ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Федеральное государственное образовательное учреждение

высшего профессионального образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт градостроительства, управления и региональной экономики

Факультет управления и бизнес-технологий

Кафедра: Информационные Технологии

**КУРСОВАЯ РАБОТА**

по дисциплине: «Базы данных»

Тема: **«БД для организации, занимающейся продажей музыкальных дисков»**

Выполнили:

студенты гр. УБ 06-09

Кочубей Ю. Д.

Волконицкая М.М.

Мамонова Ю.О.

Проверила:

Покидышева Л.И.

Красноярск 2008

**СОДЕРЖАНИЕ**

**Введение**

**Задание на курсовую работу**

**Таблицы**

*Создание таблицы*

*Списки*

Фиксированные списки

Нефиксированные списки

**Схема БД**

**Формы**

*Сложные формы*

*Простые формы*

*Кнопочные формы*

**Запросы**

**Отчеты**

**Заключение**

**Приложение**

**Введение**

Для выполнения курсовой работы была выбрана СУБД Microsoft Access, которая в настоящее время является наиболее популярным средством обработки массивов информации. Для этой среды характерны простота представления и формирования базы данных, удобство обработки данных, она универсальна. Access предоставляет возможность использования SQL запросов.

**Задание на курсовую работу**

Спроектировать БД для организации, занимающейся продажей музыкальных CD-дисков. Основными моментами, которые необходимо учесть в БД, являются:

1. Перечень наименований дисков

2. Поставщики продукции

3. Сотрудники, а именно, кассиры

4. Поставки продукции

5. Продажа продукции

Данная БД, предназначенная для хранения и обработки данных о деятельности магазина, содержит информацию о товарах, связях с поставщиками, работающих сотрудниках, какие диски и в каком количестве были проданы на определенную дату, и на какую сумму реализована продукция. На основании этих данных определяется выручка от реализации, чистая прибыль, сведения о самых популярных дисках и остатках продукции на складах магазина.

БД используется сотрудниками магазина при продаже дисков и для дальнейшего анализа полученной информации о продажах.

**Таблицы**

*Создание таблицы*

Данная БД состоит из 9 таблиц:

1. Диски

2. Партии

3. Приём товара

4. Поставщики

5. Склады

6. Покупки

7. Состав покупки

8. Клиенты

9. Кассиры

Необходимо в каждой таблице БД определить первичный ключ, т.е. поле, которое однозначно определяет каждую запись таблицы. Первичный ключ должен быть уникальным, т.е. поле, которое будет являться первичным ключом, повторяться не должно. Рассмотрим каждую таблицу БД.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| код кассира | ФИО кассира | адрес кассира | телефон сотрудника | Паспортные данные |
| 1 | Иванов И.И. | Мира, 50-34 | 54-24-89 | 0304 67445 |
| 2 | Иванов И.И. | Маркса, 48-65 | 22-15-88 | 0304 67199 |
| 3 | Петров П.П. | Маркса, 48-65 | 22-15-88 | 0304 57612 |

Таблица ***«Кассиры»*** содержит следующие поля: «код кассира», «ФИО кассира», «адрес кассира», «телефон кассира», «паспортные данные». В этой таблице первичным ключом могут являться поля «код кассира» и «паспортные данные», но для удобства дальнейшего использования таблицы в базе мы выберем поле «код кассира». Остальные поля не могут являться ключевыми, так как в них данные могут повторяться. Например:

Таблица ***«Поставщики»*** имеет поля «код поставщика», «название организации», «адрес организации», «телефон организации», «ИНН организации». Здесь первичный ключ – поле «код поставщика», информация в нем не повторится (ИНН мы не выбрали по той же причине, что в таблице «Кассиры» паспортные данные). Например:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| код поставщика | название организации | адрес организации | телефон организации | ИНН организации |
| 1 | «Красный Яр» | Иванов И.И. | 54-24-89 | 0002226661 |
| 2 | «Каравай» | Петров П.П. | 22-15-88 | 7772221110 |
| 3 | «Красный ЯР» | Иванов И.И. | 22-15-88 | 4446661110 |

Из таблицы видно, что название организаций могут быть одинаковыми, сами организации могут находиться в одном здании, также как и телефон может быть «параллельным», т.е. на два адреса один номер.

