 «СПИСОК РАБОТ» | ТАБЛИЦА 3 «УЧЕТ РЕМОНТА АВТОМОБИЛЕЙ» |
|--|--|--|
| • КОД МАСТЕРА | • КОД РАБОТЫ | • НОМЕР ЗАКАЗА |
| • ФИО МАСТЕРА | • НАИМЕНОВАНИЕ РАБОТЫ | • ДАТА ЗАКАЗА |
| • КАТЕГОРИЯ МАСТЕРА | • СТОИМОСТЬ 1 ЧАСА РАБОТЫ | • Ф И О ЗАКАЗЧИКА |
| • ДАТА ПОСТУПЛЕНИЯ НА РАБОТУ | • ДЛИТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ РАБОТЫ (час.) | • НОМЕР МАШИНЫ |
| | | • КОД РАБОТЫ |
| | | • КОД МАСТЕРА |
| | | • ЗАТРАЧЕННОЕ ВРЕМЯ (час.) |

Запросы с вычислением полей:

- Вычислить стаж определенного мастера в годах на текущую дату. Код мастера вводится с клавиатуры.
- Рассчитать стоимость выполнения каждой работы и скидку в размере 10% от стоимости в случае задержки выполнения заказа.

Итоговый запрос:

Подсчитать количество выполненных заказов каждым мастером за определенный месяц.

Перекрестный запрос:

Подсчитать количество заказанных работ каждым заказчиком по каждому мастеру и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую стоимость работ по каждому заказчику.

ВАРИАНТ 21 База данных «АТЕЛЬЕ»

| ТАБЛИЦА 1 «КЛИЕНТЫ» | ТАБЛИЦА 2 «МАСТЕРА» | ТАБЛИЦА 3 «ЗАКАЗЫ» |
|------------------------|------------------------|------------------------------|
| • КОД КЛИЕНТА | • КОД МАСТЕРА | • № ЗАКАЗА |
| • Ф И О | • Ф И О МАСТЕРА | • КОД КЛИЕНТА |
| • АДРЕС | • ВРЕМЯ РАБОТЫ | • ДАТА ЗАКАЗА |
| • № ПАСПОРТА | • МУЖСКОЙ ИЛИ ДАМСКИЙ | • КОД МАСТЕРА |
| • ТЕЛЕФОН | | • СТОИМОСТЬ ЗАКАЗА (руб.) |
| | | • ДАТА ВЫПОЛНЕНИЯ ЗАКАЗА |

Запросы с вычислением полей:

- Подсчитать время (в днях), необходимое на выполнение заказа, номер которого вводится с клавиатуры.
- Посчитать стоимость выполнения заказа в у.е. Курс 1 у.е. вводится с клавиатуры.

Итоговый запрос:

Подсчитать общее количество заказов и их общую стоимость за каждый день определенного месяца.

Перекрестный запрос:

Подсчитать общее количество заказов, выполненных каждым мастером в разрезе дат и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую стоимость выполненных заказов каждым мастером.

ВАРИАНТ 22

База данных «УЧЕТ КАДРОВ НА ПРЕДПРИЯТИИ»

| ТАБЛИЦА 1 «ПОДРАЗДЕЛЕНИЯ» | ТАБЛИЦА 2 «ПРОФЕССИИ» | ТАБЛИЦА 3 «КАДРЫ» |
|------------------------------|--------------------------|------------------------------|
| • КОД ПОДРАЗДЕЛЕНИЯ | • КОД ПРОФЕССИИ | • КОД СОТРУДНИКА |
| • НАИМЕНОВАНИЕ ПОДРАЗДЕЛЕНИЯ | • НАИМЕНОВАНИЕ | • Ф И О СОТРУДНИКА |
| • ФОНД ОПЛАТЫ ТРУДА | • ОКЛАД (руб.) | • КОД ПОДРАЗДЕЛЕНИЯ |
| | | • КОД ПРОФЕССИИ |
| | | • ДАТА ПОСТУПЛЕНИЯ НА РАБОТУ |
| | | • ПРЕМИЯ (%) |

Запросы с вычислением полей:

- Вычислить стаж каждого сотрудника в годах на текущую дату.
- Рассчитать сумму начислений каждого сотрудника с учетом оклада и премии (% от фонда оплаты труда).

Итоговый запрос:

Подсчитать общее количество сотрудников определенного стажа, размер которого вводится с клавиатуры.

Перекрестный запрос:

Подсчитать количество сотрудников каждой профессии в каждом подразделении и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести максимальный размер фонда оплаты труда каждого подразделения.

ВАРИАНТ 23
База данных «АВИАПЕРЕВОЗКИ»

| ТАБЛИЦА 1 «САМОЛЕТЫ» | ТАБЛИЦА 2 «МАРШРУТЫ» | ТАБЛИЦА 3 «АВИАПЕРЕВОЗКИ» |
|------------------------------------|-----------------------------------|--------------------------------|
| • КОД САМОЛЕТА | • КОД МАРШРУТА | • НОМЕР ПЕРЕВОЗКИ |
| • НАИМЕНОВАНИЕ | • ПУНКТ НАЗНАЧЕНИЯ | • ДАТА ПЕРЕВОЗКИ |
| • МАРКА | • ПРОДОЛЖИТЕЛЬНОСТЬ ПОЛЕТА (час.) | • КОЛИЧЕСТВО ПРОДАННЫХ БИЛЕТОВ |
| • ГОД ВЫПУСКА | • ВРЕМЯ ОТПРАВЛЕНИЯ | • КОД САМОЛЕТА |
| • ТЕХ. СОСТОЯНИЕ | • ДОП. ИНФОРМАЦИЯ | • КОД МАРШРУТА |
| • ОБЩЕЕ КОЛИЧЕСТВО МЕСТ В САМОЛЕТЕ | | • ЦЕНА БИЛЕТА (руб.) |

Запросы с вычислением полей:

- Подсчитать общую стоимость каждой перевозки.
- Подсчитать количество свободных мест на определенный самолет, код которого вводится с клавиатуры.

Итоговый запрос:

Подсчитать количество проданных билетов и среднюю цену билета по каждому маршруту за определенный месяц.

Перекрестный запрос:

Подсчитать общую стоимость проданных билетов на каждый самолет по каждому маршруту и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать количество самолетов, задействованных на каждом маршруте.

ВАРИАНТ 24
База данных «АВТОГОНКИ»

| ТАБЛИЦА 1 «АВТОМОБИЛИ» | ТАБЛИЦА 2 «АВТОГОНЩИКИ» | ТАБЛИЦА 3 «АВТОГОНКИ» |
|---------------------------------|----------------------------------|---|
| • КОД АВТОМОБИЛЯ | • КОД АВТОГОНЩИКА | • НОМЕР АВТОГОНКИ |
| • НАИМЕНОВАНИЕ | • Ф И О | • НАЗВАНИЕ АВТОГОНКИ |
| • МАРКА | • СТРАНА | • ДАТА АВТОГОНКИ |
| • ТЕХ. СОСТОЯНИЕ | • НАЦИОНАЛЬНОСТЬ | • ПРОДОЛЖИТЬ АВТОГОНКИ (км.) |
| • ЦЕНА 1 КМ. ПРОБЕГА АВТОМОБИЛЯ | • ЦЕНА 1 ЧАСА РАБОТЫ АВТОГОНЩИКА | • КОД АВТОМОБИЛЯ |
| | • КЛАСС АВТОГОНЩИКА | • КОД АВТОГОНЩИКА |
| | | • ВРЕМЯ, ЗАТРАЧЕННОЕ НА АВТОПРОБЕГ (мин.) |

Запросы с вычислением полей:

- Вычислить стоимость каждой автогонки с учетом стоимости пробега автомобиля и стоимости работы автогонщика.
- Определить день недели, в который проводилась автогонка, и вывести текстовое название дня недели в отдельном поле.

Итоговый запрос:

Подсчитать сколько раз принимал участие во всех автогонках каждый автогонщик и суммарное время его пробега (в часах).

Перекрестный запрос:

Подсчитать суммарное расстояние, пройденное каждым автомобилем в каждой автогонке, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общее время пробега (в мин.) каждым автомобилем.

ВАРИАНТ 25

База данных «НАЧИСЛЕНИЕ СТИПЕНДИИ»

| ТАБЛИЦА 1 «СВЕДЕНИЯ» | ТАБЛИЦА 2 «ВИДЫ СТИПЕНДИЙ» | ТАБЛИЦА 3 «НАЧИСЛЕНИЯ» |
|-------------------------|-------------------------------|--------------------------------|
| • НОМЕР ЗАЧЕТКИ | • КОД СТИПЕНДИИ | • НОМЕР ПЛАТЕЖНОЙ ВЕДОМОСТИ |
| • Ф И О СТУДЕНТА | • НАИМЕНОВАНИЕ | • ДАТА НАЧИСЛЕНИЯ |
| • ДАТА РОЖДЕНИЯ | • КОЭФФИЦИЕНТ | • НОМЕР ЗАЧЕТКИ |
| • НОМЕР ГРУППЫ | | • СРЕДНИЙ БАЛЛ |
| • АДРЕС | | • КОД СТИПЕНДИИ |
| | | • ЗА МЕСЯЦ |

*** – ОБЫЧНАЯ, СОЦИАЛЬНАЯ, ИМЕННАЯ, СТИПЕНДИЯ СТАРОСТИ И Т.Д.

Запросы с вычислением полей:

- Вычислить возраст каждого студента в годах.
- Подсчитать размер стипендии по каждому студенту. Для студентов, имеющих средний балл менее 3-ех, стипендию не начислять, а для остальных – начислять как коэффициент от базовой стипендии, размер которой вводится с клавиатуры.

Итоговый запрос:

На основе предыдущего запроса подсчитать количество студентов, обучающихся на бесплатной форме обучения, а также их общую сумму стипендии за конкретный месяц.

Перекрестный запрос:

Подсчитать количество студентов в разрезе начисляемых видов стипендий с выводом результата в виде перекрестной таблицы. В итоговом столбце - максимальный средний балл по каждому виду стипендии.

ВАРИАНТ 26

База данных «ХРАНЕНИЕ В СЕЙФАХ БАНКА»

| ТАБЛИЦА 1 «КЛИЕНТЫ» | ТАБЛИЦА 2 «ИНФОРМАЦИЯ О БАНКОВСКИХ СЕЙФАХ» | ТАБЛИЦА 3 «ПРЕДОСТАВЛЕНИЕ УСЛУГ» |
|------------------------|--|--|
| • КОД КЛИЕНТА | • КОД СЕЙФА | • № ПП |
| • Ф.И.О. КЛИЕНТА | • ВИД СЕЙФА | • ДАТА ПРЕДОСТАВЛЕНИЯ СЕФА |
| • АДРЕС | • РАЗМЕР СЕЙФА | • КОД КЛИЕНТА |
| • № ПАСПОРТА | • СТОИМОСТЬ ПОЛЬЗОВАНИЯ ЗА СУТКИ (y.e.) | • КОД СЕЙФА |
| • ТЕЛЕФОН | | • СРОК ХРАНЕНИЯ (дней) |

Запросы с вычислением полей:

- Определить для каждого клиента дату окончания хранения и день недели, на который она выпадает. Текстовый формат дня недели вывести в отдельном поле.
- Вычислить стоимость хранения в рублях, курс которого вводится с клавиатуры.

Итоговый запрос:

Подсчитать, сколько раз пользовался услугами банковских сейфов каждый клиент и на какую общую стоимость (в у.е.).

Перекрестный запрос:

Подсчитать общую стоимость пользования услугами разного вида банковских сейфов каждым клиентом с выводом результата в виде перекрестной таблицы. В итоговом столбце вывести максимальную продолжительность хранения по каждому клиенту.

ВАРИАНТ 27

База данных «АВТОЗАПРАВОЧНАЯ СТАНЦИЯ»

| ТАБЛИЦА 1 «ВИДЫ ТОПЛИВА» | ТАБЛИЦА 2 «СВЕДЕНИЯ О ТРК ***» | ТАБЛИЦА 3 «ПРЕДОСТАВЛЕНИЕ УСЛУГ» |
|-------------------------------|--|--|
| • КОД ВИДА ТОПЛИВА | • НОМЕР ТРК | • НОМЕР П/П |
| • НАИМЕНОВАНИЕ | • ВОЗМОЖНЫЕ ВИДЫ ТОПЛИВА | • ДАТА ЗАПРАВКИ |
| • ПОСТАВЩИК | • ПРОПУСКНАЯ СПОСОБНОСТЬ (литр/сек) | • НОМЕР ТРК |
| • СТОИМОСТЬ 1 ЛИТРА (руб.) | • РЕЖИМ РАБОТЫ | • КОД ВИДА ТОПЛИВА |
| | • ТЕКУЩЕЕ СОСТОЯНИЕ | • КОЛИЧЕСТВО ВЫДАН- НОГО ТОПЛИВА (литр) |
| | | • ВИД ОПЛАТЫ |

*** – Топливо-раздаточные комплексы (колонки)

Запросы с вычислением полей:

- Вычислить для каждой предоставленной услуги стоимость оплаты.
- Посчитать на основе предыдущего запроса стоимость оплаты в у.е., курс у.е. вводится с клавиатуры.

Итоговый запрос:

Подсчитать, сколько раз воспользовались клиенты каждым ТРК и на какой общий объем топлива за каждую дату заправки, попадающую на воскресенье.

Перекрестный запрос:

Подсчитать, сколько раз на каждом ТРК был реализован каждый отдельный вид топлива, и вывести результат в виде перекрестной таблицы. В итоговом столбце вывести общую стоимость выданного топлива по каждому ТРК.

3. Методические рекомендации к выполнению контрольной работы

3.1. Методические рекомендации к выполнению задания № 1

Ответ на вопрос представить в виде напечатанного текста в текстовом редакторе Word. Объем ответа не более 2-ух страниц печатного текста.

3.2. Методические рекомендации к выполнению задания № 2

Выполнение контрольной работы будет рассмотрено с использованием СУБД Microsoft Access 2000 на примере базы данных «Сервис – Центр» (предприятие, занимающееся ремонтом бытовой и компьютерной техники). Указанная БД имеет следующую структуру:

| ТАБЛИЦА1 «УСЛУГИ» | ТАБЛИЦА2 «КЛИЕНТЫ» | ТАБЛИЦА3 «ЗАКАЗЫ» |
|--|------------------------|---------------------------|
| • КОД УСЛУГИ | • КОД КЛИЕНТА | • НОМЕР ЗАКАЗА |
| • НАИМЕНОВАНИЕ УСЛУГИ | • НАИМЕНОВАНИЕ КЛИЕНТА | • КОД УСЛУГИ |
| • СТОИМОСТЬ УСЛУГИ | • АДРЕС КЛИЕНТА | • КОД КЛИЕНТА |
| • КОЛИЧЕСТВО ДНЕЙ ДЛЯ ВЫПОЛНЕНИЯ УСЛУГИ | • ТЕЛЕФОН | • ДАТА ПРИНЯТИЯ ЗАКАЗА |
| | | • ОПЛАЧЕНО |

Запросы с вычислением полей:

- Вычислить стоимость услуги в у.е., курс которой вводится с клавиатуры.
- Определить дату выполнения заказа, а также день недели выполнения заказа.

Итоговый запрос:

Сколько всего заказов сделал каждый клиент и на какую сумму?

