**Федеративное агентство по образованию**

**Государственное образовательное учреждение**

**Высшего профессионального образования**

**Рязанская радиотехническая академия**

**Кафедра автоматизированных систем управления**

**Разработка базы данных для оценки неполной оплаты отгруженной продукции**

2005

**Содержание**

Аннотация

Введение

1. Назначение и область применения

1.1 Описание предметной области

1.2 Документы предметной области

2. Постановка задачи

2.1 Организационно-экономическая сущность задачи

2.2 Описание входной информации

2.2.1 Условно-постоянная информация

2.2.2 Оперативно-учетная информация

Договор

2.3 Описание выходной информации

3. Разработка информационного обеспечения задачи

3.1 Информационный анализ предметной области

3.2 Выделение информационных объектов

3.3 Определение связей и построение информационно-логической модели

4. Решение задачи в системе управления базами данных Microsoft Access

4.1 Определение логической структуры реляционной БД

4.2 Решение задачи в системе управления базами данных Microsoft Access

4.3 Обоснование выбора программных средств

4.4 Создание базы данных

4.5 Создание таблиц

4.6 Создание связей между таблицами

4.7 Ввод и редактирование данных

4.8 Создание запросов

4.9 Создание отчетов

4.10 Создание форм

4.11 Создание графиков

5. Руководство пользователя

6.Контрольный пример

Заключение

Список литературы

Приложение 1

Приложение 2

**Аннотация**

В данном курсовом проекте разработана база данных для оценки неполной оплаты отгруженной продукции и анализа состояния экономического положения описываемой фирмы при помощи СУБД ACCESS 2000, в которой содержится информация о заказчиках, изделиях, товаротранспортных накладных и платёжных требованиях.

Документ содержит описание данных, которые содержаться в разработанной базе данных, и может быть использован в качестве инструкции для применения рассматриваемого продукта.

**Введение**

Одной из проблем во взаимоотношениях поставщика с заказчиками является неполная оплата отгруженной заказчику продукции, что усложняет финансовое положение предприятия-поставщика. Своевременное получение информации о недооплате (неплатежах) путем автоматизации решения рассматриваемой задачи позволяет оперативно выяснить причины недоплаты, избежать банкротства компании.

Для эффективного контроля оплаты заказчиком отгруженной продукции необходимо вести автоматизированный учет отгрузки, учет оплаты и производить анализ оплаты с целью выявления оплаченной части продукции и недооплаты. В данной задаче необходимо иметь возможность по запросу пользователя получать результаты такого анализа для любого задаваемого в запросе изделия. При помощи графика оценить финансовое состояние организации.

**1. Назначение и область применения**

## 1.1 Описание предметной области

В предметную область (ПО) комплекса задач, к которому относится задача оценки оплаты, рассматриваются процессы планирования и учета выпуска изделий на предприятии, сдача готовых изделий на склады и отгрузка готовой продукции заказчикам в соответствии с договорами, анализ текущего запаса продукции на складах, учет и анализ полноты оплаты за отгруженную продукцию.

Задача оценки неполной оплаты отгруженного заданного изделия касается процессов отгрузки готовых изделий и оплаты их заказчиками.

Функциям, которые должны быть реализованы в задаче:

1. учет отгрузки готовой продукции;
2. учет оплаты заказчиками отгруженной продукции;
3. анализ полноты оплаты за отгруженную продукцию.

Существует ряд ограничений, существенных для процессов, которые относятся к рассматриваемой задаче:

1. вся отгрузка заказчикам осуществляется по договорам;
2. каждому договору соответствует один заказчик;
3. номер договора неизменен и уникален;
4. товаротранспортная накладная (далее ТТН) привязана к одному договору;
5. рассматриваются договора только текущего года;
6. один заказчик может иметь несколько договоров;
7. у изделия всегда только одна единица измерения;
8. номер **ПТР** соответствует одной ТТН, этот номер уникален для заказчика;
9. одной ТТН может соответствовать несколько ПТР;
10. дата оплаты отгруженной продукции определяется датой из **ПТР.**

## 1.2 Документы предметной области

Входная информация задачи «Оценка неполной оплаты отгруженного заданного изделия» разделяется на условно-постоянную и оперативно-учетную информацию. Условно-постоянная информация, необходимая для решения этой задачи, включает справочные данные о номенклатуре изделий, выпускаемых предприятием, их наименовании, единицах измерения и ценах. Эта информация отражена в справочнике готовой продукции. Входная информация, содержащая данные оперативного учета, включает данные об отгрузке и оплате. Данные об отгрузке готовых изделий заказчикам содержатся в ТТН. Этот документ об отгрузке готовой продукции всегда привязан к одному договору и, соответственно, заказчику. ТТН может содержать несколько наименований изделий, и его номер уникален для предприятия, отгружающего продукцию.

Данные об оплате заказчиками отгруженных предприятием товаров содержатся в платежном требовании (поручении). Номер этого документа уникален только для заказчика и всегда соответствует конкретной ТТН. Документ ПТР может содержать несколько наименований изделий. Одной ТТН может соответствовать несколько ПТР.

**2. Постановка задачи**

## 2.1 Организационно-экономическая сущность задачи

Одной из проблем во взаимоотношениях поставщика с заказчиками является неполная оплата отгруженной заказчику продукции, что усложняет финансовое положение предприятия-поставщика. Своевременное получение информации о недооплатах (неплатежах) путем автоматизации решения рассматриваемой задачи позволяет оперативно выяснить причины недоплаты и принять необходимые меры.

Для эффективного контроля оплаты заказчиком отгруженной продукции необходимо вести автоматизированный учет отгрузки, учет оплаты и производить анализ оплаты с целью выявления оплаченной части продукции и недоплаты. В данной задаче необходимо иметь возможность по запросу пользователя получать результаты такого анализа для любого задаваемого в запросе изделия.