Таблица ***«Дискиа»*** состоит из полей «код диска», «серия», «исполнитель», «название», «вид диска», «стиль», «год издания», «цена поставки». Первичный ключ – «код диска», так как информация, внесенная сюда, уникальна.

Таблица ***«Покупки»*** содержит поля «№ чека», «№ карты», «дата покупки», «код кассира». Здесь первичный ключ – «№ чека». Например:

|  |  |  |  |
| --- | --- | --- | --- |
| № чека | № карты | дата | код кассира |
| 1 | 5 | 12.09.07 | 1 |
| 2 | 5 | 13.09.07 | 1 |
| 3 | 6 | 12.09.07 | 2 |

В один день может быть осуществлено несколько продаж. Один и тот же кассир может обслужить несколько покупателей.

Таблица ***«Состав покупки»*** содержит поля «№ чека», «код диска», «количество дисков». Предыдущие таблицы имели аналогичные ситуации. Здесь же мы сталкиваемся с тем, что одно поле не может являться первичным ключом (информация во всех полях может повториться), т.е. здесь мы имеем составной первичный ключ, который сочетает поля «чека» и «код диска».

|  |  |  |
| --- | --- | --- |
| № чека | код диска | количество |
| 2 | 7 | 20 |
| 2 | 9 | 20 |
| 5 | 9 | 30 |

Данное сочетание полей не повторяется.

Таблицы можно просмотреть в *Приложении 1.*

**Списки**

***Фиксированные списки***

**Фиксированные списки** – фиксированный набор значений, который никогда не будет меняться.

Фиксированные списки нужны для экономии времени, чтобы при вводе информации не вводить данные, а просто выбрать необходимое из вышедшего списка.

В БД был создан фиксированный список на поле «стили» в таблице «диски», и раскрывал перечень существующих музыкальных стилей. Здесь фиксированный список необходим, так как все диски, продаваемые в данном магазине, можно разделить на группы. Также были созданы списки в полях «тип» (зарубежная, русская) и «вид диска» (альбом, сборник (CD), сборник (MP3)). Эти списки созданы по такому же принципу, как и в «стилях».

***Нефиксированные списки***

Бывают случаи, когда данные постоянно изменяются. Для таких данных существуют нефиксированные списки.

**Нефиксированные списки** – нефиксированный набор значений, причем значения в этом списке будут постоянно меняться. Эти списки создаются только на внешние ключи. Максимальное количество нефиксированных списков определяется по количеству взаимосвязей. В данной БД четыре взаимосвязи, а значит, четыре внешних ключа, следовательно, можно создать четыре нефиксированных списка. Но количество нефиксированных списков также зависит и от количества сложных форм (в данной БД две сложных формы, в которой списки делать нет необходимости). Из всего этого следует, что можно сделать шесть нефиксированных списков:

* на поле «код диска» в таблице «партии». Здесь для создания списка используем поля «код диска», «серия» и «исполнитель».
* на поля «код поставщика» и «код склада» в таблице «приём товара». Используются поля «код поставщика», «название организации» и «код склада» и «адрес склада».
* на поле «код диска» в таблице «состав продажи». Здесь для создания списка используем поля «код диска», «серия» и «исполнитель».
* на поля «№ карты» и «код кассираа» в таблице «покупки». Используются поля «№ карты», «ФИО клиента» и «кад кассира», «ФИО кассира».

В нефиксированных списках информация постоянно обновляется, этот список пополняется автоматически, и нужные данные мы будем видеть всегда перед собой.

**Схема БД**

**Схема БД** – группа взаимосвязанных таблиц.

Схему БД создают для того, чтобы в дальнейшем иметь возможность создавать формы, запросы и отчеты на основании нескольких таблиц, включенных в базу данных, необходимо определить взаимосвязи между таблицами. Связь между таблицами будет установлена путем определения главной и подчиненной таблиц. Главной будет таблица с первичным ключом, подчиненная – с внешним ключом. Возле первичного ключа должна стоять цифра «1», возле внешнего ключа – знак «∞». Если взаимосвязь проходит от простого первичного ключа к части составного, то «1» будет возле простого ключа, а «∞» возле составного. Тип взаимосвязей в данной БД – «один-ко-многим», т.е. одна запись в одной таблице может соответствовать нескольким записям в другой таблице.