Перекрестный запрос:

Подсчитать по каждому клиенту общую стоимость услуг в разрезе дат и вывести результат в виде перекрестной таблицы. В итоговом столбце подсчитать среднюю стоимость услуг по каждому клиенту.

3.2.1. Создание файла БД

- Запустить СУБД Microsoft Access;
- в окне Microsoft Access установить переключатель в положение Новая БД и нажать **ОК** (см. рис 3.1.);

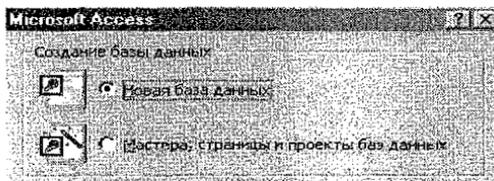


Рис. 3.1. Создание файла новой БД

- в появившемся окне **Файл новой БД** указать, где создать файл (на рабочем диске в папке **Мои документы** или на другом диске), с каким именем (например, **ЗАКАЗЫ**), и нажать кнопку **Создать**.

3.2.2. Создание таблиц

Для создания первой таблицы Услуги необходимо выполнить следующие действия:

1. Находясь в окне БД на вкладке Таблицы, нажать **Создать**;
2. В окне Новая таблица выбрать **Конструктор** и нажать **ОК**;
3. В окне **Конструктора** таблицы заполнить последовательно поля: **Имя поля**, **Тип поля** и, если нужно, поле **Описание**. Тип поля и свойства поля определяем произвольно по смыслу.

Замечание. Изменение впоследствии типов данных в полях может привести к искажению внесенной в эти поля информации.

4. Для того, чтобы поле **Код услуги** сделать **ключевым**, необходимо:
 - выделить поле **Код услуги**;
 - щелкнуть на панели инструментов кнопку **Ключевое поле**.В результате в окне конструктора получим результат :

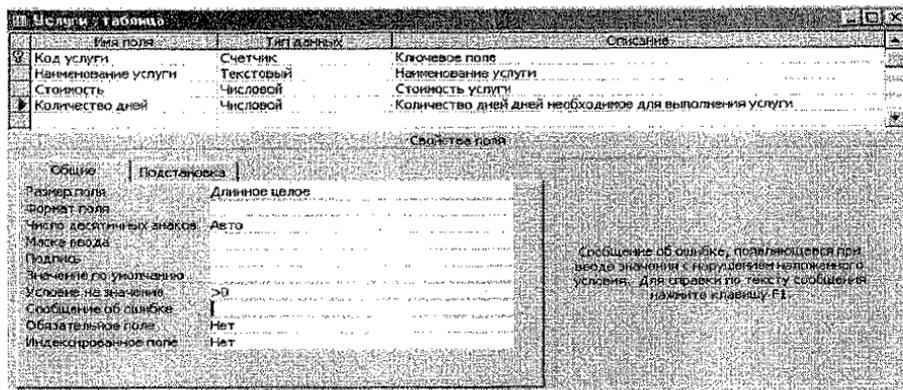


Рис. 3.2. Таблица Услуги в режиме Конструктора

5. Закрыть окно таблицы, сохраняя созданную таблицу с именем **Услуги**. Аналогично создать таблицы: **Клиенты** и **Заказы**. Вид режима **Конструктора** для данных таблиц представлен на рис. 3.3. и рис. 3.4.

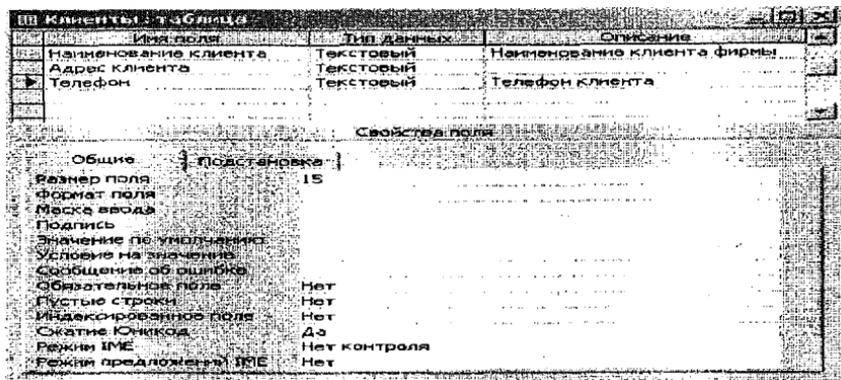


Рис. 3.3. Таблица Клиенты в режиме Конструктора

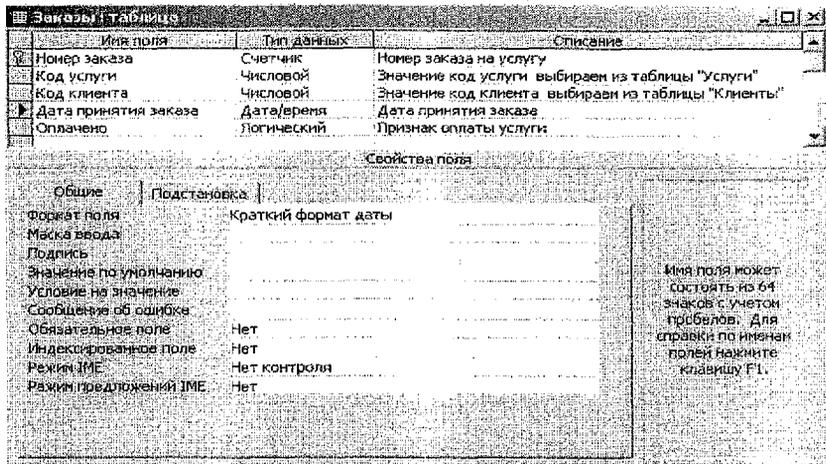


Рис. 3.4. Таблица Заказы в режиме Конструктора

Следует обратить внимание на то, что данные в поля **Код клиента** и **Код услуги** берутся в таблицу **Заказы** из соответствующих полей таблиц **Клиенты** и **Услуги**. Для того, чтобы заполнение этих полей в таблице **Заказы** было удобным, необходимо выполнить следующие действия:

- открыть таблицу **Заказы** в режиме **Конструктора**;
- установить курсор в строку **Код услуги** в поле **Тип данных**;
- из раскрывающегося списка выбрать **Мастер подстановок** (рис. 3.5).

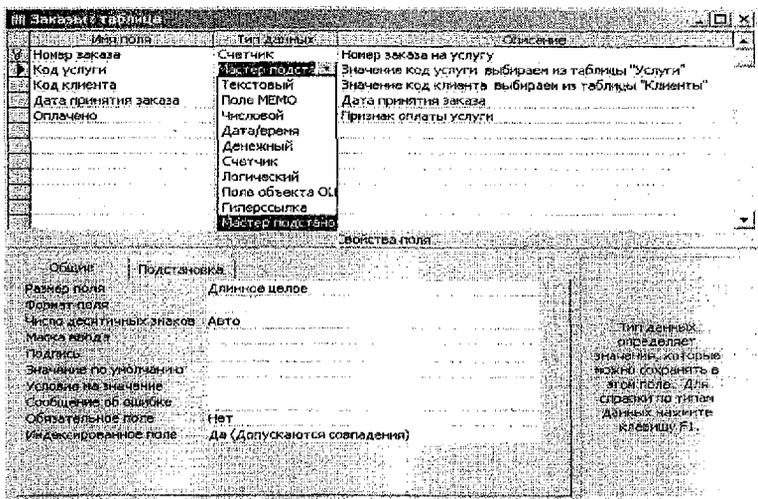


Рис. 3.5. Использование Мастера подстановок

Последовательность шагов **Мастера подстановки**:

1. В первом диалоговом окне **Создание подстановки**, необходимо выбрать, каким способом подстановки будет получать значения для выбора. В данном случае можно оставить вариант по умолчанию, т. е. выбор из таблицы или запроса. Нажать кнопку **Далее** (рис. 3.6).

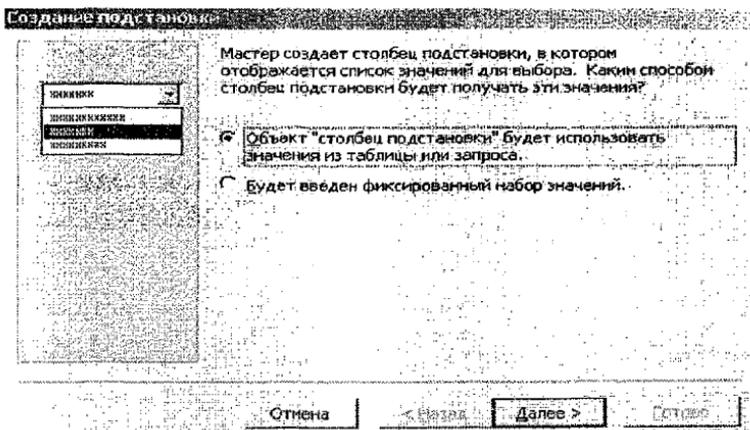


Рис. 3.6. Первый шаг **Мастера подстановки**

2. Во втором диалоговом окне **Создание подстановки** следует выбрать конкретную таблицу **Услуги** (рис. 3.7.).

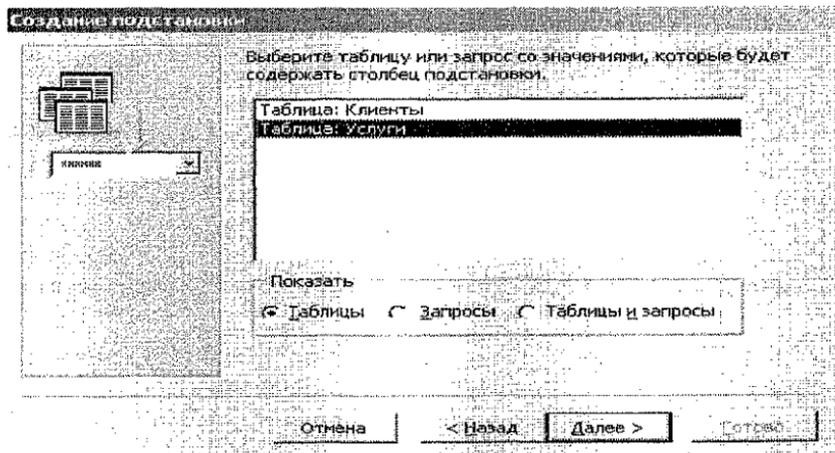


Рис. 3.7. Второй шаг **Мастера подстановки**

3. В третьем диалоговом окне необходимо отобразить отдельные поля таблицы с помощью кнопки **>** (либо все поля таблицы с помощью кнопки **>>**). В данном случае выбираются поля **Код услуги** и **Наименование услуги** (рис. 3.8.).

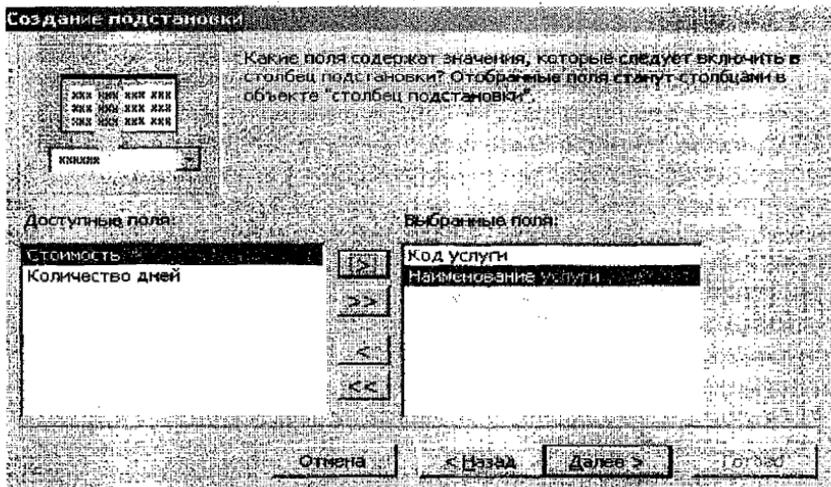


Рис. 3.8. Третий шаг Мастера подстановок

4. В следующем диалоговом окне все параметры можно оставить без изменения (рис. 3.9.).

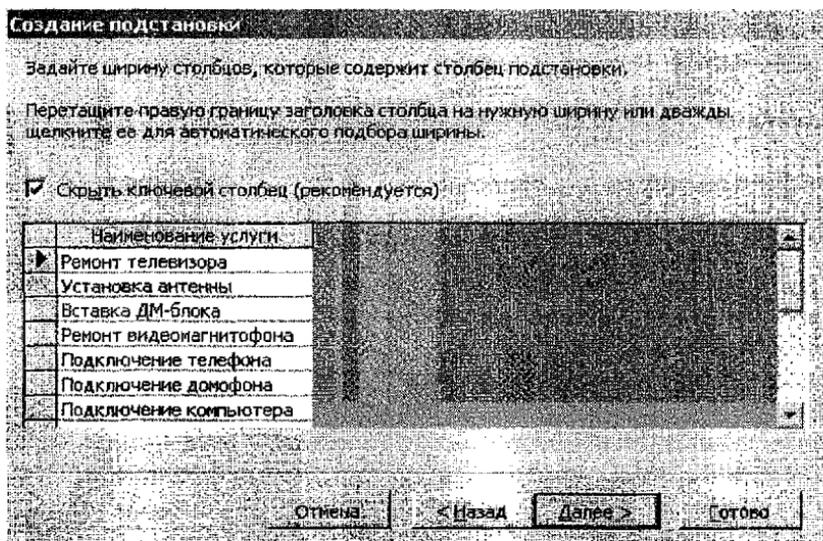


Рис. 3.9. Четвертый шаг Мастера подстановок

5. В последнем диалоговом окне необходимо ввести подпись к столбцу подстановки, например, Код услуги (рис. 3.10.) и нажать кнопку Готово.

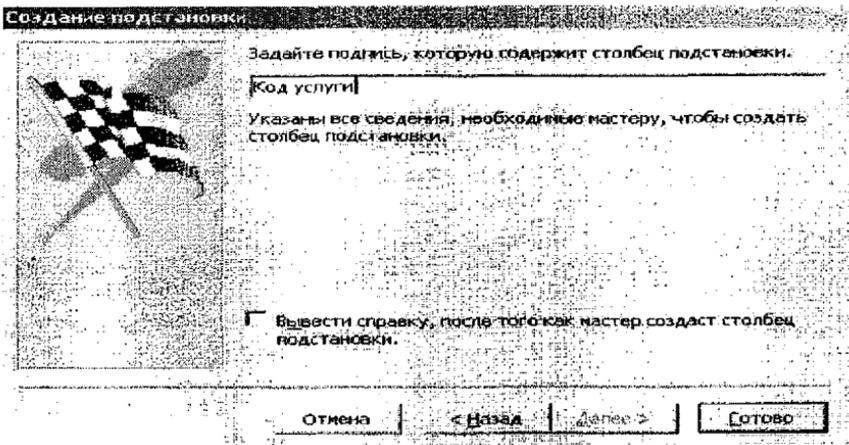


Рис. 3.10. Пятый шаг Мастера подстановок

Для поля Код клиента проделать ту же последовательность шагов, что и для поля Код услуги. Сохранить структуру таблицы Заказы и можно приступить к созданию связей между таблицами.