Автоматизированный учет отгрузки должен осуществляться на основе данных из документа ТТН. По мере поступления ТТН на обработку эти данные должны вводиться, накапливаться и храниться в базе данных в течение регламентированного периода. Автоматизированный учет оплаты должен осуществляться на основе данных документа ПТР. Эти данные также должны накапливаться и храниться в базе данных.

На основе хранимых данных (об отгрузках и оплатах за отгруженную продукцию) по запросу пользователя должен производиться автоматизированный анализ оплаты для заданного изделия и выдача отчета. Отчет должен содержать сведения об оплаченной части отгруженных изделий одного наименования и недоплате. Кроме того, должен быть выдан список изделий по всей номенклатуре, по которым имеется недоплата.

**2.2 Описание входной информации**

Как уже было указано ранее, входная информация для данной задачи подразделяется на условно-постоянную и оперативно-учетную информацию. К условно-постоянной информации можно отнести справочник готовой продукции, а также данные о заказчиках. Формы входных документов, содержащих оперативно-учетную информацию, приведены в таблицах ниже.

### 

### 2.2.1 Условно-постоянная информация

Для расшифровки кодов изделий в процессе решения задачи, получения достоверных сведений о наименовании, единице измерения и цене каждого из изделий необходима информация из «Справочника готовой продукции, выпускаемой на предприятии» (см. табл.1). Справочник является первичным носителем этих сведений, и поэтому они должны загружаться в базу данных с него.

**Таблица 1.** Справочник готовой продукции

|  |  |  |  |
| --- | --- | --- | --- |
| Код изделия | Наименование изделия | Цена за единицу | Единица измерения |
|  |  |  |  |

К условно-постоянной информации также относится информация о заказчиках. Сведения о заказчиках (табл. 2) необходимы для расшифровки кода заказчика в ходе решения задачи и получения сведений о его наименовании и адресе.

**Таблица 2**. Сведения о заказчиках

|  |  |  |  |
| --- | --- | --- | --- |
| Код заказчика | Наименование заказчика | Адрес заказчика | Телефон |
|  |  |  |  |

### 2.2.2 Оперативно-учетная информация

Сведения, характеризующие отгрузку, находятся в ТТН (таблица 3) Для решения задачи данные из ТТН (номер документа, дата отгрузки, идентификатор заказчика, идентификаторы отгруженных изделий и их количества, а также общая сумма отгруженного) обязательно должны поступать в базу данных.

**Таблица 3**. Документ, содержащий сведения об отгрузке продукции **Товаротранспортная накладная** № Форма14

**Дата отгрузки:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Заказчик**

Наименование:

Код:

Адрес:

# Договор

Номер:

Дата:

|  |  |  |  |
| --- | --- | --- | --- |
| Изделие | | Единица измерения | Отгружено, шт. |
| Наименование | Код |
|  |  |  |  |

Остальные реквизиты ТТН относятся к справочной информации, которая к моменту решения задачи уже должна находиться в БД. Она должна поступать в БД с документов условно-постоянной информации па этапе первоначальной загрузки, так как эти документы являются первичными и соответственно достоверными носителями этой информации. Из «Справочника готовой продукции» берутся наименование изделия, единица измерения, из справочника «Заказчики» - дата заключения, наименование заказчика и его адрес.

Сведения, характеризующие оплату за отгруженные изделия, находятся в платежном требовании ПТР (таблица 4).

С документа ПТР для решения задачи в базу данных обязательно должны поступать основные данные, определяющие оплату: номер ПТР, идентификатор заказчика, оплатившего изделия, дата оплаты, номер ТТН, определяющий оплачиваемую отгрузку, идентификаторы оплаченных изделий и оплаченные количества, а также общая сумма оплаты по ПТР.

**Таблица 4**. Документ входной учетной информации по оплате

Платежное требование №\_\_\_\_\_\_\_

Дата оплаты:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ПоТТН № \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Заказчик**

Наименование:\_\_\_\_\_\_\_\_\_\_\_\_\_

Код: \_\_\_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| Изделие | | Единицы измерения | Оплачено, шт. |
| Наименование | Код |
|  |  |  |  |

Остальные реквизиты ПТР могут быть определены в базе данных по кодам из имеющейся там информации, которая должна быть загружена из справочников готовых изделий и заказчиков на этапе, предшествующем решению данной задачи. К этим данным относятся: наименование и единица измерения изделия, наименование заказчика и его адрес. Описание реквизитов входной информации приведено в таблице 5.

**Таблица 5**. Описание реквизитов входной информации

|  |  |  |
| --- | --- | --- |
| Наименование реквизитов | Характеристика | |
| входных документов | Тип | Максимальная длина |
| НТТН | Числ. | Целое (2 байта) |
| Дата отгрузки | Дата | 8 |
| Код заказчика | Числ. | Целое (2 байта) |
| Наименование заказчика зазззаказчиказаказчика | Текст | 50 |
| Адрес заказчика | Текст | 20 |
| Код изделия | Числ. | Целое (2 байта) |
| Наименование изделия | Текст | 30 |
| Единица измерения | Текст | 10 |
| Цена за (руб) кг | Денежн. | Длинное целое (8 байт) |
| Количество отгрузки | Числ. | С плавающей точкой(4 байта) |
| Сумма отгрузки | Денежн. | Длинное целое (8 байт) |
| НПТР | Числ. | Целое (2 байта) |
| Дата оплаты | Дата | 8 |
| Количество оплаты | Числ. | С плавающей точкой(4 байта) |
| Сумма оплаты | Денежн. | Длинное целое (8 байт) |

## 

## 2.3 Описание выходной информации

Выходной информацией задачи для пользователя являются два документа:

• отчет по заданному изделию, содержащий сведения об оплаченной части отгрузки и недоплате. Отчет должен быть выдан по всем заказчикам, ТТН и датам отгрузки;

• список изделий, по которым имеется недоплата. Список может содержать по всем изделиям данные об отгрузке, оплате и заказчике.

Эти документы должны выводиться на экран и печать. Формы выходных документов приведены в таблицах 6 и 7.