При создании взаимосвязей необходимо обеспечивать целостность данных.

Схема данных предназначена для просмотра всех таблиц одновременно, их первичных ключей и взаимосвязей между ними. Схему данных можно рассмотреть в *Приложении 5*.

**Формы**

**Формы** – объект БД, в котором будет представлена информация таблиц в удобном для работы виде. Создаются они на основании таблиц. Форма удобна для ввода и редактирования записей.

***Сложные формы***

Сложная форма – форма, созданная на основании нескольких таблиц. Необходимости создания всех сложных форм нет, т.к. некоторые таблицы могут повториться несколько раз. Поэтому нужно рассмотреть все варианты и выбрать наиболее удобную. Необходимо знать центральную таблицу, которая определяется по потоку информации, который продолжается, пока не прекратится последовательность 1- ∞:

1 поток – Кассира-Покупки-Состав покупки

2 поток – Клиенты-Покупки-Состав покупки

3 поток – Склад-Приём товара-Партии

4 поток – Поставщики-Приём товара-Партии

Центральная таблица – «состав покупки», так как она находится на пересечении двух потоков информации. Так как она имеет составной первичный ключ, к этой таблице подходит наибольшее количество связей, и все они с (∞). С ней можно создать две сложные формы:

* Форма на основании таблиц «состав покупки» и «покупки»
* Форма на основании таблиц «состав покупки» и «диски»

Мы выбрали первую форму. Заносить информацию в нее нам удобней. Этот вариант несет в себе основную информацию о продажах, с ним меньше работы, при создании этой сложной формы вводимых значений будет меньше, чем во втором варианте, что позволит экономить время работы с формой. Здесь будет представлен более удобный вид для просмотра необходимой информации. Будет проще в «покупках» внести № чека, № карты, дату и код кассира и в подчиненную таблицу «состав покупки» занести код диска и количество в данной покупке. Если бы мы выбрали форму «состав покупки» - «диски», пришлось бы для того, чтобы узнать № чека открывать «покупки», перед этим внести информацию о кассирах и дате, а затем возвращаться обратно в сложную форму и там уже выбирать «код диска». Чтобы этого не делать лучше и удобней создать ***сложную форму из таблиц «покупки» и «состав покупки»***. Главной будет таблица «покупки», т.к. на взаимосвязи возле нее стоит «1», а подчиненной – таблица «состав покупки» со знаком «∞».

Второй сложной формой будет форма***, состоящая из таблиц «приём товара» и «партии»***, т.к. здесь есть поток информации 1-∞ от таблицы «приём товара» до таблицы «партии». Причем «приём товара» - главная, а «партии» - подчиненная таблица, и это так же определялось по типу взаимосвязи.

(см. *Приложение 2*)

***Простые формы***

Простые формы используется для удобного занесения простых данных, но данные формы также могут быть использованы в других формах. В данной БД пять простых форм, так как из четырех таблиц – «покупки», «состав покупки» и «приём товара», «партии» - созданы две сложные формы. Таблицы «кассиры», «клиенты», «диски», «склады» и «поставщики» будут представлены в виде простых форм.

Создать форму можно при помощи «мастера». Для этого необходимо:

1. выбрать таблицу, на основании которой будет создана форма (необходимо использовать все поля таблицы)
2. выбрать внешний вид формы (в данном курсовом проекте используется вид «в один столбец»)
3. выбрать стиль формы
4. задать имя формы

Так же необходимо в режиме «конструктор» создать кнопки перехода по записям, кнопку добавления новой записи и кнопку закрытия формы. Также в «конструкторе» пишется заголовок формы.

(см. *Приложение 2*)

***Кнопочные формы***

Главная кнопочная форма представляет собой форму, состоящую из кнопок, при нажатии на которые должны открываться созданные ранее формы.