3.2.3. Создание связей между таблицами

Для определения связей нужно:

- 1) находясь в окне БД закрыть все объекты;
- 2) выбрать команду **Сервис/Схема данных** или нажать одноименную кнопку на панели инструментов (если связи определяются **ВПЕРВЫЕ**, то появляется окно **Схема данных**, а поверх него – окно **Добавление таблицы**);
- 3) в окне **Добавление таблицы** необходимо последовательно выделять каждую из таблиц и нажимать кнопку **Добавить** (см. рис. 3.11.).

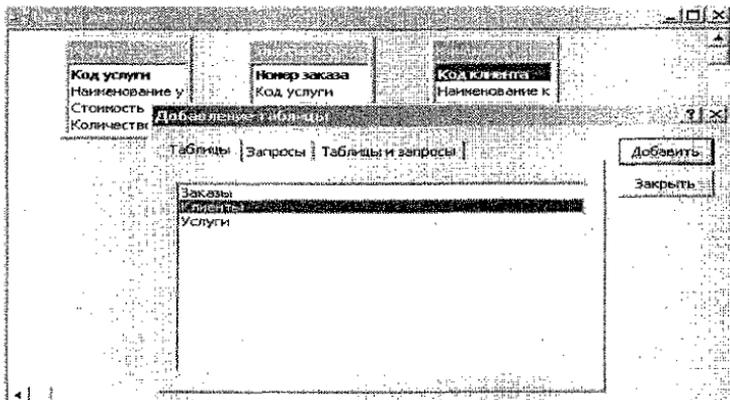


Рис. 3.11. Добавление таблиц в окно Схема данных

- 4) после выбора участвующих в связях таблиц, нажать кнопку **Заккрыть**, тем самым активизировать окно **Схема данных**;
- 5) перетащить левой кнопкой мыши *ключевое* поле *главной* таблицы на одноименное поле в *подчиненной* таблице (поле **Код услуги** – из таблицы **Услуги** в таблицу **Заказы**, поле **Код клиента** – из таблицы **Клиенты** в таблицу **Заказы**)
- 6) в диалоговом окне **Изменение связей** (рис. 3.12.) установить флажок **обеспечение целостности данных**
- 7) при необходимости, в этом же окне можно установить *каскадные операции*:
 - каскадное обновление связанных полей**
 - каскадное удаление связанных полей**;

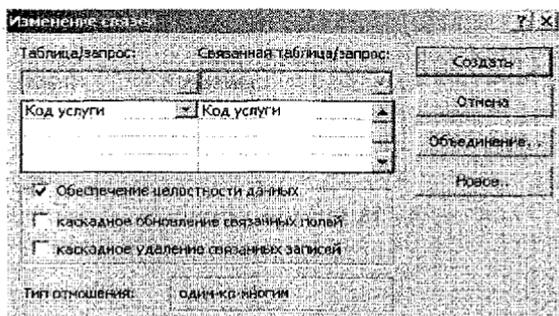


Рис. 3.12. Диалоговое окно **Изменение связей**

- 8) нажать кнопку **Создать**;
- 9) закрыть окно **Схема данных**, при этом на вопрос **Сохранить изменения макета Схема данных?** ответить – «Да».

Примечание. Если по какой либо причине окно **Добавление таблицы** не открылось, то его можно открыть по команде **Связи/Добавить таблицу** или выбрать одноименный пункт меню из контекстно-зависимого меню (щелчок правой клавиши мыши).

Замечание. При использовании **Мастера подстановок** по ходу создания таблиц, **Схема данных** автоматически заполняется таблицами, участвующими в формировании списка значений вводимых данных, а также автоматически устанавливаются между ними линии связи. Однако такие линии связи целостностью не обеспечены, поэтому их необходимо изменить.

Для этого (см. рис. 3.13.):

- находясь в окне **Схема данных**, активизировать левой кнопкой мыши имеющуюся линию связи;
- из контекстно-зависимого меню выбрать команду **Изменить связь...**;

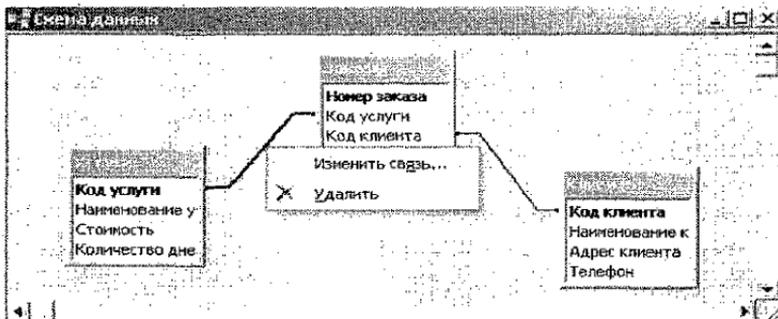


Рис. 3.13. Изменение линий связи в окне **Схема данных**

- выполнять пп. 6 – 9 раздела «Создание связей между таблицами» (см. методический материал выше).

3.2.4. Печать схемы данных

Для вывода на печать полученной схемы БД можно воспользоваться командой **Сервис/Анализ/Архивариус** следующим образом:

- находясь в окне БД, выбрать команду **Сервис/Анализ/Архивариус**;
- на вкладке **Текущая база данных** установить флажок **Связи**;
- нажать **ОК**;
- выбрать команду **Файл/Печать**.

Если вышеуказанная команда недопустима, то печать схемы БД можно осуществить посредством стандартного приложения Windows – графического редактора **Paint**, выполнив следующие действия:

- находясь в окне БД, выбрать команду **Сервис/Схема данных**;
- копию экрана поместить в буфер обмена, нажав клавишу **PrintScreen**;
- загрузить графический редактор **Paint**;
- вставить содержимое буфера обмена;
- выделить только схему БД при помощи кнопки выделения прямоугольных фрагментов и скопировать ее в буфер обмена;
- очистить холст с помощью команд: **Правка/Выделить все**, **Правка/Очистить выделение**;
- вставить схему БД из буфера и напечатать ее по команде **Файл/Печать**.

В результате получим схему данных, представленную на рис. 3.14.



Рис. 3.14. Схема данных

3.2.5. Заполнение таблиц данными

Для заполнения таблиц данными необходимо:

- в окне БД выделить необходимую таблицу, например, таблицу *Услуги*;
- щелкнуть кнопку *Открыть* на панели инструментов окна БД или двойным щелчком по выделенной таблице;
- заполнить таблицу 10 записями.

Аналогично заполняется таблица *Клиенты* 10 записями, а таблица *Заказы* должна быть заполнена 15 записями.

3.2.6. Печать заполненных таблиц

- в окне БД выделить таблицу *Услуги*;
- осуществить ее предварительный просмотр по команде *Файл / Предварительный просмотр*;
- напечатать таблицу по команде *Файл / Печать*.

Аналогично выполняется печать таблиц *Клиенты* и *Заказы*.

В результате получим следующие распечатки заполненных таблиц:

Услуги

| Код услуги | Наименование услуги | Стоимость | Количество дней |
|------------|---------------------------|-----------|-----------------|
| 1 | Ремонт телевизора | 15000 | 4 |
| 2 | Установка антенны | 8000 | 2 |
| 3 | Вставка ДМ-блока | 3000 | 1 |
| 4 | Ремонт видеомэгагнитофона | 20000 | 7 |
| 5 | Подключение телефона | 7000 | 2 |
| 6 | Подключение домофона | 13000 | 3 |
| 7 | Подключение компьютера | 17000 | 2 |
| 8 | Диагностика компьютера | 12000 | 5 |
| 9 | Монтаж ЛВС | 18000 | 0 |
| 10 | Подключение модема | 4000 | 1 |

Клиенты

| Код клиента | Наименование клиента | Адрес клиента | Телефон |
|-------------|----------------------|-------------------|---------|
| 1 | ПКФ "Марс" | Советская, 34 | 345-678 |
| 2 | БрГТУ | Московская, 267 | 456-678 |
| 3 | ПКП "Сож" | Речная, 45 | 876-123 |
| 4 | ЗАО "Альбатрос" | Вишневая, 78 | 123-567 |
| 5 | ПКП "Мур-Мур" | Пограничников, 39 | 345-129 |
| 6 | Магазин "Южный" | Южная, 56 | 789-342 |
| 7 | ЦУМ | Московская, 47 | 120-703 |
| 8 | АПК "Западный Буг" | Лесная, 67 | 234-009 |
| 9 | ООО "Санта" | Гоголя, 123 | 504-034 |
| 10 | ПКЦ "Геркулес" | Вокзальная, 67 | 423-786 |

Заказы

| Номер заказа | Код услуги | Код клиента | Дата принятия заказа | Оплачено |
|--------------|------------------------|-----------------|----------------------|----------|
| 1 | Ремонт телевизора | ЗАО "Альбатрос" | 1/15/2005 | Да |
| 3 | Подключение компьютера | ЦУМ | 1/15/2005 | Да |
| 4 | Диагностика компьютера | БрГТУ | 1/15/2005 | Нет |
| 5 | Установка антенны | ПКП "Сож" | 3/19/2005 | Да |
| 6 | Подключение телефона | БрГТУ | 3/19/2005 | Да |
| 7 | Диагностика компьютера | ЦУМ | 3/23/2005 | Да |
| 8 | Подключение домофона | ПКП "Мур-Мур" | 3/23/2005 | Нет |
| 9 | Вставка ДМ-блока | АПК "Зап Буг" | 4/21/2005 | Да |
| 10 | Ремонт видеомэгнофона | ПКФ "Марс" | 4/21/2005 | Да |
| 11 | Подключение компьютера | ООО "Санта" | 5/12/2005 | Да |
| 12 | Диагностика компьютера | ООО "Санта" | 5/12/2005 | Да |
| 13 | Подключение модема | БрГТУ | 5/25/2005 | Да |
| 14 | Подключение телефона | БрГТУ | 5/25/2005 | Нет |
| 15 | Подключение модема | ЦУМ | 5/25/2005 | Да |

3.3. Методические рекомендации к выполнению задания № 3

3.3.1. Создание запросов в режиме Конструктора

Для вызова Конструктора запросов необходимо выполнить последовательность действий:

- 1) находясь в окне БД на вкладке **Запросы**, нажать кнопку **Создать**;
- 2) в появившемся диалоговом окне **Новый запрос** выбрать *Конструктор* и нажать **ОК**;
- 3) в окне **Добавление таблицы** на вкладке **Таблицы** выделять имена таблиц, участвующих в запросе и нажимать кнопку **Добавить**, после чего нажать кнопку **Закреть**.

Примечание. Если по какой-либо причине окно **Добавление таблицы** не открылось, то его можно открыть по команде **Запрос/Добавить таблицу**.

В результате появится окно Конструктора запроса, в верхней панели которого отображаются списки полей таблиц, а в нижней – бланк запроса.

Для рассматриваемого варианта БД два запроса выбора формулируются следующим образом:

Запрос 1. Из таблицы **Услуги** выбрать сведения об услугах с разным количеством дней выполнения (использование логического оператора ИЛИ). Количество дней вводится с клавиатуры.

Запрос 2. Из таблицы **Заказы** выбрать сведения о заказах, которые **оплачены** определенным клиентом (использование логического оператора И). Код клиента вводится с клавиатуры.

Для создания запросов необходимо:

- сначала перетащить участвующие в запросе поля из списков полей таблиц в верхней панели в строку **Поле** нижней панели бланка Конструктора запроса;
- затем задать условия отбора записей;

- У сформировав запрос, выполнить его с помощью команды **Запрос/Запуск** или соответствующей кнопки на панели инструментов;
- У закрывая окно запроса, выполнить его сохранение с заданным именем;
- У напечатать результат созданных запросов из окна базы данных по команде **Файл/Печать**.

Пример записи условия отбора в случае первого запроса приведен на рис. 3.15.

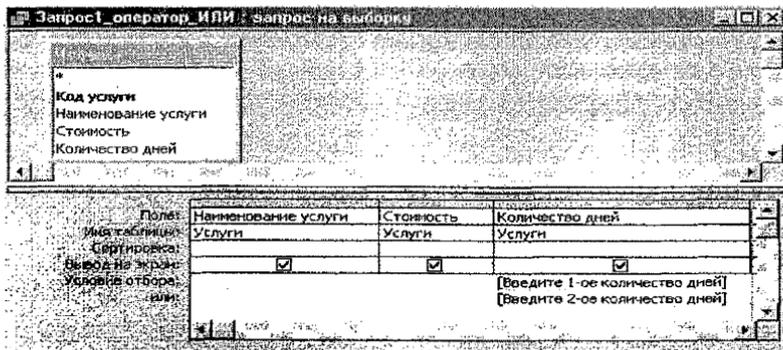


Рис. 3.15. Бланк запроса **Запрос1_оператор_ИЛИ**

В результате получится следующая выборка:

Запрос1_оператор_ИЛИ

| Наименование услуги | Стоимость | Количество дней |
|------------------------|-----------|-----------------|
| Установка антенны | 8000 | 2 |
| Вставка ДМ-блока | 3000 | 1 |
| Подключение телефона | 7000 | 2 |
| Подключение компьютера | 17000 | 2 |
| Подключение модема | 4000 | 1 |

Пример записи условия отбора в случае второго запроса приведен на рис. 3.16.

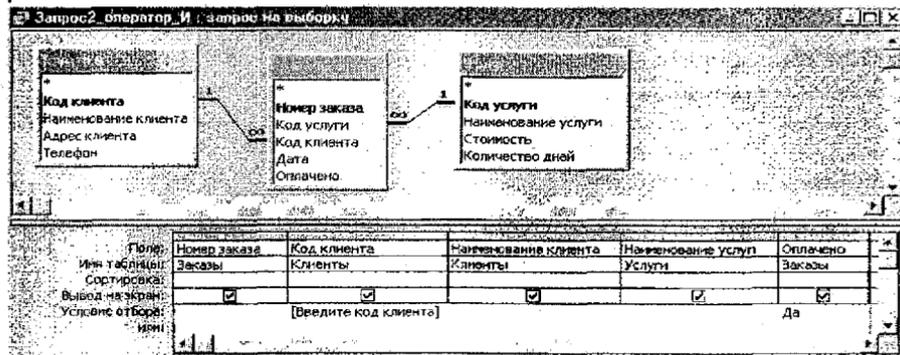


Рис. 3.16. Бланк запроса для **Запрос2_оператор_И**

В результате получится следующая выборка:

Запрос2_оператор_И

| Номер заказа | Код клиента | Наименование клиента | Наименование услуги | Оплачено |
|--------------|-------------|----------------------|----------------------|----------|
| 6 | 2 | БГТУ | Подключение телефона | Да |
| 13 | 2 | БГТУ | Подключение модема | Да |

Пример записи условия отбора в случае третьего запроса с вычисляемыми полями по условию задания (*вычислить стоимость услуги в у.е., курс которой вводится с клавиатуры*) приведен на рис. 3.17.