**Таблица 6.** Форма выходного документа задачи с результатом анализа оплат заданного изделия

**Отчет**

Оценка оплаты отгруженной части изделия

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №ТТН | Код заказчика | Дата | Отгружено, шт. | Оплачено, шт. | Недоплата |
|  |  |  |  |  |  |

**Таблица 7.** Форма выходного документа задачи, содержащая список изделий, по которым имеется недоплата

**Отчет**

**Изделия, по которым имеется недоплата**

Код изделия:

Наименование изделия:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №ТТН | Дата отгрузки | Отгружено, шт. | Недоплата | Наименование заказчика | Адрес заказчика |
|  |  |  |  |  |  |

# 3. Разработка информационного обеспечения задачи

Разработка информационного обеспечения включает подготовку документов, содержащих информацию, необходимую для решения задачи, и анализ этой информации. Такой анализ позволит произвести формализацию данных, которая имеет целью их однозначное определение для хранения, поиска и обработки во внутримашинной сфере.

Для организации информационной базы будем использовать реляционную СУБД. Поэтому должна быть разработана логическая структура реляционной базы данных, на основе которой будет осуществляться решение задачи.

## 

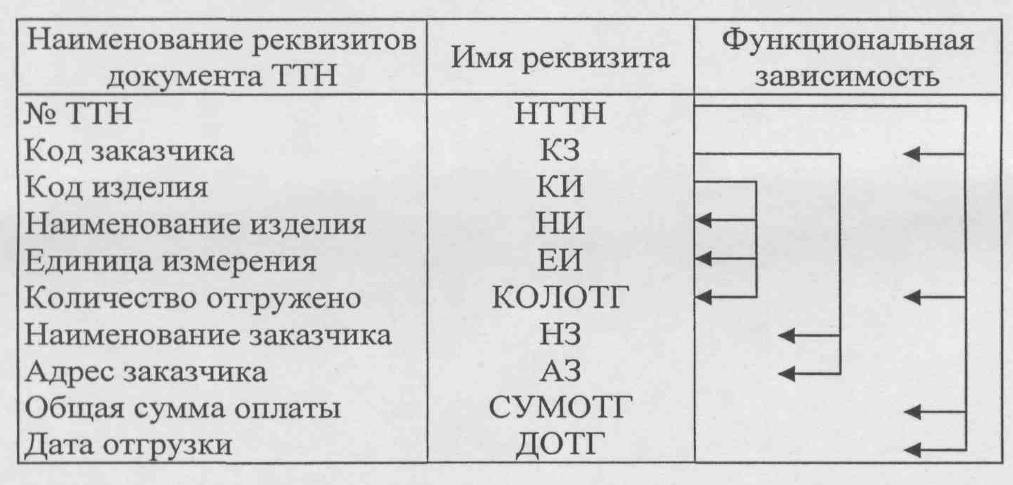
## 3.1 Информационный анализ предметной области

Произведем анализ исходной информации предметной области с целью определения состава и структуры информации для последующей формализации и построения информационно-логической модели данных. Приведенные выше формы входных документов, а также дополнительные сведения из описания предметной области позволяют определить роль реквизитов во взаимосвязанной информации, содержащейся в документе. На основе такого анализа установим функциональные зависимости реквизитов в соответствии с рекомендациями и требованиями нормализации данных.

Рассмотрим информацию, содержащуюся в документе ТТН. В заголовочной части ТТН содержатся реквизиты (номер документа, дата отгрузки, код заказчика, сумма отгрузки), общие для всего документа. Причем каждый из них имеет единственное значение, так как один документ ТТН всегда привязан к одному договору и соответственно к одному заказчику. Среди этих реквизитов особую роль играет номер ТТН***.*** Поскольку номер ТТН уникален, то есть его значение не может повториться, он выполняет роль идентификатора всей информации по отгрузке изделий, содержащейся в документе с данным номером. Остальные реквизиты заголовочной части ТТН, а также общая сумма оплаты однозначно определяются номером ТТН.

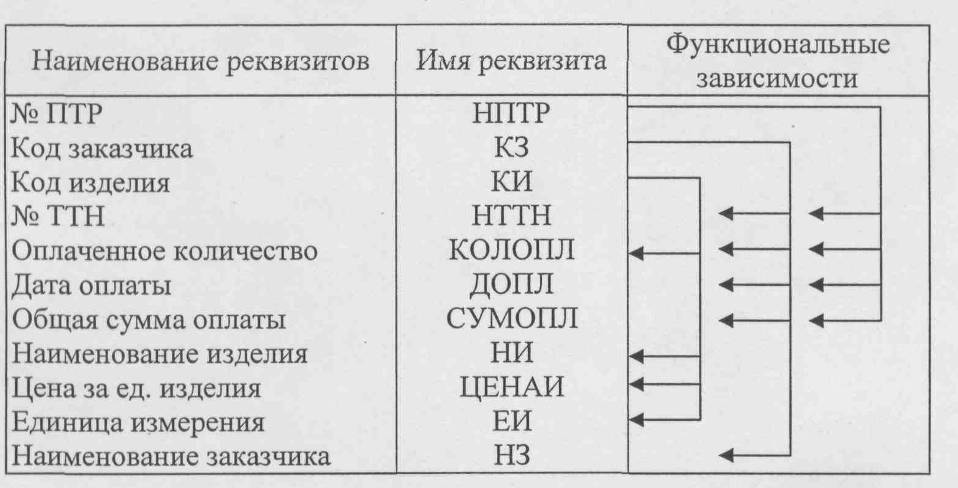
Табличная часть ТТН содержит реквизиты, имеющие множество значений в соответствующих столбцах, так как ТТН может содержать несколько наименований отгруженных изделий. Среди этих реквизитов особую роль играет код изделия***,*** который является идентификатором каждой строки документа. Таким образом, количественные данные об отгрузке, содержащиеся в каждой строке, идентифицируются (однозначно определяются) общим идентификатором документа - *номером ТТН****,*** а также *кодом изделия.*Описательные реквизиты изделия (в том числе единица измерения) и заказчика определяются однозначно их кодами. На основе проведенного анализа установим функциональные зависимости реквизитов документа ТТН и отобразим их в таблице 8.