Главная кнопочная форма создается в режиме конструктора. В «области данных» при помощи панели инструментов и панели элементов можно производить различные преобразования формы: писать заголовок, делать кнопку выхода. Затем делаются кнопки, при нажатии на которые открываются соответствующие формы. Еще на «Главной кнопочной форме» должна быть кнопка, при нажатии на которую открывается форма отчетов, которая также состоит из нескольких кнопок, при нажатии на которые открываются соответствующие отчеты.

Данная главная кнопочная форма содержит следующие кнопки: «оформление покупки», «продукция», «клиенты», «кассиры», «приём товара», «поставщики», «склады» и «отчеты».

Главная кнопочная форма должна открываться автоматически при запуске базы данных. Для этого на нужно выбрать «Сервис», «Параметры запуска», «Вывод формы», «Главная кнопочная форма».

**Запросы**

**Запросы** – это отбор нужной информации по заданным условиям. Для создания запроса необходимо сформулировать цель или задачу запроса. Необходимо также выяснить, сколько таблиц будет участвовать в запросе. В запросах можно только просматривать результат запроса, а редактировать и вносить изменения нельзя.

Можно задать отображение только нужных полей базовых таблиц. Отображаемые в запросе данные могут исходить от одной или нескольких таблиц. Названия запросов должны быть уникальны. Существующий запрос можно в дальнейшем использовать в качестве основы для создания нового запроса. Выделяют простые и сложные запросы.

После выполнения запроса отображается только те данные, которые удовлетворяют заданным условиям в табличном виде.

В данной БД было создано 12 запросов, а именно:

1. «стили», основанный на таблицах «диски», в результате выводится информация о названии дисков определённого музыкального стиля;
2. «остаток» основан на таблицах «приём товара», «партии» и «состав покупки» и предоставляет в результате выполнения информацию об остатках продукции на складах;
3. «выручка» основан на таблицах «диски», «состав покупки» и «покупки», с его помощью идёт подсчёт цены продажи (+70% к цене поставки) и выручки без учёта налогов и скидок;
4. запрос «скидка» основан на таблицах «клиенты», «покупки» и запросе «выручка», здесь мы считаем сумму скидки для каждого клиента (скидка 3%).;
5. «выручка со скидкой», выручка с учётом скидок;
6. «НДС», подсчёт НДС;
7. «выручка без НДС», выручка с учётом скидок и с вычетом НДС;
8. «затраты», подсчёт затрат на реализацию продукции (85% от выручки);
9. «прибыль», прибыль без учёта налога на прибыль;
10. «налог на прибыль», подсчёт налога на прибыль;
11. «чистая прибыль», прибыль с учётом налога на прибыль;
12. «популярный диск», информация о самом продаваемом диске (выборка по количеству проданных дисков, которое больше 1шт.).

(см. *Приложение 3*)

**Отчеты**

**Отчет** – это итоговый документ, создаваемый на основе базы данных. Отчеты можно создавать основе запросов. Данные, выдаваемые отчетом, всегда соответствуют текущему состоянию базы данных. Здесь можно сгруппировать записи по различным критериям. И для отдельных групп и для всего отчета можно выполнить необходимые вычисления. Отчет называют так же, как и запрос. Отчет – это наиболее удобная форма просмотра информации.

Отображенные в отчете значения нельзя изменить, отчет используется только для вывода данных на печать. Создаются отчеты с помощью мастера.

В данной БД было создано 5 отчетов:

* 1. выручка без НДС - выводит информацию о выручке, как за каждую покупку, так и за все покупки в целом;

2. чистая прибыль – позволяет увидеть прибыль, полученную с каждой покупки;

1. приём товара – дает информацию о количестве прибывшей от поставщиков продукции;
2. склады – информирует об остатках продукции на складах магазина;
3. схема данных – схема данных БД.

Каждый отчет имеет группировки с отступами и итоговые данные.

**Заключение**

Спроектировать БД для магазина, который занимается продажей музыкальных дисков, учитывая следующие потоки информации: поставщики, приём товара, партии и клиенты, покупки, состав покупки удалось. БД выполняет все требуемые от нее задачи. Результатом выполнения работы БД являются отчеты, с помощью которых можно наглядно просмотреть всю необходимую нам информацию.