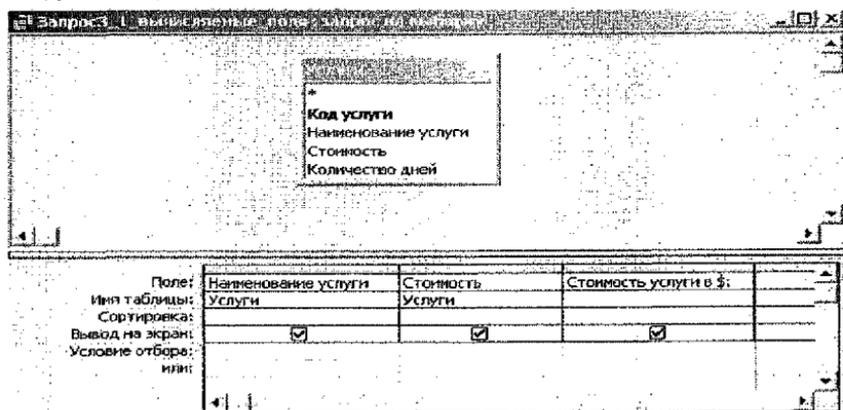


Рис.3.17. Бланк запроса для *Запрос3_1_вычисляемые_поля*

Выражение, построенное в построителе выражений для вычисляемого поля, следующее:

Стоимость услуги в \$: [Услуги]![Стоимость] / [Введите курс \$]

Замечание. Имена вычисляемых полей (*Выражение 1, Выражение 2 и т.д.*) можно изменять по смыслу запроса или в соответствии с условием задания.

В результате получится следующая выборка:

Запрос3_1_вычисляемые_поля

| Наименование услуги | Стоимость | Стоимость услуги в \$ |
|------------------------|-----------|-----------------------|
| Ремонт телевизора | 15000 | 7.00 |
| Установка антенны | 8000 | 3.74 |
| Вставка ДМ-блока | 3000 | 1.40 |
| Ремонт видеоманитфона | 20000 | 9.34 |
| Подключение телефона | 7000 | 3.27 |
| Подключение домофона | 13000 | 6.07 |
| Подключение компьютера | 17000 | 7.94 |
| Диагностика компьютера | 12000 | 5.61 |
| Монтаж ЛВС | 18000 | 8.41 |
| Подключение модема | 4000 | 1.87 |

Пример записи условия отбора в случае четвертого запроса с вычисляемыми полями по условию задания (рассчитать дату выполнения заказа и день недели выполнения заказа) приведен на рис. 3.18.

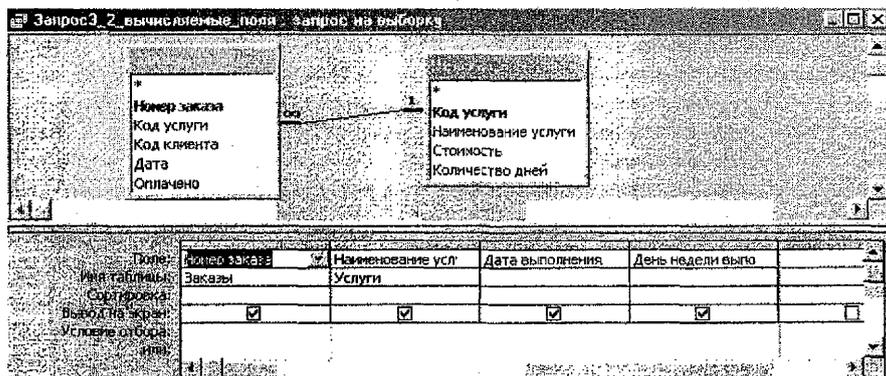


Рис.3.18. Бланк запроса для **Запрос3_2_вычисляемые_поля**

Выражения, построенные в построителе выражений для вычисляемых полей, следующие:

Дата выполнения заказа:

[Заказы].[Дата принятия заказа] + [Услуги].[Количество дней]

День недели выполнения заказа: Format([Заказы].[Дата принятия заказа] + [Услуги].[Количество дней], "dddd")

В результате получится следующая выборка:

| Запрос3_2_вычисляемые_поля | | | |
|----------------------------|--------------------------|------------------------|-------------------------------|
| Номер заказа | Наименование услуги | Дата выполнения заказа | День недели выполнения заказа |
| 1 | Ремонт телевизора | 19.01.2005 | среда |
| 5 | Установка антенны | 21.03.2005 | понедельник |
| 9 | Вставка ДМ-блока | 22.04.2005 | пятница |
| 10 | Ремонт видеомэагнитофона | 28.04.2005 | четверг |
| 6 | Подключение телефона | 21.03.2005 | понедельник |
| 14 | Подключение телефона | 27.05.2005 | пятница |
| 8 | Подключение домофона | 26.03.2005 | суббота |
| 3 | Подключение компьютера | 17.01.2005 | понедельник |
| 11 | Подключение компьютера | 14.05.2005 | суббота |
| 4 | Диагностика компьютера | 20.01.2005 | четверг |
| 7 | Диагностика компьютера | 28.03.2005 | понедельник |
| 12 | Диагностика компьютера | 17.05.2005 | вторник |
| 13 | Подключение модема | 26.05.2005 | четверг |
| 15 | Подключение модема | 26.05.2005 | четверг |

Пример записи условия отбора в случае итогового запроса приведен на рис. 3.19.

Поле: Код клиента Номер заказа Стоимость
 Имя таблицы: Заказы Заказы Услуги
 Групповая операция: Группировка Count Sum
 Сортировка:
 Вывод на экран:
 Условия отбора:
 или:

Рис.3.19. Бланк запроса для *Запрос4_итоговый*

В результате получится следующая выборка:

Запрос4_итоговый

| Наименование клиента | Общее количество заказов | Суммарная стоимость услуг |
|----------------------|--------------------------|---------------------------|
| ПКФ "Марс" | 1 | 20000 |
| БрГТУ | 4 | 30000 |
| ПКП "Сож" | 1 | 8000 |
| ЗАО "Альбатрос" | 1 | 15000 |
| ПКП "Мур-Мур" | 1 | 13000 |
| ЦУМ | 3 | 33000 |
| АПК "Западный Буг" | 1 | 3000 |
| ООО "Санта" | 2 | 29000 |

Пример записи условия отбора в случае перекрестного запроса приведен на рис. 3.20.

Поле: Наименование клиента Дата Стоимость Стоимость
 Имя таблицы: Клиенты Заказы Услуги Услуги
 Групповая операция: Группировка Группировка Sum Avg
 Перекрестная таблица: Заголовки строк Заголовки столбцов Значение Заголовки строк
 Сортировка:
 Условия отбора:
 или:

Рис. 3.20. Бланк запроса для *Запрос5_перекрестный*

Замечания. 1. Итоговый столбец, в котором вычисляется среднее значение стоимости услуг, по умолчанию отображается в результирующей таблице на втором месте. Для того, чтобы переместить его в конец таблицы, необходимо выделить этот столбец и перетащить за заголовок.

2. Для того, чтобы изменить подпись итогового столбца, необходимо, находясь в режиме *Конструктора* запроса, вызвать окно **Свойства поля** (щелкнуть кнопкой мыши в строке **Поле** бланка запроса для итогового столбца и вызвать команду **Вид/Свойства**) и ввести в строке **Подпись** необходимое текстовое выражение (в рассматриваемом примере это – **Средняя стоимость**).

В результате получится следующая выборка:

Запрос5_перекрестный

| Наименование клиента | 15.01.05 | 19.03.05 | 23.03.05 | 21.04.05 | 12.05.05 | 25.05.05 | Средняя стоимость |
|----------------------|----------|----------|----------|----------|----------|----------|-------------------|
| АПК "Западный Буг" | | | | 3000 | | | 3000 |
| БрГТУ | 12000 | 7000 | | | | 11000 | 7500 |
| ЗАО "Альбатрос" | 15000 | | | | | | 15000 |
| ООО "Санта" | | | | | 29000 | | 14500 |
| ПКП "Мур-Мур" | | | 13000 | | | | 13000 |
| ПКП "Сож" | | 8000 | | | | | 8000 |
| ПКФ "Марс" | | | | 20000 | | | 20000 |
| ЦУМ | 17000 | | 12000 | | | 4000 | 11000 |

3.4. Методические рекомендации к выполнению задания № 4

3.4.1. Создание формы с помощью Мастера форм

Последовательность шагов:

- в окне БД на вкладке **Формы** нажать кнопку **Создать**;
- в окне **Новая форма** задать режим **Мастер форм**;
- выбрать из раскрывающегося списка в качестве источника данных для формы таблицу с именем **ЗАКАЗЫ** и нажать **ОК**;
- в окне **Создание форм** Мастер потребует:
 - на 1-м шаге – выбора полей для формы посредством кнопок с изображением знака **>>** – всех полей и знака **>** – отдельных полей;
 - на 2-м шаге – выбора внешнего вида формы (например, *в один столбец*);
 - на 3-м шаге – выбора стиля для формы (например, *стандартный*);
 - на 4-м шаге – задания имени формы (*Заказы*).

В результате получим вид формы для одной записи, представленный на рис 3.21.

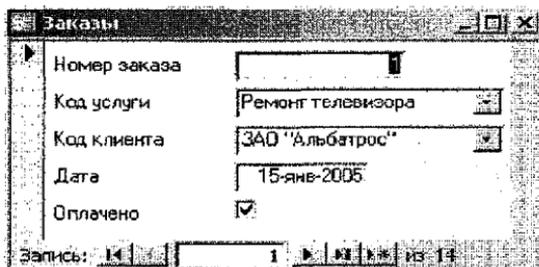


Рис. 3.21. Форма «Заказы»

Можно просмотреть записи по созданной форме, используя кнопки внизу окна формы.

Далее форма редактируется – необходимо добавить вычисляемые поля, в которых следует разместить название месяца и название дня недели для поля типа Дата/Время.

Для этого нужно перевести форму в режим *Конструктора* с помощью команды **Конструктор** п. м. **Вид**, затем открыть панель элементов (соответствующая команда в п. м. **Вид**). Далее выполнить следующее:

- 1) увеличить размеры окна Конструктора;
- 2) увеличить размеры области данных формы с помощью стрелок ,  на границах области;
- 3) передвинуть, если необходимо, поля формы, для чего сначала выделить поле (щелкнуть по нему кнопкой мыши), затем подвести указатель к границе до появления «ладошки», зажать кнопку мыши, перетащить в другое место и отпустить;
- 4) на панели элементов выбрать элемент **Поле** **abi** (щелкнуть на нем кнопкой мыши);
- 5) начертить с помощью мыши прямоугольник в области данных отчета (можно рядом с полем Дата/Время) – в результате появится выделенное поле вместе с надписью;

- 6) из п. м. **Вид** выбрать команду **Свойства**;
- 7) на вкладке **Все** появившегося окна свойств в строке **Данные** вызвать **Построитель выражений** с помощью кнопки **...**;

- 8) сформировать вычисляемое поле, используя встроенную функцию **Format** из категории **Текстовые** (см. рис 3.22) и нажать **ОК**;

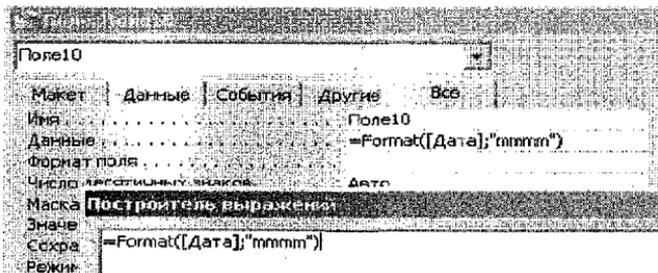


Рис. 3.22. Построение вычисляемого поля в форме «Заказы»

9) не закрывая окна свойств, выделить **надпись** поля (в результате в окне свойств появятся свойства надписи) и в строке **Подпись** на вкладке **Все** ввести название новому полю (в нашем случае, *Месяц даты заказа*) (см. рис 3.23);

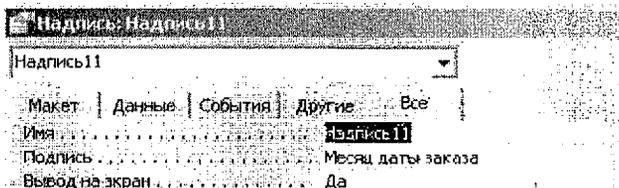


Рис. 3.23. Изменение надписи вычисляемого поля

10) закрыть окно свойств и откорректировать ширину поля и надписи по содержимому;

11) второе вычисляемое поле (*название дня недели*) создать аналогично предыдущему (пп. 4 – 10), изменив, соответственно, аргументы функции **Format**;

12) сохранить изменения командой **Файл/Сохранить**;

13) перейти в режим **Формы** (п. м. **Вид**) и просмотреть записи, используя кнопки внизу окна формы.

Для вывода на печать формы для одной записи выполняется команда **Файл/Печать** и в окне **Печать** устанавливается переключатель в положение **Выделенные записи**, после чего нажимается **ОК**.

В результате получим следующий вид отредактированной формы для первой записи, представленный на рис 3.24.

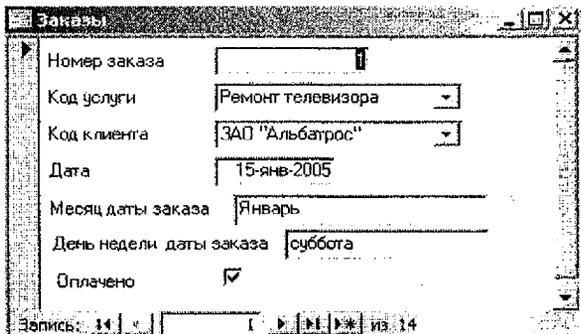


Рис. 3.24. Отредактированная форма «Заказы»

3.4.2. Создание отчета с помощью Мастера отчетов

Последовательность шагов:

1) в окне **БД** на вкладке **Отчеты** нажать кнопку **Создать**;

2) в окне **Новый отчет** задать режим **Мастер отчетов**;

- 3) выбрать из раскрывающегося списка в качестве источника данных для отчета таблицу **Заказы** и нажать **ОК**;
- 4) в окне **Создание отчетов** Мастер потребует:
 - на 1-м шаге – выбора полей для отчета (кроме полей подчиненной таблицы **Заказы** можно добавить числовое поле **Стоимость** из таблицы **Услуги**, по которому будут подводиться итоги);
 - на 2-м шаге – указания поля, по которому должна осуществляться группировка записей в отчете (в нашем случае – это поле **Код услуги**);
 - на 3-м шаге – выбора порядка сортировки в группах и задания итогов по группам, используя кнопку **Итоги** (в нашем случае итоги подведем по полю **Стоимость**, выбрав итоговую функцию **SUM**);
 - на 4-м шаге – выбора макета отчета (например, *ступенчатый*);
 - на 5-м шаге – выбора стиля для отчета (например, *строгий*);
 - на 6-м шаге – задания имени отчету (**Заказы**).

Можно просмотреть созданный отчет, используя полосы прокрутки.

Далее отчет редактируется – необходимо добавить в заголовок отчета **ФИО студента и группу**. Для этого нужно перевести отчет в режим **Конструктора** с помощью команды **Вид/Конструктор**, затем открыть панель элементов (соответствующая команда в п. м. **Вид**). Далее выполнить следующее:

- 1) на панели элементов выбрать элемент **Надпись** **Aa**;
- 2) начертить с помощью мыши прямоугольник в заголовке отчета и напечатать нужный текст.

Полученный отчет необходимо распечатать, выполнив команду **Файл/Печать**.

В результате получится отчет, представленный на рис. 3.25.