**Таблица 8.** Функциональные зависимости реквизитов ТТН



Аналогично проанализируем информацию, содержащуюся в ПТР и «Справочнике изделий». В соответствии с описанием предметной области установим функциональные зависимости их реквизитов. Поскольку номер ПТР уникален только для заказчика, для идентификации каждого ПТР кроме номера документа используем код заказчика (или номер ТТН). Такой составной идентификатор является общим для всей информации, содержащейся в конкретной ПТР. Поскольку для каждой ПТР всегда есть ТТН, по которой оплачивается отгрузка, номер ТТН однозначно определяется идентификатором ПТР. Общая сумма оплаты также однозначно определяется этим идентификатором. В табличной части данные по оплате определяются общим идентификатором документа и кодом изделия, указанным в строке. В результате такого анализа установлены функциональные зависимости реквизитов из ПТР и «Справочника изделий», которые представлены в табл. 9.

**Таблица 9***.* Функциональные зависимости реквизитов ПТР и справочника изделий



## 3.2 Выделение информационных объектов

Выявим ключевые и описательные реквизиты на основе полученных выше функциональных.

**Таблица 10.** Соответствие описательных и ключевых рекеизитов

|  |  |  |  |
| --- | --- | --- | --- |
| Описательные реквизиты | Ключевые реквизиты | Вид ключа | Названия ИО, в кот. включ. реквизит |
| КЗ | НТТН | П,У | ТТН |
| НИ | КИ | П,У | Изделие |
| ЕИ | КИ | П,У | Изделие |
| НЗ | КЗ | П,У | Заказчик |
| АЗ | КЗ | П,У | Заказчик |
| КОЛОТГ | КИ, НТТН | С,У | Отгрузка (ОТТН) |
| СУМОТГ | НТТН | П,У | ТТН |
| ДОТГ | НТТН | П,У | ТТН |
| НТТН | КЗ, НПТР | С,У | ПТР |
| КОЛОПЛ | КЗ, КИ, НПТР | С,У | Оплата (ОПТР) |
| ДОПЛ | КЗ, НПТР | С,У | ПТР |
| СУМОПЛ | КЗ, НПТР | С,У | ПТР |
| ЦЕНАИ | КИ | П,У | Изделие |

Проанализируем, установленные функциональные взаимосвязи реквизитов установим для каждого из зависимых реквизитов, от каких реквизитов он зависит. После чего сгруппируем реквизиты, имеющие одинаковые ключи в один информационный объект. В результате получим таблицу реквизитного состава информационных объектов (таблица 10), в которой выделены все ИО, отображающие данные, которые используются в рассматриваемой задаче.

**Таблица 11.** Реквизитный состав информационных объектов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Реквизиты ИО | Признак ключа | ИмяИО | Название ИО | Описание |
| **КЗ**  НЗ  АЗ  АЗ | П,У | ЗАК | Заказчик | Данные о заказчике изделий |
| **КИ**  НИ  ЦЕНАИ  ЕИ | П,У | ИЗД | Изделие | Данные об изделии предприятия |
| **НТТН**  СУМОТГ  ДОТГ  КЗ | П,У | ТТН | ТТН | Общие сведения о ТТН |
| **НТТН**  **КИ**  КОЛОТГ | С,У | ОТТН | Отгрузка (спецификация ТТН) | Данные из ТТН по отгрузке изделия |
| **НПТР**  **КЗ**  НТТН  ДОПЛ  СУМОПЛ | С,У | ПТР | ПТР | Общие данные ПТР |
| **НПТР**  **КЗ**  **КИ**  КОЛОПЛ | С,У | ОПТР | Оплата (спецификация ПТР) | Данные из ПТР по оплате отгруженного изделия |

В данной таблице приняты обозначения:

П - простой вид ключа; С - составной ключ; У - уникальный ключ. Подчеркиванием выделены ключевые поля.

## 

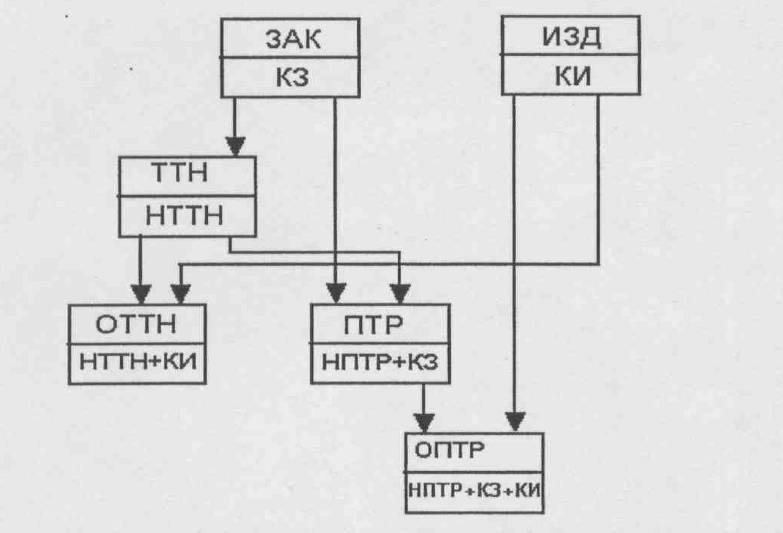
## 3.3 Определение связей и построение информационно-логической модели

Связи между выявленными информационными объектами определяются реальными отношениями между парами объектов, показанными в таблице 12. При их определении учитывались сведения из описания предметной области и семантика ИО. В частности, известно, что в одной ТТН - несколько строк по отгрузке изделий; в одном ПТР - несколько строк по оплате изделий; в одном ТТН и ПТР может быть указан только один заказчик, но для одного заказчика может быть много ТТН и ПТР, по одной ТТН может быть несколько ПТР и так далее.