3.4.3. Создание управляющей формы

1. Находясь в окне БД на вкладке **Формы** нажать кнопку **Создать**.
2. В окне **Новая форма** задать режим **Конструктор** и, не указывая источника данных, нажать кнопку **ОК**.
3. В окне **Форма** включить панель элементов (если она не включена), выполнив команду **Вид/Панель элементов**.
4. На **Панели элементов** активизировать кнопку **Мастера**.
5. Для создания первой кнопки управляющей формы выбрать элемент **Кнопка** на **Панели элементов** и начертить с помощью мыши прямоугольник в области данных. Загрузится **Мастер** создания кнопок.
6. В окне **Создание кнопок** выполнить следующие действия:
 - на 1-м шаге выбрать из списка **Категории** – **Работа с формой**, а из списка **Действия** – **Открытие формы**;
 - на 2-м шаге установить флажок в положение **текст** и ввести текст, помещаемый на кнопку – **Открытие Формы ЗАКАЗЫ**.

| Код услуги | Номер заказа | Код клиента | Дата | Оплачено | Стоимость |
|---|--------------|--------------------|----------------|-------------------------------------|---------------------|
| 1 | 1 | ЗАО "Альбатрос" | 15-января-2005 | <input checked="" type="checkbox"/> | 15000 |
| Итого для 'Код услуги' = 1 (1 запись) | | | | | Sum 15000 |
| 2 | 5 | ПНТ "Сож" | 19-марта-2005 | <input checked="" type="checkbox"/> | 8000 |
| Итого для 'Код услуги' = 2 (1 запись) | | | | | Sum 8000 |
| 3 | 9 | АПК "Западный Буя" | 21-апреля-2005 | <input checked="" type="checkbox"/> | 3000 |
| Итого для 'Код услуги' = 3 (1 запись) | | | | | Sum 3000 |
| 4 | 10 | ПКФ "Марс" | 21-апреля-2005 | <input checked="" type="checkbox"/> | 20000 |
| Итого для 'Код услуги' = 4 (1 запись) | | | | | Sum 20000 |
| 5 | 14 | БГУ | 25-мая-2005 | <input type="checkbox"/> | 7000 |
| | | БГУ | 19-марта-2005 | <input checked="" type="checkbox"/> | 7000 |
| Итого для 'Код услуги' = 5 (2 записей) | | | | | Sum 14000 |
| 6 | 8 | ПНТ "Мур-Мур" | 23-марта-2005 | <input type="checkbox"/> | 13000 |
| Итого для 'Код услуги' = 6 (1 запись) | | | | | Sum 13000 |
| 7 | 11 | ООО "Санта" | 12-мая-2005 | <input checked="" type="checkbox"/> | 17000 |
| | | ЦУМ | 15-января-2005 | <input checked="" type="checkbox"/> | 17000 |
| Итого для 'Код услуги' = 7 (2 записей) | | | | | Sum 34000 |
| 8 | 12 | ООО "Санта" | 12-мая-2005 | <input checked="" type="checkbox"/> | 12000 |
| | | ЦУМ | 23-марта-2005 | <input checked="" type="checkbox"/> | 12000 |
| | | БГУ | 15-января-2005 | <input type="checkbox"/> | 12000 |
| Итого для 'Код услуги' = 8 (3 записей) | | | | | Sum 36000 |
| 10 | 15 | ЦУМ | 25-мая-2005 | <input checked="" type="checkbox"/> | 4000 |
| | | БГУ | 25-мая-2005 | <input checked="" type="checkbox"/> | 4000 |
| Итого для 'Код услуги' = 10 (2 записей) | | | | | Sum 8000 |
| | | | | | ИТОГО 151000 |

Рис. 3.25. Отчет «Заказы»

7. Аналогично создается вторая кнопка управляющей формы. Только в окне **Создание кнопок** выполняются следующие действия:
- > на 1-м шаге выбираются опции: из списка Категории – Работа с отчетом, а из списка Действия – Просмотр отчета;
 - > на 2-м шаге устанавливается флажок в положение текст и вводится текст, помещаемый на кнопку – **Просмотр Отчета ЗАКАЗЫ**.
8. Для установки одинаковых размеров обеих кнопок их необходимо выделить при нажатой клавише **Shift** и выполнить следующие команды:

Формат/Размер/по самому высокому
Формат/Размер/по самому широкому
Формат/Выровнять/по верхнему краю

Созданную форму сохранить под именем **Управляющая форма** и вывести ее на печать, выполнив команду **Файл/Печать**.

В результате получится следующего вида форма (рис. 3.26):

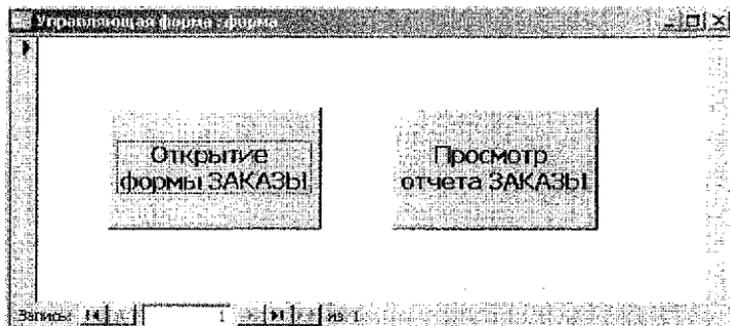


Рис. 3.26. Управляющая форма

4. Теоретический материал

4.1. Первоначальные сведения о базах данных

Первоначально средства вычислительной техники (ВТ) предполагалось использовать для выполнения различного рода математических вычислений. Развитие этого направления привело к развитию разделов математики, связанных с численными методами вычислений, и к появлению алгоритмических языков, удобных для математических расчетов. Затем, по мере увеличения возможностей и уменьшения стоимости ВТ, получило развитие второе направление, связанное с использованием средств ВТ в автоматизированных информационных системах. Здесь вычислительные возможности отходят на второй план – основной функцией становится поддержка надежного хранения информации и выполнение специфических преобразований информации. Со временем именно второе направление, связанное с хранением и обработкой данных, стало доминирующим, особенно после появления персональных компьютеров.

4.2. Определение базы данных (БД)

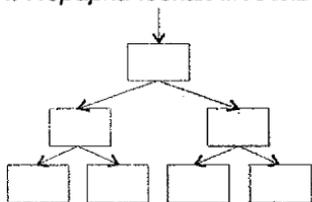
База данных (database) – это поименованная совокупность взаимосвязанных данных, организованных на машинном носителе средствами СУБД, отображающая отношения и свойства объектов в некоторой предметной области.

Объектом может быть предмет, вещество, событие, лицо, явление, абстрактное понятие, т.е. все то, что характеризуется набором значений некоторой совокупности **атрибутов** – информационного отображения свойств объекта. Например, объект «книга» характеризуется атрибутами: наименование, авторы, количество страниц, тираж, цена и т.п.

4.3. Модели данных

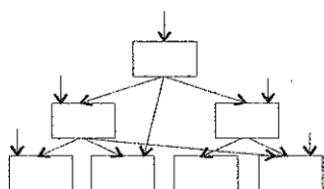
В настоящее время существует три основные модели организации БД.

I. Иерархическая модель



В данной модели любой объект может подчиняться только одному объекту вышестоящего уровня. Таким образом один объект главный, остальные – подчиненные.

II. Сетевая модель



В сетевой модели любой объект может быть подчинен нескольким объектам, т.е. может быть и главным и подчиненным, и может участвовать в любом количестве взаимосвязей.

III. Реляционная модель

В реляционной модели информация представляется в виде прямоугольных таблиц. Каждая таблица состоит из строк и столбцов и имеет имя, уникальное внутри БД. В свою очередь, каждая строка (запись) такой таблицы содержит информацию, относящуюся только к одному конкретному объекту, а каждый столбец (поле) таблицы имеет уникальное для своей таблицы имя.

Реляционные базы данных (РБД), в отличие от *иерархических* и *сетевых моделей*, позволяют организовывать связи между таблицами в любой момент. Для этого в РБД реализован механизм *внешних ключей*. В каждой таблице БД имеется хотя бы одно поле, служащее ссылкой для другой таблицы. В терминологии РБД такие поля называются полями внешних ключей. С помощью внешних ключей можно связывать любые таблицы БД на любом этапе работы с БД.

4.4. Реляционная модель данных

Реляционная БД (РБД) – это совокупность простейших двумерных логически взаимосвязанных таблиц-отношений, состоящих из множеств полей и записей, отражающих некоторую предметную область.

Реляционная модель данных была предложена Е. Коддом, известным американским специалистом в области баз данных. Основные концепции этой модели были впервые опубликованы в 1970 г. Будучи математиком по образованию, Кодд предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность,

декартово произведение). Он показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как отношение (по-английски – relation, отсюда и название – реляционные базы данных).

Одна из главных идей Кодда заключалась в том, что связь между данными должна устанавливаться в соответствии с их внутренними логическими взаимоотношениями. Второй важный принцип, предложенный Коддом, заключается в том, что в реляционных системах одной командой могут обрабатываться целые файлы данных, в то время как ранее одной командой обрабатывалась только одна запись.

Базовые понятия реляционных баз данных (РБД)

- **Отношение** – информация об объектах одного типа, например, о клиенте, заказах, сотрудниках. В реляционной БД отношение хранится в виде таблицы.
- **Атрибут** – определенная часть информации о некотором объекте – например, адрес клиента или зарплата сотрудника. Атрибут обычно хранится в виде **столбца (поля)** таблицы.
- **Тип данных** – понятие, которое в реляционной модели полностью эквивалентно соответствующему понятию в алгоритмических языках. Набор поддерживаемых типов данных определяется СУБД и может сильно различаться в разных системах.
- **Связь** – способ, которым связана информация в одной таблице с информацией в другой таблице. Связи осуществляются с помощью совпадающих полей, которые называются **ключом**.
- **Объединение** – процесс объединения таблиц или запросов на основе совпадающих значений определенных атрибутов.

Правила (нормализации) построения реляционной БД

Нормализация представляет собой процесс реорганизации данных путем ликвидации повторяющихся групп и иных противоречий с целью приведения таблиц к виду, позволяющему осуществлять непротиворечивое и корректное редактирование данных. Окончательная цель нормализации сводится к получению такого проекта БД, в котором каждый факт появляется только в одном месте, т.е. исключена избыточность информации.

1. Каждое поле любой таблицы должно быть уникальным.
2. Каждая таблица должна иметь уникальный идентификатор (**первичный ключ**), который может состоять из одного или нескольких полей таблицы.
3. Для каждого значения первичного ключа должно быть одно и только одно значение любого из столбцов данных, и это значение должно относиться к объекту таблицы (т.е. в таблице не должно быть данных, которые не относятся к объекту, определяемому первичным ключом, а также информация в таблице должна полностью описывать объект).
4. Должна иметься возможность изменять значения любого поля (не входящего в первичный ключ), и это не должно повлечь за собой изменения другого поля (т.е. не должно быть вычисляемых полей).

4.5. Системы управления базами данных (СУБД)

Поддержание баз данных в компьютерной среде осуществляют программные средства – системы управления базами данных (database management system), которые представляют собой совокупность программных и языковых средств общего или специализированного назначения, необходимых для создания баз данных на машинных носителях, поддержания их в актуальном состоянии и организации доступа к ним различных пользователей в условиях принятой технологии обработки данных.

СУБД – это управляющие программы, которые обеспечивают все манипуляции с базами данных: создание базы, ее ведение, ее использование многими пользователями и др., т. е. реализуют сложный комплекс функций по централизованному управлению базой данных и обслуживают интересы пользователей.

СУБД можно рассматривать как программную оболочку, которая находится между базой данных и пользователем. Она обеспечивает централизованный контроль защиты и целостности данных, доступ к данным, их обработку, формирование отчетов на основе базы данных и другие операции и процедуры.

Реляционная система управления базами данных (РСУБД)

Набор средств для управления РБД называется реляционной системой управления базами данных, которая может содержать утилиты, приложения, службы, библиотеки, средства создания приложений и другие компоненты. Будучи связанной посредством общих ключевых полей, информация в РБД может объединяться из множества таблиц в единый результирующий набор.

Основными функциями РСУБД являются:

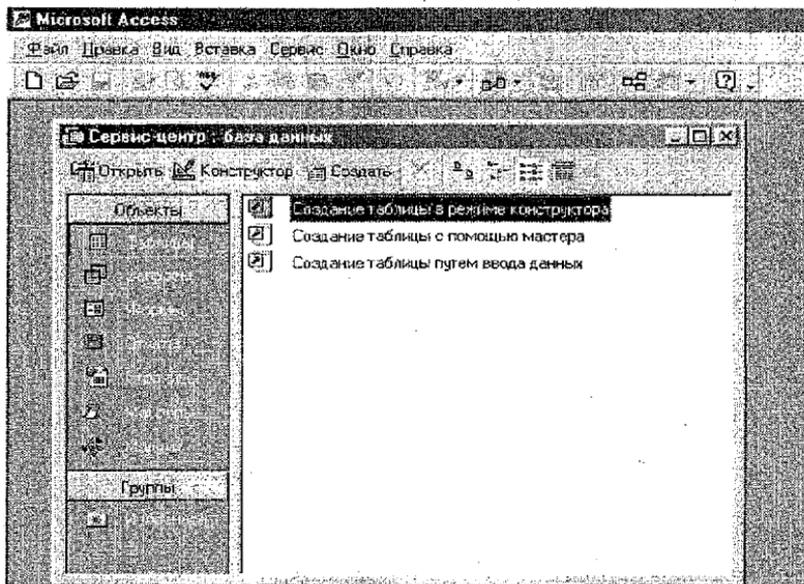
- *Определение данных* – какая информация будет храниться, задать структуру БД и их тип.
- *Обработка данных* – можно выбирать любые поля, сортировать и фильтровать данные. Можно объединять данные и подводить итоги.
- *Управление данными* – корректировать и добавлять данные.

4.6. Общая характеристика СУБД ACCESS

Microsoft Access – это функционально полная реляционная СУБД, в которой предусмотрены все необходимые средства для определения и обработки данных, а также для управления ими при работе с большими объемами информации. Различные ее версии входят в состав программного пакета MS Office и работают в среде Windows (3.11/95/98/2000/XP).

Окно БД в Access

После создания нового файла БД или открытия существующего в рабочей области окна Access появляется окно базы данных:



Основные объекты СУБД Access

База данных в Access может содержать различные элементы, называемые *объектами*. Объектами в Access являются таблицы, запросы, формы, отчеты, страницы доступа к данным, макросы и модули, переключиться между которыми можно на панели в левой части окна БД.

Таблица – объект, используемый для хранения пользовательских данных. Таблица содержит поля и записи.

Запрос – объект, который позволяет пользователю получить нужные данные из одной или нескольких таблиц.

Форма – объект, предназначенный для ввода данных, отображения их на экране или управления работой приложения.

Отчет – объект, предназначенный для формирования выходного документа, который может быть распечатан.

Страницы доступа к данным позволяют редактировать, просматривать и обрабатывать данные, используя интернет-браузер.

Макрос – набор команд, который позволяет автоматизировать часто выполняемые операции.

Модуль – программы, написанные на языке программирования Visual Basic, которые могут разрабатываться пользователем для реализации нестандартных процедур при создании приложения.