**Таблица 12.** Связи информационных объектов

|  |  |  |  |
| --- | --- | --- | --- |
| **Ключ связи** | **Главный ИО** | **Подчиненный ИО** | **Тип отношения** |
| НТТН+КИ | ОТТН | Отгрузка(ОТТН) | Один ко многим |
| НПТР + КЗ+КИ | ОПТР | Оплата(ОПТР) | Один ко многим |
| КЗ | Заказчик | ТТН | Один ко многим |
| НПТР+КЗ | Заказчик | ПТР | Один ко многим |
| КИ | Изделие | Отгрузка(ОТТН) | Один ко многим |
| КИ | Изделие | Оплата(ОПТР) | Один ко многим |
| НТТН | ТТН | ПТР | Один ко многим |

Графическое изображение информационно-логической модели (ИЛМ) в канонической форме, наглядно показывающей иерархические отношения подчиненности информационных объектов, приведено на рис. 1.



**Рис. 1.**Информационно-логическая модель

Логическая структура реляционной базы данных представлена на рис.2.

Под заказчиком подразумевается предприятие, заключившее договор на поставку определенной продукции с предприятием-поставщиком. Именно поэтому заказчики и изделия стоят на верху.

**4. Решение задачи в системе управления базами данных**

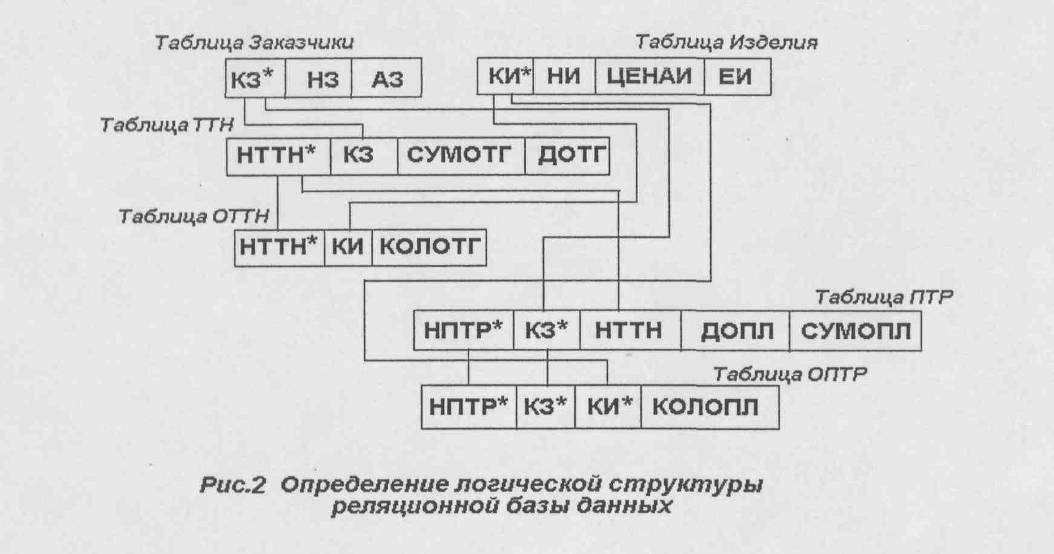
# Microsoft Access

## 

## 4.1 Определение логической структуры реляционной БД

Реляционная база данных - это БД, которая воспринимается пользователем в виде совокупности таблиц.

Логическая структура реляционной базы данных представлена на рис2:



## 4.2 Решение задачи в системе управления базами данных

## Microsoft Access

По мере поступления документов, содержащих данные об отгрузке и оплате по ней, должен осуществляться ввод этих данных в БД, где они должны храниться. Таким образом, осуществляется накопление необходимой для решения задачи оперативно-учетной информации.

При оценке оплаты необходимо обеспечить анализ оплаты по заданному пользователем изделию и получить список всех изделий, не полностью оплаченных заказчиком. В первом случае надо по запросу пользователя для задаваемого им изделия формировать отчет, содержащий сведения об оплаченной части отгрузки изделия и недоплатах по нему. Для ввода запроса пользователю в диалоге должна быть выведена экранная форма. Отчет должен выводиться на экран и принтер. Во втором случае необходимо сформировать список изделий, по которым имеется недоплата. Список должен включать изделия, по которым нет равенства количества отгруженных изделий (в ТТН) и количества оплаченных изделий (по всем ПТР, соответствующим данной ТТН). Список может содержать данные, показывающие отгрузку и оплату изделий. Список должен выводиться на экран и принтер. Расчеты в данной задаче должны производиться в натуральном выражении, в соответствии с контрольным примером. Расчеты производятся за период, в течение которого накапливалась информация. Недоплата рассчитывается как разность данных по отгрузке изделия и его оплате.

Для решения рассматриваемой задачи должен производиться ввод и накопление оперативно-учетной информации об отгрузке изделий по ТТН и об оплате отгруженных изделий по ПТР. Справочная информация об изделиях, выпускаемых предприятием, и заказчиках должна быть ранее занесена в базу данных в процессе решения других задач данной предметной области с соответствующих документов - «Справочника готовых изделий» и «Договора». Соответственно, в данной курсовой работе должны быть созданы таблицы: ТТН, ПТР, ОТТН, ОПТР, Изделий и Заказчиков

## 4.3 Обоснование выбора программных средств

Вкачестве СУБД для решения задачи курсовой работы была выбрана программа Microsoft Access 2000, входящая в пакет офисных программ Microsoft Office 2000. Этот выбор обосновывается следующими причинами:

1) Microsoft Access в данный момент широко распространена и является несложной в освоении;

2) данная задача не требует сложных вычислений (используются лишь небольшие арифметические расчеты);

3) Microsoft Access специально была выпущена на рынок для создания и управления несложными базами данных с интуитивно понятным интерфейсом.

Microsoft Access - программа, предназначенная для создания баз данных (БД) и управления ими, являющаяся весьма гибкой структурой, которая может использоваться как для работы с простыми базами данных, так и для создания весьма развитого информационного обеспечения на уровне руководства фирмы или корпорации.

Microsoft Access хорошо подходит как для создания новых БД, так и для обновления уже существующих. Программа способна импортировать данные самых разных форматов и преобразовывать их в единую форму. Microsoft Access обладает дружественным и весьма удобным графическим интерфейсом. Microsoft Access объединяет сведения из разных источников в одной реляционной базе данных. Создаваемые формы, запросы и отчеты позволяют быстро и эффективно обновлять данные, получать ответы на вопросы, осуществлять поиск нужных данных, анализировать данные, печатать отчеты, диаграммы и почтовые наклейки.

В базе данных сведения из каждого источника сохраняются в отдельной таблице. Таблица базы данных содержит сведения по определенной теме или категории объектов. Каждое поле содержит определенную его характеристику. При работе с данными из нескольких таблиц устанавливаются связи между таблицами. Для поиска и отбора данных, удовлетворяющих определенным условиям, создается запрос. Запросы позволяют также обновить или удалить одновременно несколько записей, выполнить встроенные или специальные вычисления. Для просмотра, ввода или изменения данных прямо в таблице применяются формы. Форма позволяет отобрать данные из одной или нескольких таблиц и вывести их на экран, используя стандартный или созданный пользователем макет. Разработанные на основе запросов формы позволяют одновременно просматривать и редактировать несколько таблиц, сочетая в себе производительность и удобство в применении.

Одной из основных функций баз данных является предоставление информации по различным аспектам деятельности объекта, характеристики которого содержатся в таблицах БД. Такого рода информация должна поступать вовремя и быть надлежащим образом оформлена. Отчеты позволяют извлечь из базы нужные сведения и представить их в виде, удобном для восприятия. Для анализа данных или распечатки их определенным образом используется отчет. Например, можно создать и напечатать отчет, группирующий данные и вычисляющий итоги.

Отчеты позволяют извлечь из базы нужные сведения и представить их в виде, удобном для восприятия.

## 

## 4.4 Создание базы данных

База данных Access содержит ряд объектов:

1) таблица - это основная единица хранения данных в БД.

2) Форма - объект БД, в котором разработчик БД размещает элементы управления, необходимые для ввода, отображения и изменения данных в полях таблиц.

3) Запрос - это требование на отбор данных из таблиц или выполнение определенных действий над ними. Существуют такие запросы как: запрос на выборку, на обновление, на удаление.

Новую базу данных можно создать несколькими способами:

- при запуске Access без использования Мастера;

- при запуске Access с использованием Мастера;

- в запущенном Access из меню Файл;

- с использование кнопки Новая.

Создадим базу данных без использования мастера. При запуске Access открывается диалоговое окно, в котором предлагается создать новую БД или открыть существующую. Выбираем пункт «Новая база данных» и нажимаем <ОК>. На экране открывается окно диалога «Файл новой базы данных». В списке «Папка» выбираем папку, в которой предполагается создать новую БД. В поле «имя файла» задаем имя БД. После ввода имени нажимаем на кнопку <Создать>.

На экране появляется окно базы данных. Оно позволяет получить доступ ко всем объектам БД и выбрать режим работы с объектом.

В левой части окна находится панель объектов Access, которая включает:

- Таблицы

- Запросы

- Формы

- Отчеты

- Страницы

- Макросы

- Модули

Щелкнув по ярлыку кнопкой мыши, в правой части окна получим список соответствующих объектов.

## 4.5 Создание таблиц

Таблица - основная единица хранения данных в БД, основной объект Access.

Разработка и создание базы данных осуществляется на основе ранее разработанной информационной модели. Сущности информационной модели будут являться основой для создания таблиц в БД.

В Access используются несколько способов создания таблиц:

- путем ввода данных;

- с помощью Конструктора таблиц;

- с помощью Мастера создания таблиц;

Создание таблицы путем ввода данных осуществляется вводом данных в окно с традиционной табличной формой.

Создание таблицы с помощью Мастера осуществляется путем выбора полей таблиц из определенного набора полей.

В данном курсовом проекте все таблицы создаются с помощью Конструктора таблиц. В Конструкторе нам предоставляется возможность самим задавать имена полей, их типы и свойства. Такой способ дает наиболее широкие возможности для создания необходимых таблиц.

Создадим в режиме конструктора таблицу Изделия. Для этого:

1) После открытия БД выберем слева категорию «таблицы»

2) Нажмем кнопку <Создать>. В окне «Новая таблица» выбираем вариант создания Конструктор

3) В столбце «имя поля» вводим имена полей создаваемой таблицы:

- Код изделия

- Наименование изделия

- Цена изделия

- Еденица изделия

4) В столбце «Описание» мы можем ввести поясняющий текст к полям таблицы.

5) Чтобы сделать поле «Код изделия» ключевым, выделим его и щелкнем мышью на кнопке «ключевое поле», а можно выделить поле щелкнуть правой кнопкой мыши и выбрать в раскрывающемся меню пункт «ключевое поле». (Для удаления ключа необходимо выделить это поле и щелкнуть по кнопке «ключевое поле»)

6) Завершив ввод структуры таблицы, сохраним её. Выполним команду Файл\Сохранить, здесь же введем имя таблицы Изделия. Теперь, если дважды щелкнуть на ней мышью, на экране появится таблица Изделия, внешний вид которой изображен на следующей таблице 13:

**Таблица13**. Изделия



Созданную таблицу можно закрыть и после этого ее название появится в окне созданной базы данных.

Аналогичным образом создадим таблицы: Заказчики, ТТН, ПТР, ОТТН, ОПТР. После этого в окне базы данных отобразятся названия всех созданных таблиц. Структуры всех таблиц базы данных можно найти в Приложении 1.

## 4.6 Создание связей между таблицами

Связи между таблицами будут использоваться при создании запросов таблиц при вводе данных таблицы, при создании форм и отчетов. Поля таблиц, с помощью которых устанавливается связь, могут иметь различные имена, но рекомендуется использовать одинаковые имена.