Режимы работы с объектами

Кнопки для работы с объектами БД расположены на Панели инструментов окна БД:

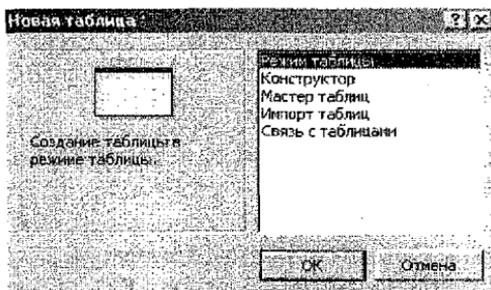
Открыть – позволяет перейти в режим редактирования таблицы, выполнения запроса, загрузки формы, построения отчета, запуска макроса.

Конструктор – обеспечивает переход к режиму настройки выбранного объекта.

Создать – позволяет приступить к созданию нового объекта выбранного типа.

4.7. Работа с таблицами

Чтобы создать таблицу, нужно перейти к списку таблиц и нажать кнопку **Создать**. Появится новое диалоговое окно **Новая таблица**:

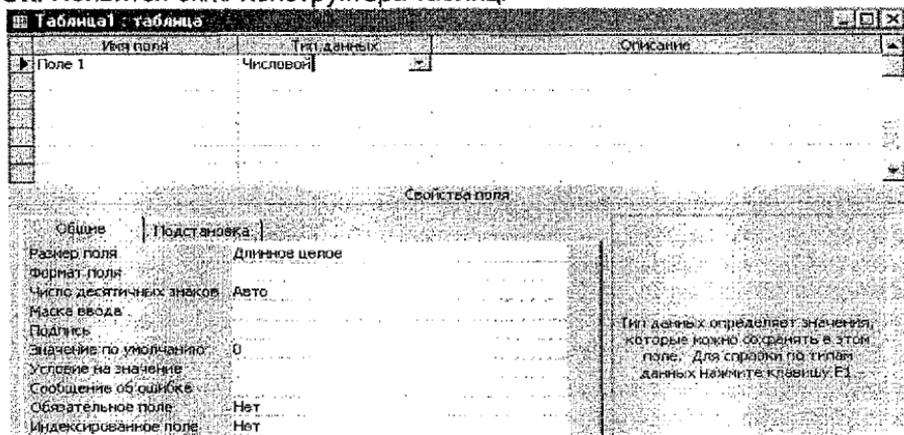


Таблицы в Access можно создать несколькими способами:

- построить новую таблицу «с нуля», воспользовавшись **Конструктором**;
- запустить **Мастер таблиц** – специальную программу, предлагающую создать таблицу в пошаговом режиме на базе типовых решений, имеющихся в Access;
- импортировать таблицу БД из файла какой-либо программы, например, FoxPro или Excel.

Создание таблицы БД с помощью Конструктора

Чтобы приступить к созданию новой таблицы БД, необходимо в диалоговом окне «Новая таблица» выбрать пункт **Конструктор** и нажать **ОК**. Появится окно Конструктора таблиц:



Условно это окно можно поделить на две области:

- **Область полей таблицы.** В этой части окна вводится название полей, определяется тип данных каждого поля, а также делаются краткие комментарии по поводу того, какие данные находятся в этом поле (графа *Описание*).
- **Область свойств поля.** Здесь указываются отдельные характеристики каждого поля, необходимые для определения параметров сохранения данных в поле, их дальнейшего отображения и редактирования. Эта область содержит две вкладки: **Общие** и **Подстановка**. Вкладка **Подстановка** содержит список некоторых дополнительных параметров, необходимых, в частности, для настройки связей с полями других таблиц.

Задание имени поля

Имя поля задается в столбце **Имя поля**. Имя может содержать не более 64 знаков, при этом допустимы любые символы, кроме точки, восклицательного знака и угловых скобок. Повторение имен полей не допускается.

Определение типа данных

Для каждого поля необходимо указать тип данных, содержащихся в нем. Тип данных выбирается из списка, который можно вызвать щелчком мыши в столбце **Тип данных**. Access оперирует следующими типами данных:

- **Текстовый** – для хранения обычного текста с максимальным количеством символов 255.
- **Поле МЕМО** – для хранения больших объемов текста до 65 535 символов.
- **Числовой** – для хранения действительных чисел.
- **Дата/время** – для хранения календарных дат и текущего времени.
- **Денежный** – эти поля содержат денежные суммы.
- **Счетчик** – для определения уникального системного ключа таблицы. Обычно используется для порядковой нумерации записей. При добавлении в таблицу новой записи значение этого поля увеличивается на 1 (единицу). Значения в таких полях не обновляются.
- **Логический** – для хранения данных, принимающих значения: Да или Нет.
- **Поле объекта OLE** – для хранения объектов, созданных в других приложениях.
- **Гиперссылка** – для хранения ресурсов сети Internet или Intranet.

Описание свойств полей

Как уже отмечалось, характеристики отдельных полей определяются в области свойств поля (вкладка **Общие**). Каждое поле имеет определенный набор свойств – в зависимости от типа поля. Некоторые типы полей имеют схожие наборы свойств полей. Ниже перечислены основные свойства полей.

- **Размер поля** – максимальная длина текстового поля (по умолчанию 50 знаков) или тип данных числового поля. Рекомендуется задавать минимально допустимое значение этого свойства, потому что обработка данных меньшего размера выполняется быстрее.

Если тип данных – числовой, допустимы следующие значения свойства **Размер поля**:

| Значение | Описание | Дробная часть |
|---------------------------------------|---|---------------|
| Байт | Числа от 0 до 255 (без дробной части) | нет |
| Целое | Числа от -32 768 до 32 767 (без дробной части) | нет |
| Длинное целое (значение по умолчанию) | Числа от -2 147 483 648 до 2 147 483 647 (без дробной части) | нет |
| Одинарное с плавающей точкой | Числа от -3,402823Е38 до -1,401298Е-45 для отрицательных значений, и от 1,401298Е-45 до 3,402823Е38 для положительных | 7 |
| Двойное с плавающей точкой | Числа от -1,79769313486232Е308 до -4,94085645841247Е-324 для отрицательных значений, и от 1,79769313486231Е308 до 4,94085645841247Е-324 для положительных | 15 |

Замечание. В случае преобразования поля в меньшее по размеру, может произойти потеря данных.

- **Формат поля** – формат отображения данных на экране или печати. Как правило, используется формат, заданный по умолчанию.
- **Число десятичных знаков** – задает для числового и денежного типа данных число десятичных знаков после запятой.
- **Маска ввода** – определяет форму, в которой данные вводятся в поле (средство автоматизации ввода данных).
- **Подпись** – обозначение для поля, которое будет использоваться для отображения поля в таблице, форме или отчете. Если это значение не определено, в качестве подписи будет взято имя поля.
- **Значение по умолчанию** – стандартное значение, которое автоматически вводится в поле при формировании новой записи данных.
- **Условие на значение** – задает ограничения на вводимые значения, тем самым позволяет осуществлять контроль над правильностью ввода данных.
- **Сообщение об ошибке** – задает текст сообщения, выводимый на экран в случае нарушения условия на значение.
- **Обязательное поле** – определяет, может ли данное поле содержать значения Null (т.е. оставаться пустым), или нужно обязательно вводить в это поле данные.
- **Индексированное поле** – используется для операций поиска и сортировки записей по значению, хранящемуся в данном поле, а также для автоматического исключения дублирования записей. Поля типа **МЕМО**, **Объект OLE** и **Гиперссылка** не могут индексироваться.

Определение ключевого поля

После задания характеристик всех полей следует выбрать, по крайней мере, одно ключевое поле. Как правило, в качестве ключевых полей указываются поля, которые имеют неповторяющиеся данные или создаются поля с типом данных **Счетчик**. В любом случае, поле ключа не должно содержать повторяющихся данных. Чтобы определить ключ, не-

необходимо выделить нужное поле (или поля) и нажать кнопку **Ключевое поле** на панели инструментов или соответствующую команду в п. м. **Правка**. Слева от маркера появится изображение ключа.

Сохранение таблицы

Перед вводом информации спроектированную таблицу необходимо сохранить: нажать кнопку **Сохранить** на панели инструментов или соответствующую команду в п. м. **Файл** и ввести название таблицы, после чего на экране появляется вопрос «Создать ключевое поле сейчас?» (Да или Нет)

Если выбирается ответ «Да», то Access создаст автоматически поле с именем «Код» и типом данных **Счетчик**, если «Нет», – то таблица будет создана без ключевого поля. В этом случае необходимо открыть созданную таблицу в режиме **Конструктора** и определить «вручную» ключевое поле.

Ввод данных

Чтобы перевести таблицу в режим ввода информации, нужно перейти в режим **Таблицы**. Поля заполняются последовательно. Переход от одного поля к другому удобно выполнять клавишей **Tab** (или комбинацией **Shift+Tab** – в обратном направлении). Если при проектировании таблицы для некоторых полей были предусмотрены значения по умолчанию, эти значения автоматически появятся в соответствующих полях. Записи в таблице можно перемещать, копировать и удалять теми же способами, что и в электронных таблицах, то есть сначала выделить строки, а потом выполнить необходимую операцию. Столбец можно выделить щелчком мыши по заголовку. Столбцы можно перемещать вправо и влево, пользуясь методом *drag and drop* (перетащить и бросить).

При необходимости можно вернуться в режим **Конструктора**. Это даст возможность что-либо подправить в структуре таблицы.

Сортировка данных в таблице

Данные, находящиеся в таблице, можно отсортировать в порядке возрастания или убывания. Для этого нужно поместить курсор мыши в любую ячейку столбца, значения которого будут отсортированы и из п. м. **Записи** выбрать команду **Сортировка** или нажать на панели соответствующую кнопку.

4.8. Создание связей между таблицами БД

Связь между таблицами устанавливается путем определения в одной таблице (**подчиненной**) поля, соответствующего ключу другой таблицы (**главной**). Установленная связь свяжет записи, содержащиеся в заданном поле одинаковые значения. Созданные связи позднее Access будет использовать в запросах, формах или отчетах.

Замечания.

- Оба связываемых поля должны иметь одинаковый **тип данных**.
- Свойства **Размер поля** для обоих связываемых полей **числового типа** должны быть одинаковыми.
- Если ключевым полем главной таблицы является поле с типом данных **Счетчик**, то это поле можно связать с числовым полем подчиненной таблицы. При этом для числового поля связанной таблицы для свойства **Размер поля** должно быть задано значение **Длинное целое**.

Целостность данных

Целостность данных – это набор правил, которые поддерживают корректность связей между записями в связанных таблицах и обеспечивают защиту данных от случайных изменений или удалений.

Эти правила включают:

- В подчиненной таблице нельзя вводить записи, которые не связаны с записью главной таблицы.
- В главной таблице нельзя изменять значение ключевого поля, если в подчиненной таблице существуют записи, которые с ней связаны.
- В главной таблице нельзя удалять записи, если в подчиненной таблице существуют связанные с ней записи.

Каскадные операции

Целостность данных в связанных таблицах обеспечивают каскадные операции двух видов:

- **операции каскадного обновления;**
- **операции каскадного удаления.**

Эти операции можно включать и выключать путем установки соответствующих флажков: «Каскадное обновление связанных полей» и «Каскадное удаление связанных полей».

Если установлен флажок «Каскадное обновление связанных полей», то любые изменения в значении ключевого поля в главной таблице, которая стоит на стороне «один» в отношениях 1:М, ведут к автоматическому обновлению соответствующих значений во всех связанных записях.

При установке флажка «Каскадное удаление связанных таблиц» при удалении записи из главной таблицы обеспечивается автоматическое удаление связанных записей в подчиненных таблицах.

Удаление (изменение) связей

- Открыть окно **Схема данных**;
- активизировать левой кнопкой мыши связь, которую необходимо удалить (изменить);
- правой кнопкой мыши вызвать контекстно-зависимое меню и выбрать команду **Удалить (Изменить)** соответственно.

4.9. Типы отношений между таблицами

Существует три типа отношений между таблицами:

Один-к-одному (1:1). Значению ключа в каждой записи в главной таблице могут соответствовать значения в связанном поле только в одной записи подчиненной таблицы. В этом случае связь между таблицами может быть установлена только через ключевые поля обеих таблиц.

Один-ко-многим (1:М). Значению ключа в каждой записи в главной таблице могут соответствовать значения в связанном поле (полях) в нескольких записях подчиненной таблицы. Этот тип отношения довольно часто используется в реляционных БД.

Много-ко-многим (М:М). Возникает между двумя таблицами, когда одна запись из первой таблицы А (выходная связь) может быть связана

больше чем из одной записью другой таблицы В (принимающая), в свою очередь, одна запись с другой таблицы может быть связана больше чем с одной записью первой таблицы. Эта схема реализуется только при помощи третьей соединительной таблицы, ключ связи которой состоит, как минимум, из двух полей. Эти поля являются полями внешнего ключа в таблицах А и В. Первичный ключ для соединительной таблицы – это обычно комбинация из внешних ключей.

Если между таблицами имеются связи типа М:М, создается дополнительная таблица пересечений, с помощью которой связь М:М будет сведена к двум связям типа 1:М. Access не позволяет определить прямую связь М:М между двумя таблицами.

4.10. Формирование запросов

Понятие запроса к базе данных

Запрос (query) есть механизм выбора и представления информации из базы данных. Запрос направляется пользователем или программой в СУБД для поиска отдельных записей в базе данных. Вопрос, сформированный по отношению к базе данных, и есть запрос.

Access позволяет создавать **запросы на выборку** и **активные запросы**. Они могут быть *однотабличными* (в качестве исходной информации используются данные одной таблицы) и *много табличными* (в качестве исходной информации используются данные нескольких таблиц).

Запрос на выборку – это наиболее часто используемый тип запроса, который задает вопрос базе данных и отображает полученный ответ в виде динамического набора данных. Среди запросов на выборку можно выделить следующие:

- *простой запрос на выборку* – выбирает данные из взаимосвязанных таблиц или других запросов с учетом условий пользователя. Результатом его является таблица, которая существует до закрытия запроса;
- *запросы с групповыми операциями (итоговый запрос)* – выполняют вычисления с использованием данных из групп записей;
- *запросы с параметром (параметрический запрос)* – это запросы, в которых критерий отбора может задать сам пользователь, введя нужный параметр при вызове запроса;
- *запросы с вычислениями* – позволяют осуществлять вычисления над полями БД;
- *перекрестные запросы* – обеспечивают создание результирующей таблицы на основе расчетов, полученных при анализе группы таблиц. С помощью данного запроса можно сгруппировать большой объем информации и представить его в удобном для восприятия виде.

Активные запросы, или запросы-действия, вносят изменения в саму БД. Среди них выделяют следующие:

- *запросы на создание таблицы* - создают новую таблицу БД, в которой сохраняется результат запроса;
- *запросы на обновление* - изменяют информацию в полях таблицы БД;

- *запросы на добавление* - производят добавление записей из таблицы с результатами запроса в таблицу БД;
- *запросы на удаление* - удаляют группу записей из одной таблицы или нескольких взаимосвязанных таблиц базы, для которых задано каскадное удаление связанных записей.

Для выборки информации из БД предназначены **Мастера запросов и Конструкторы запросов**.