В отношении «один ко многим» (создается в том случае, когда только одно поле является ключевым или имеет уникальный индекс) главной таблицей является та, которая содержит первичный ключ и составляет часть «1» в отношении «один ко многим». Вторая таблица называется подчиненной таблицей. В данном курсовом проекте все связи будут иметь тип «один ко многим».

Внешний ключ должен иметь такой же тип информации, что и поля первичного ключа.

Для того, чтобы наиболее удобным образом можно было просматривать, создавать, удалять и модифицировать связи между таблицами, в Microsoft Access используется Схема данных. Для того, чтобы ее открыть необходимо выполнить команду Сервис\Схема данных или щелкнуть мышкой на кнопке <Схема данных>, расположенной на панели инструментов.

Рассмотрим процесс создания связей на примере создание связи «один ко многим» между таблицами Изделия и ОТТН:

1) Щелкнем мышью в пространстве схемы данных и из контекстного меню выберем команду «Добавить таблицу» или щелкнем мышью на кнопке <Отобразить таблицу>, которая расположена на панели инструментов

2) В диалоговом окне «Добавление таблицы» раскроем вкладку «Таблицы» и выберем из списка таблицу для добавления: Изделия. Затем повторим эти действия для таблицы ОТТН.

В данном отношении главной таблицей является таблица Изделия, она содержит первичный ключ «Код изделия» и составляет часть «один». Таблица ОТТН является подчиненной. В ней также содержится поле «Код изделия», но здесь оно является внешним ключом. Это поле имеет такой же тип информации, что и поле первичного ключа главной таблицы.

3) Для создания связи щелкнем мышью на поле главной таблице Изделия, по которому хотим осуществить связывание, в данном случае на поле «Код изделия», и, не отпуская кнопку мыши перетащим его к аналогичному полю подчиненной таблицы ОТТН. После чего на экране появится диалоговое окно «Изменение связей».

При создании связей между таблицами можно обеспечить целостность данных.

Целостность данных означает систему правил, используемых для поддержания связей между записями в связанных таблицах, а также обеспечивает защиту от случайного удаления или изменения связанных данных. Чтобы обеспечить целостность данных, необходимо в окне «Изменение связей» установить флажок в поле «Обеспечение целостности данных».

4) В диалоговом окне «Изменение связей» установим флажок в поле «Обеспечение целостности данных» . После этого нажимаем на клавишу «ОК>. Мы увидим, что на схеме данных отразилась созданная нами связь. Подобным образом создадим связи между другими таблицами.

Полная схема данных отображена в Приложении 1.

## 

## 4.7 Ввод и редактирование данных

Реализацию процесса ввода и редактирования данных приведем на примере таблицы Изделия.

- Выбираем в списке таблиц таблицу Изделия и дважды щелкаем по ней мышью. Перед нами открывается окно таблицы.

- Вводим информацию в поле. Указатель в виде треугольника -указатель выбора записей. \* - появляется в новой пустой записи в конце таблицы. + - означает, что с этой таблицей есть связанные таблицы.

- Чтобы изменить уже существующую запись, достаточно поместить указатель мыши на поле, в котором необходимо изменить данные. После этого вводить данные, а затем нажать «ТаЬ» , для перехода на следующее поле. Окончательно данные будут изменены только после перехода на другую запись. До тех пор, пока работа выполняется над одной записью, все изменения можно отменить, нажав клавишу <Езс>.

Есть определенные правила работы со связанными таблицами:

- Нельзя удалять запись из главной таблицы, если существует запись в связанной с ней таблице.

- Нельзя изменить значение первичного ключа в главной таблице, если есть записи, связанные с данной записью в подчиненной таблице.

- Нельзя ввести в поле внешнего ключа связанной таблицы значение, не содержащееся в ключевом поле главной таблице.

Чтобы правила соблюдались, необходимо установить «Обеспечение целостности данных». Чтобы преодолеть эти ограничения следует установить «Каскадное обновление связанных полей», что означает, что при изменении значения ключевого поля главной таблице, автоматически меняются внешние ключи в связанных таблицах. «Каскадное удаление связанных полей» - автоматическое удаление записей в подчиненных таблицах, связанных с записью главной таблицы.

## 4.8 Создание запросов

Запросы используются для просмотра, анализа и изменения данных из одной или нескольких таблиц. Сам запрос не содержит данных, но позволяет выбирать данные из таблиц и выполнять над ними операции.

Наиболее часто применяемый тип запросов - запрос на выборку, осуществляющий выборку данных из таблиц.

Для создания запроса в режиме Конструктора необходимо выполнить следующие действия:

1) В окне базы данных выбираем категорию «Запросы» и нажимаем кнопку «Создать» на панели инструментов.

2) Выбираем вариант создания Конструктор и нажимаем <ОК>.

3) Выбираем таблицу, информацию по которым хотим выводить в результате запроса, и нажимаем кнопку «Добавить». Затем выбираем другую таблицу и тоже нажимаем «Добавить» и т.д. В верхней части окна запросов отражается схема данных запроса. В нижней части окна расположен бланк запроса (см. Приложение 2).

В запрос не следует включать все поля таблиц, нужно включать только интересующие нас.

В запросе устанавливаются критерии, которым должны удовлетворять отбираемые записи. Так можно произвести сортировку записей (например, по дате заказа).

Чтобы удалить таблицу или запрос, необходимо в верхней части окна запроса выделить удаляемую таблицу или запрос, и нажать клавишу Del.

Запросы добавления используются для добавления записей из одной таблицы в другую. Запросы обновления позволяют изменять некоторые значения в таблице. Запросы, позволяющие создавать таблицы, удалять значения, изменять обновление называются модифицирующими.