Запрос по образцу (QBE) – средство для отыскания необходимой информации в базе данных (из одной или нескольких таблиц), обычно путем занесения известных значений в таблицу. Запрос формируется не на специальном языке, а путем заполнения бланка запроса в окне **Конструктора запросов**. Указываются критерии выборки записей в исходной таблице. С точки зрения пользователя, конструирование запроса – это просто пометка нужных элементов формы в окне Конструктора запросов.

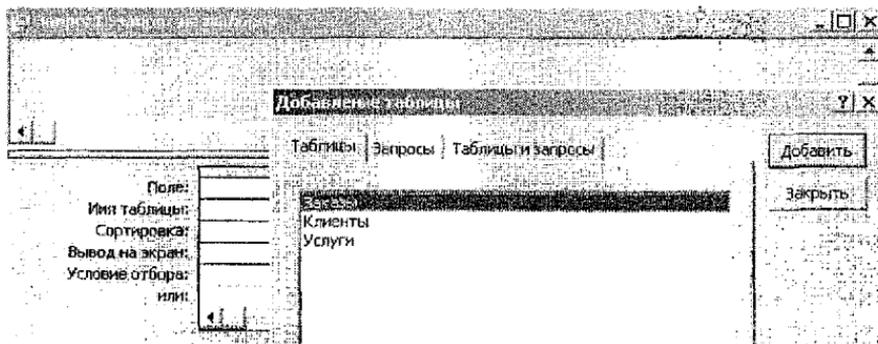
Создание запросов выбора с помощью Конструктора

При создании запроса необходимо определить:

- поля в базе данных, по которым будет идти поиск информации;
- предмет поиска в базе данных;
- вид (перечень полей) в результате выполнения запроса.

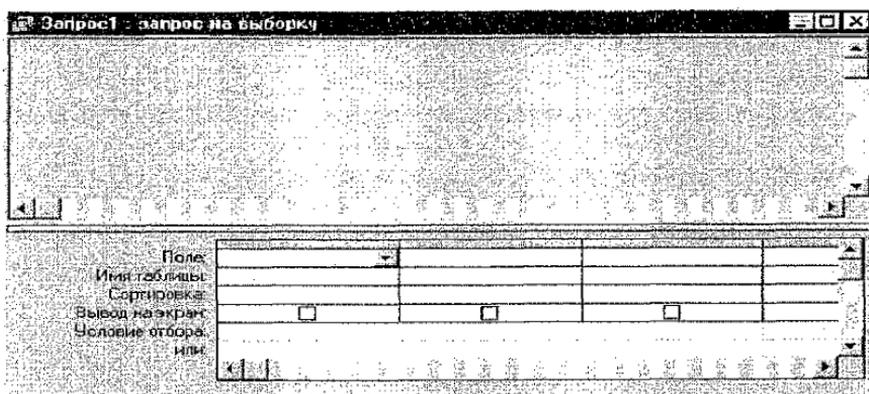
Последовательность действий:

- 1) в окне БД выбрать вкладку **Запросы** и щелкнуть кнопку **Создать** – откроется окно диалога **Новый запрос**;
- 2) выбрать опцию **Конструктор** и нажать кнопку **ОК**;
- 3) в появившемся диалоговом окне **Добавление таблицы** из списка следует выбрать имя таблицы-источника или запроса-источника, на основе которых будет проводиться выбор данных, и щелкнуть кнопку **Добавить**:



Если запрос формируется на основе нескольких таблиц, то действия, указанные в п. 3, повторяются для каждой таблицы-источника;

- 4) закрыть окно диалога **Добавление таблицы** кнопкой **Закреть** – на экране появится окно **Конструктора запросов**:



Окно Конструктора условно состоит из двух частей – верхней и нижней.

В *верхней части* окна Конструктора запроса размещается схема данных запроса, которая содержит список таблиц-источников и отражает связь между ними.

В *нижней части* окна находится **Бланк запроса (бланк QBE)**, в котором каждая строка выполняет конкретную функцию:

- в строке **Поле** указываются имена полей, участвующих в запросе;
- в строке **Имя таблицы** отображаются имена таблиц, из которых взяты поля;
- в строке **Сортировка** задается порядок сортировки записей в запросе;
- в строке **Вывод на экран** осуществляется пометка полей, выводимых в результате выполнения запроса (посредством установки флажков);
- в строке **Условие отбора** задаются условия отбора записей по значению поля. Условие отбора – это выражение, которое состоит из операторов сравнения и операндов, используемых для сравнения. В качестве операндов могут быть выбраны некоторые *заданные значения* и *идентификаторы* (имена полей, таблиц, запросов, форм, отчетов и т.д.). Идентификаторы необходимо заключать в квадратные скобки. Если надо указать ссылку на поле в конкретной таблице, форме, отчете, то прописывается **полное имя поля** в виде:

[Имя таблицы]![Имя поля]

- строка **или** позволяет объединять условия отбора логической операцией ИЛИ, причем она может состоять из нескольких строк бланка запроса.

С помощью инструментария окна *Конструктора* запрос формируется в два этапа:

- выбираются поля результатов запроса;
- указываются критерии для выборки, группирования и упорядочения данных.

Добавление (удаление) полей в Бланке запроса

Выбор полей с таблицы-источника (из верхней части окна **Запрос на выборку**) в *Бланк запроса* (нижнюю часть окна) можно выполнить следующими способами:

- путем «перенести-и-бросить»;
- путем выбора поля таблицы в верхней части окна двойным щелчком мыши;
- с помощью выбора нужного поля из списка, находящегося непосредственно в ячейке строки *Поле Бланка запроса*.

В строке *Поле* появится имя поля, а в строке *Имя таблицы* автоматически появится имя таблицы-источника; в строке *Вывод на экран* будет установлен флажок.

Бланк запроса можно очистить (удалить все поля) командой **Правка/Очистить бланк**.

Сортировка результатов выборки

Для сортировки результатов выборки по одному полю необходимо перейти на строку **Сортировка** нужного поля и в раскрывающемся списке выбрать способ сортировки *по возрастанию* или *по убыванию*. Для отказа от сортировки по выбранному ранее полю следует выбрать значение **Отсутствует**.

Запуск запроса

Для запуска запроса на исполнение из окна *Конструктора* надо на панели инструментов нажать кнопку «**Запуск**»  или выполнить команду **Запрос/Запуск**. Результаты выборки данных по запросу выводятся на экран в режиме таблицы.

Вычисляемые поля в запросах. Построитель выражений

Существует ряд вычислений, которые можно выполнить в запросе. Например, найти сумму или среднее по значениям одного поля, перемножить значения двух полей или вычислить дату, отстоящую от текущей на три месяца.

Результаты вычислений, выводящиеся в поле, не запоминаются в базовой таблице. Вычисления снова производятся всякий раз, когда выполняется запрос, поэтому результаты всегда представляют текущее содержимое базы данных. Обновить вычисленные результаты вручную невозможно.

Для определения вычисляемого поля можно использовать встроенные функции **Access** или выражения, создаваемые пользователем. Выражения, определяемые пользователем, дают возможность выполнять действия с числами, датами и текстовыми значениями в каждой записи с использованием данных из одного или нескольких полей.

Результаты вычислений не должны обязательно отображаться в поле.

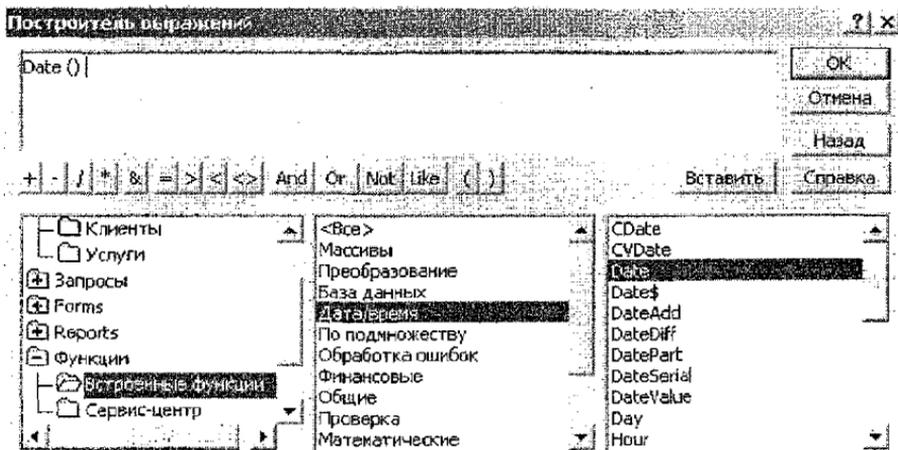
Их можно использовать в условиях отбора для определения записей, которые выбираются в запросе, или для определения записей, над которыми производятся какие-либо действия.

Новое вычисляемое поле создается прямо в *Бланке запроса*. Выражение записывается в пустую ячейку *Поле бланка* с помощью **Построителя выражений**, который предназначен для составления математических, логических и других выражений.

Построитель выражений в открытом окне **Конструктора** вызывается одним из следующих способов:

- нажать кнопку **Построить** на панели инструментов;
- щелкнуть правой кнопкой мыши по строке **Поле** (или **Условие отбора**) для нового (или выбранного) поля и из контекстного меню выбрать команду **Построить**.

Окно построителя выражений состоит из трех разделов:



В верхней части окна построителя расположено поле построения выражения.

В средней части окна построителя находятся кнопки с часто используемыми операторами. При нажатии на одну из этих кнопок построитель вставит соответствующий оператор в текущую позицию поля выражения.

Замечание. Чтобы вывести полный список операторов, можно выбрать папку *Операторы* в нижней левой области и нужный тип в средней – в правой области будут выведены все операторы выбранного типа.

В нижней части окна построителя находятся три области.

В левой области содержится список источников данных (таблицы, запросы, формы, объекты базы данных) и средства для записи выражений (встроенные и определенные пользователем функции, константы, операторы и общие выражения).

В *средней* области задаются определенные элементы или типы элементов для папки, заданной в левой области. Например, если выбрать в левом поле *Встроенные функции*, то в среднем поле появится список всех типов функций Access.

В *правой* области выводится список значений (если они существуют) для элементов, заданных левым и средним полями. Например, если выбрать в левой области *Встроенные функции* и тип *Дата/время* в средней области, то в правой области будет выведен список всех встроенных функций выбранного типа.

Поиск нужных элементов начинается с левой области. Списки объектов в левой, средней и правой областях имеют иерархическое подчинение.

Последовательность вставки элемента выражения в верхнее поле ввода:

- 1) в левой области выбрать необходимую папку;
- 2) открыть папку для выбора нужного объекта (элемента) – двойным щелчком мыши;
- 3) последовательно в средней и правой областях выбрать нужный элемент и перенести его в поле ввода выражения (в верхнюю часть окна) кнопкой **Вставить** или дважды щелкнуть элемент мышью;
- 4) в средней части окна *Построителя* выбрать нужный оператор и щелкнуть мышью для переноса его в создаваемое выражение;
- 5) после формирования нужного выражения нажать кнопку **OK**.

Формирование Условий отбора

Список операторов, используемых при задании выражений, следующий:

➤ операторы сравнения:

| | |
|---------------|----------------|
| = (равно) | >= (не меньше) |
| <> (не равно) | < (меньше) |
| > (больше) | <= (не больше) |

BETWEEN – позволяет задать диапазон значений. Синтаксис: **Between** «Выражение» **And** «Выражение» (например: **BETWEEN 10 And 20** означает тоже, что и логическое выражение **>=10 AND <=20**).

IN – позволяет задавать используемый для сравнения список значений (операндом является список, заключенный в круглые скобки). Например: **IN**("Брест", "Минск", "Гродно") означает то же самое, что и логическое выражение **"Брест" OR "Минск" OR "Гродно"**.

➤ логические операторы:

AND (например: **>=10 AND <=20**)

OR (например: **<50 OR >100**)

NOT (например: **Is Not Null** – поле, содержащее какое-либо значение).

➤ оператор **LIKE** – проверяет соответствие *текстового* или *Мето поля* по заданному шаблону символов.

Таблица символов шаблона

| Символы шаблона | Соответствие в выражении |
|------------------------|--|
| ? | Любой один текстовый символ |
| * | Соответствует любой цифре или любому символу |
| # | Любая одна цифра |
| [список знаков] | Любой один знак в «списке знаков» |
| [!список знаков] | Любой один знак, который не входит в «список знаков» |

Примеры использования оператора Like:

LIKE "C ** – строки, начинающиеся с символа C;

LIKE "[A - Z] #" – любой символ от A до Z и цифра;

LIKE "[! 0 - 9 ABC] * # #" – строки, начинающиеся с любого символа кроме цифры или букв A, B, C и заканчивающиеся на 2 цифры.

Сложные критерии выборки

Часто приходится выбирать записи по условию, которое задается для нескольких полей таблицы или по нескольким условиям для одного поля. В этом случае применяются «И-запросы» (выбор записей только при условии выполнения всех условий) и «ИЛИ-запросы» (выбор записей при выполнении хотя бы одного из условий).

При задании «ИЛИ-запроса» каждое условие выборки должно размещаться на отдельной строке *Бланка запроса*.

При задании «И-запроса» каждое условие выборки должно размещаться на одной строке, но в разных полях *Бланка запроса*.

Эти операции могут быть заданы явно с помощью операторов **OR** и **AND** соответственно.

Условие отбора для даты и времени

Для того, чтобы сообщить Access, что вводится дата (или время), необходимо значение даты (или времени) заключить в знак #. Например: #15.10.05#

В MS Access используются следующие встроенные функции, которые вводят условия отбора для даты и времени:

Date () – возвращает текущую системную дату.

Day(дата) – возвращает значение дня месяца от 1 до 31.

Month(дата) – возвращает значение месяца от 1 до 12.

Year(дата) – возвращает значение года от 100 до 9999.

Weekday(дата) – по умолчанию возвращает целое число от 1 (Воскресенье) до 7 (Суббота), соответствующее дню недели.

Hour(дата) – возвращает целое число от 0 до 23, представляющее значение часа в дате.

Datpart(интервал, дата) – возвращает номер квартала или номер недели в зависимости, какой код интервала задается:

"q" – определение квартала (значение от 1 до 4);

"ww" – определение номера недели в году (значение от 1 до 53).

Функции Iif() и Format()

Функция **Iif(условие; еслиИстина; еслиЛожь)** – возвращает один из двух аргументов в зависимости от результата вычисления выражения.

| Аргумент | Назначение |
|-----------------|--|
| условие | Выражение, значение которого нужно вычислить |
| еслиИстина | Значение или выражение, возвращаемые, если значением выражения является "Истина" (1) |
| еслиЛожь | Значение или выражение, возвращаемые, если значением выражения является "Ложь" (0) |

Функция **Format(выражение; инструкция форматирования)** – возвращает строку, содержащую выражение, отформатированное согласно инструкциям форматирования.

Для выражений **даты/времени** можно применять следующие символы в инструкции форматирования:

| Символ | Описание |
|---------------|---|
| c | Полный формат даты |
| ddd | Первые три буквы названия дня недели (от Sun до Sat). |
| dddd | Полное название дня недели (от Sunday до Saturday). |
| w | День недели (от 1 до 7). |
| ww | Неделя года (от 1 до 53). |
| mmm | Первые три буквы названия месяца (от Jan до Dec). |
| mmmm | Полное название месяца (от January до December). |
| q | Квартал года (от 1 до 4). |
| y | День года (от 1 до 366). |
| yy | Последние две цифры года (от 01 до 99). |

Проектирование параметрических запросов

Запрос с параметрами выводит одно или несколько предопределенных диалоговых окон, в которых пользователь должен ввести значения параметров (условия отбора).

Параметрический запрос каждый раз при выполнении требует ввода определенных параметров. Преимущества данного вида запросов:

- не нужно постоянно модифицировать запрос в режиме *Конструктора*;
- удобно использовать в формах и отчетах, т. к. каждый раз при их открытии Access запрашивает у пользователя требуемый параметр.

Чтобы установить параметр, нужно, вместо конкретных данных, в *Бланк запроса* в строку **Условие отбора** ввести имя или фразу, заключенную в квадратные скобки, т. е. то, что заключено в [] Access рассматривает как параметр.

Например, для поля, которое выводит даты, можно ввести приглашения следующего вида «Введите начальную дату:» и «Введите конечную дату:», чтобы задать границы диапазона значений. В ячейке **Условие отбора** поля типа Дата/время будет стоять выражение *Between [Введите начальную дату:] And [Введите конечную дату:]*.

Проектирование итоговых запросов

Запросы, которые выполняют вычисления в группах записей, называются *итоговыми*. Можно вычислить сумму, среднее, минимальное и максимальное значения поля.

Для создания итогового запроса надо, находясь в окне *Конструктора* запросов, вызвать команду *Вид/Групповые операции* или нажать на кнопку **Групповые операции** на панели инструментов. После этого в *Бланке запроса* появится новая строка **Групповые операции**. В ней для соответствующего поля указывается тип выполняемого вычисления из раскрывающегося списка установок групповых операций:

| Групповая функция | Результат |
|--------------------------|---|
| Sum | Сумма значений поля |
| Avg | Среднее значение поля |
| Min | Минимальное значение поля |
| Max | Максимальное значение поля |
| Count | Число значений в поле (не считая пустые) |
| StDev | Среднеквадратичное отклонение значений поля |
| Var | Дисперсия значений поля |
| First | Возвращают значение поля из первой записи результирующего набора запроса |
| Last | Возвращают значение поля из последней записи результирующего набора запроса |

| Групповая установка | Результат |
|----------------------------|---|
| Группировка | Определяет группы записей, для которых рассчитываются итоговые значения. |
| Выражение | Определяет в запросе вычисляемое поле. Создание вычисляемого поля позволяет использовать в выражении несколько функций. |
| Условие | Задаёт условия отбора для поля, не используемого для определения группы. После выбора параметра «Условие» поле делается скрытым (снимается флажок в строке «Вывод на экран»). |

Проектирование перекрестных запросов

В перекрестном запросе отражаются результаты расчетов (средние величины, суммы и т. д.) по значению одного поля таблицы. Результаты вычислений группируются *по двум наборам данных*: один находится в левом столбце таблицы, другой – в верхней строке. Перекрестный запрос представляет информацию в более компактном виде, чем обычные запросы. Данные сгруппированы как по горизонтали, так и по вертикали и размещены компактно в виде двумерной таблицы.

Последовательность действий при создании перекрестного запроса с помощью Мастера:

- находясь в окне БД, выбрать вкладку **Запросы**;

- нажать кнопку **Создать**;
- в появившемся диалоговом окне **Новый запрос** выбрать опцию **Перекрестный запрос** и нажать кнопку **ОК**;
- в появившемся окне **Создание перекрестных таблиц** выбрать из списка таблицу-источник;
- далее выполнять инструкции в диалоговых окнах Мастера.

Последовательность действий при создании перекрестного запроса без Мастера:

- находясь в окне БД, на вкладке **Запросы** нажать кнопку **Создать**;
- в окне **Новый запрос** выбрать режим *Конструктор*;
- в появившемся диалоговом окне **Добавление таблицы** из списка следует выбрать имя таблицы-источника или запроса-источника, на основе которых будет проводиться выбор данных, и щелкнуть кнопку **Добавить**, после чего закрыть окно диалога **Добавление таблицы** кнопкой **Закрыть**;
- находясь в окне *Конструктора* из п.м. **Запрос** выбрать команду **Перекрестный**;
- для полей, значения которых будут *заголовками строк*, выбрать в строке **Перекрестная таблица** опцию **Заголовки строк** и оставить в строке **Групповая операция** значение **Группировка**;
- для полей, значения которых будут представлены в запросе в качестве *заголовков столбцов*, выбрать в строке **Перекрестная таблица** опцию **Заголовки столбцов** и оставить в строке **Групповая операция** значение **Группировка**;
- для расчетных (итоговых) полей, значения которых будут находиться на пересечении строк и столбцов, выбрать в строке **Перекрестная таблица** опцию **Значение**, а в строке **Групповая операция** из списка выбирается необходимая статистическая функция.

4.11. Проектирование форм и отчетов

Понятие «**Форма**» в БД означает структурированное окно, экранное поле либо независимый элемент интерфейса с заранее установленными областями для ввода либо изменения информации; визуальный шаблон, который упорядочивает предоставляемые формой данные, что позволяет лучше их организовывать и просматривать. Форма дает возможность вводить и просматривать информацию в БД.

Как объект базы данных форма используется:

- для ввода данных в таблицу, их корректировки;
- в качестве специального окна диалога для выбора;
- в виде кнопочной формы для открытия других форм либо отчетов.

В форме может быть представлена информация из нескольких связанных таблиц. Представление данных в формах может быть различным: поля располагаются в виде столбца или таблицы.

Источником данных для формы могут являться записи таблицы либо запроса. Форма облегчает восприятие информации – это гибкий способ ее представления.

Инструментарий для форм ничем не отличается от табличного: данные в формах можно искать, заменять, сортировать, фильтровать, словом, делать с ними то же самое, что и в таблицах.

Как правило, отчеты создаются для подготовки информации, содержащейся в таблицах и запросах, к выводу на печать. На этапе создания отчета над данными можно выполнить некоторые дополнительные действия: группировку и сортировку данных, а также произвести необходимые вычисления.

Подробная последовательность действий при создании и редактировании форм и отчетов приводится в разделе **«Методические рекомендации к выполнению задания № 4»**.

Вопросы к зачету и экзамену

1. Понятие базы данных (БД). Виды моделей организации данных.
2. Реляционные БД. Базовые понятия РБД.
3. Правила построения реляционных БД.
4. Системы управления базами данных (СУБД).
5. Общая характеристика СУБД Microsoft Access. Создание файла БД в Access.
6. Окно СУБД Access. Общие принципы и способы работы с объектами.
7. Объекты Microsoft Access. Их назначение и особенности.
8. Таблица как объект БД. Способы создания таблиц в Microsoft Access. Переименование, удаление, копирование таблиц.
9. Создание таблиц в режиме *Конструктора*. Создание, изменение и сохранение структуры таблицы.
10. Понятие записи и поля. Ограничения для названия полей. Ввод данных в таблицу.
11. Типы данных и их особенности.
12. Формирование значений поля при помощи таблицы подстановки из списка.
13. Ключевые поля в таблицах. Назначение и использование. Способы задания и изменения.
14. Первичный и вторичный ключи. Свойство поля: *индексированное поле*.
15. Свойства поля: *размер поля*. Его значения для числового поля. Примеры.
16. Свойства полей: *значение по умолчанию, условие на значение, сообщение об ошибке*. Примеры.
17. Свойства полей: *формат поля, маска ввода, подпись*. Примеры.
18. Работа с данными в режиме *таблицы*: удаление, добавление записей, сортировка данных.
19. Схема данных. Определение связей между таблицами. Изменение существующих связей.
20. Типы отношений между таблицами.
21. Целостность данных. Каскадные операции.
22. Запрос как объект БД. Типы и возможности запросов.
23. Запрос-выборка. Назначение. Виды.
24. Способы создания запросов в Microsoft Access.
25. Создание запросов в режиме *Конструктора*. Окно *Конструктора* запроса. Вид бланка запроса (QBE).
26. Способы задания и удаления полей в бланке запроса. Выполнение и сохранение запроса.
27. Задание условий отбора в бланке QBE: комбинация нескольких условий отбора (И-запросы, ИЛИ-запросы). Примеры.
28. Формирование условий отбора с помощью операторов *сравнения*. Примеры.
29. Формирование условий отбора с помощью *логических* операторов. Примеры.

30. Формирование условий отбора с помощью оператора *Like*. Используемые символы шаблона. Примеры.
31. Вычисляемое поле. Назначение. Правила и способы создания.
32. Условие отбора. Полное имя поля в выражении.
33. Построитель выражений. Способы его вызова.
34. Использование встроенных функций в Microsoft Access. Группы функций.
35. Задание условий отбора для полей типа Дата/время. Функции: *Day()*, *Month()*, *Weekday()*. Примеры.
36. Задание условий отбора для полей типа Дата/время. Функции: *Date()*, *Year()*, *Datepart()*. Примеры.
37. Назначение и использование функции *Format()* для выражений *Даты/времени*. Примеры.
38. Функция *Iif()* – назначение, синтаксис. Примеры.
39. Запрос с параметром. Назначение. Особенности. Правила создания. Примеры.
40. Создание итоговых запросов. Установки групповых операций. Примеры.
41. Перекрестный запрос. Назначение. Способы создания.
42. Создание перекрестного запроса с помощью *Мастера*.
43. Создание перекрестного запроса в режиме *Конструктора*.
44. Форма как объект БД. Назначение. Способы проектирования форм в Microsoft Access.
45. Создание формы с помощью *Мастера*.
46. Редактирование формы в режиме *Конструктора*. Вид окна конструктора форм. Области формы.
47. Создание управляющей формы. Примеры.
48. Отчет как объект БД. Назначение. Способы проектирования отчетов в Microsoft Access.
49. Редактирование отчета в режиме *Конструктора*.
50. Создание отчетов с помощью *Мастера*.

Литература

1. Основная литература:

- 1.1. Левчук Е.А. Технологии организации, хранения и обработки данных. - Минск: Выш. школа, 2005.
- 1.2. Савицкий Н.И. Технологии организации, хранения и обработки данных: Учеб. пособие. – М.: ИНФРА-М, 2001.
- 1.3. Бекаревич Ю. Microsoft Access за 21 занятие для студента. СПб.: БХВ – Санкт-Петербург, 2005.
- 1.4. Рудикова Л. Базы данных. Разработка приложений. СПб.: БХВ – Санкт-Петербург, 2006.
- 1.5. Кузин А.В., Демин В.М. Разработка баз данных в системе Microsoft Access: Учебник. – М.: ФОРУМ: ИНФРА-М, 2005.
- 1.6. Оскерко В.С. Практикум по технологиям баз данных: Учеб. пособие / В.С. Оскерко, З.В. Пунчик. – Мн.: БГЭУ, 2004.
- 1.7. Гончаров А.Ю. ACCESS 2003. Самоучитель с примерами – М.: КУДИЦ-ОБРАЗ, 2004.
- 1.8. Бекаревич Ю., Пушкина Н. MS Access 2000 за 30 занятий. – СПб.: БХВ – Санкт-Петербург, 2000.
- 1.9. Каратыгин С.А., Access 2000. Руководство пользователя с примерами – Лаборатория базовых знаний, 2000.
- 1.10. Робинсон С. Microsoft Access 2000. Учебный курс. – СПб.: Питер, 2000.
- 1.11. Каратыгин С.А. Access 97. Серия «Без проблем». – М.: Бином, 1997.
- 1.12. Дженнинг Р. Microsoft Access 97 в подлиннике. – СПб.: Питер, 1997 – 2 тома.
- 1.13. Виннер Р. Microsoft Access 97. Справочник. – СПб.: Питер, 1998.
- 1.14. Горев А., Ахаян Р., Макашарипов С. Эффективная работа с СУБД. – СПб.: Питер, 1997.

2. Дополнительная литература:

- 2.1. Рудикова Л.В. Microsoft Office для студента. – СПб.: БХВ – Петербург, 2005.
- 2.2. Вейскас Д. Эффективная работа с Microsoft Access 2. – СПб.: Питер, 1997.
- 2.3. Керри Н. Прагг, Мишель Р. Ирвин. Библия пользователя. Access для Windows 95. – Киев: Диалектика, 1996.

СОДЕРЖАНИЕ

| | |
|---|----|
| Введение | 3 |
| Общие методические указания к выполнению контрольной работы | 4 |
| 1. Содержание контрольной работы | 5 |
| 1.1. Условие задания № 1 | 5 |
| 1.2. Условие задания № 2 | 5 |
| 1.3. Условие задания № 3 | 5 |
| 1.4. Условие задания № 4 | 6 |
| 2. Варианты заданий | 7 |
| 2.1. Варианты к заданию № 1 | 7 |
| 2.2. Варианты к заданиям №№ 2, 3, 4 | 8 |
| 3. Методические рекомендации к выполнению контрольной работы | 24 |
| 3.1. Методические рекомендации к выполнению задания № 1 | 24 |
| 3.2. Методические рекомендации к выполнению задания № 2 | 24 |
| 3.2.1. Создание файла БД | 24 |
| 3.2.2. Создание таблиц | 25 |
| 3.2.3. Создание связей между таблицами | 29 |
| 3.2.4. Печать схемы данных | 31 |
| 3.2.5. Заполнение таблиц данными | 32 |
| 3.2.6. Печать заполненных таблиц | 32 |
| 3.3. Методические рекомендации к выполнению задания № 3 | 33 |
| 3.3.1. Создание запросов в режиме Конструктора | 33 |
| 3.4. Методические рекомендации к выполнению задания № 4 | 38 |
| 3.4.1. Создание формы с помощью Мастера форм | 38 |
| 3.4.2. Создание отчета с помощью Мастера отчетов | 40 |
| 3.4.3. Создание управляющей формы | 41 |
| 4. Теоретический материал | 43 |
| 4.1. Первоначальные сведения о базах данных | 43 |
| 4.2. Определение базы данных (БД) | 43 |
| 4.3. Модели данных | 44 |
| 4.4. Реляционная модель данных | 44 |
| 4.5. Системы управления базами данных (СУБД) | 46 |
| 4.6. Общая характеристика СУБД ACCESS | 46 |
| 4.7. Работа с таблицами | 48 |
| 4.8. Создание связей между таблицами БД | 51 |
| 4.9. Типы отношений между таблицами | 52 |
| 4.10. Формирование запросов | 53 |
| 4.11. Проектирование форм и отчетов | 62 |
| Вопросы к зачету и экзамену | 64 |
| Литература | 66 |

Учебное издание

Составители:

Екатерина Николаевна Рубанова

Ирина Николаевна Аверина

Ирина Михайловна Гучко

ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к контрольной работе по дисциплине

«ТЕХНОЛОГИИ БАЗ ДАННЫХ И ЗНАНИЙ»

для студентов экономических специальностей

заочной формы обучения

Ответственный за выпуск *Рубанова Е.Н.*

Редактор *Строкач Т.В.*

Компьютерная верстка *Боровикова Е.А.*

Корректор *Никитчик Е.В.*

Подписано в печать 20.12.2006 г. Формат 60×84 ¹/₁₆. Бумага «Чайка».

Усл. п. л. 3,95. Уч.-изд. л. 4,25. Заказ 1155. Тираж 120 экз.

Отпечатано на ризографе учреждения образования
«Брестский государственный технический университет».

224017, Брест, ул. Московская, 267.