Исходя из требований, предъявляемых к информационной системе, реализуем следующие запросы с целью получения определенной информации:

1) Оценка оплаченной части. Этот запрос позволяет получить данные о недоплате по заданному изделию. На примере данного запроса рассмотрим процесс построения запросов. Данный запрос будет содержать следующие поля:

Код \_изделия, №ТТН, Код\_заказчика, Дата\_отгрузки, Кол\_отг\_изделия, Кол\_опл\_изделия, Цена\_изделия

Так же в запросе будет содержаться вычисляемое поле со значением НедоплатаСум:(([ОТТН]![Кол\_отг\_изделия-[ОПТР]![Кол\_опл\_изделия])\*[Изделия]![Цена\_изделия])

Для построения этого вычисляемого поля необходимо проделать следующие действия:

- В окне конструктора запроса перейдем на строку «поле» пустого столбца бланка и нажмем кнопку <Построить> на панели инструментов или щелкнем правой кнопкой мыши и в раскрывающемся списке выберем «Построить»

- Открывается специальное окно построителя выражений. В окне построения выражений набираем недоплата. Слева внизу показываются типы объектов: запросы, таблицы и т.д. выделяем «таблицы» и выбираем таблицу ОТТН, затем в появившемся списке полей щелкаем два раза мышью на поле «Кол\_опл\_изделия», потом на знаке «-» и т.д.. В окне построения выражений будет все видно. После этого <ОК>.

Во всех полях запроса устанавливается флажок в строке «Вывод на экран. В этом запросе приведена информация, касающаяся частично оплаченной отгруженной продукции. Структура запроса Оценка оплаченной части отгрузки изделия расположена в Приложении 2. По такому же принципу построим и остальные запросы.

2) Недоплата за товары. Этот запрос позволяет получить данные обо всех изделиях, по которым имеется недоплата. Запрос будет содержать следующие поля: Код\_изделия, Наименование\_изделия, Цена\_изделия, Наименование\_заказчика, НедоплатаСум.

Недоплата из первого отчета с применением групповой операции , т.е. среднее значение.

3) Фирмы должники. Этот запрос позволяет получить информацию о предприятиях-должниках и о сумме их долга. Запрос будет содержать поля:

Код\_заказчика, Наименование\_заказчика, а также вычисляемое поле: Sum\_Недоплата: Недоплата с применением групповой операции Sum.

4) Долги фирм. Этот запрос позволяет получить информацию о том, долги по какому изделию имеются и каким заказчиком.

Запрос будет содержать поля:

№ТТН, Код\_изделия, Колл\_отг\_изделия, Кол\_опл\_изделия, Код\_заказчика,Цена\_изделия, Дата\_отгрузки, Недооплата.

Во всех столбцах ставим флажки в строке «Вывод на экран». Результаты выполнения запроса можно увидеть, переключившись в режим таблицы с помощью кнопки <3апуск> на панели инструментов. Если нужно изменить запрос, необходимо открыть его в режиме Конструктора и внести необходимые изменения.

Чтобы удалить таблицу из изменяемого запроса, необходимо в верхней части окна запроса выделить удаляемую таблицу или запрос, выбрав любое место в списке полей, и нажать клавишу <OK>.

## 4.9 Создание отчетов

Отчет представляет собой эффективный способ представления данных в печатном формате. Имея возможность управлять размером и внешним видом всех элементов отчета, пользователь может отобразить сведения желаемым образом.

Большая часть сведений в отчете поступает из базовой таблицы, запроса или инструкции SQL, являющейся источником данных для отчета. Остальные сведения отчета хранятся в его структуре.

В данном курсовом проекте будем использовать отчеты в формах для вывода результатов запросов. Результаты запроса удобнее выводить в виде отчета, чем в виде таблицы, так как он предоставляет пользователю более наглядное представление выводимых данных.

Существует три способа создания отчета:

Создание отчета с помощью Автоотчета. После выбора источника записей и макета (в столбец, ленточный или табличный) Автоотчет создает отчет, который использует все поля источника записей и применяет последний использованный автоформат.

Создание отчета с помощью мастера. Мастер задает подробные вопросы об источниках записей, полях, макете, требуемых форматах и создает отчет на основании полученных ответов.

Для создание отчета с помощью мастера необходимо выполнить следующие действия:

1. В окне базы данных выберите значок Отчеты списке Объекты.

2. Нажмите кнопку <Создать> на панели инструментов окна базы данных.

3. В диалоговом окне Новый отчет выберите нужного мастера. Описание действий, выполняемых мастером, выводится в левой половине диалогового окна.

4. Выберите таблицу или запрос, содержащие данные, на которых должен быть основан отчет.

Примечание: Microsoft Access по умолчанию использует эту таблицу или запрос как источник данных для отчета. Однако мастер позволяет изменить источник данных, а также выбрать поля из других таблиц или запросов.

5. Нажмите кнопку <ОК>.

С помощью перечисленных действий создадим отчеты:

«Долги», «Фирмы-должники», «Недоплата за товары», «Оценка оплаченной части», «Анализ». Данными для этих таблиц будут являться соответствующие запросы.

## 4.10 Создание форм

Форма - объект БД, в котором разработчик БД размещает элементы управления, необходимые для ввода, отображения и изменения данных в полях таблиц.

Большая часть сведений в форме поступает из базового источника записей. Остальные сведения формы хранятся в ее структуре.

Связь между формой и источником данных для нее создается при помощи графических объектов, называемых элементами управления. Наиболее часто используемым для вывода и ввода данных элементом управления является поле.

Существует несколько способов создания форм:

- с помощью Конструктора;

- с помощью Мастера;

- с помощью Автоформы;

Рассмотрим процесс создания форм с помощью Конструктора. Для создания формы в режиме Конструктора необходимо выполнить следующие действия:

В окне базы данных выделяем объект «Формы». Затем нажимаем на кнопку <Создать>. Выбираем вариант создания Конструктор и нажимаем<ОК>. И в открывшемся окне конструктора форм приводим форму к желаемому виду, изменяя цвета, размеры и размещая на ней различные элементы управления. Мы будем создавать следующие формы:

- Главная (содержит кнопки перехода к соответствующим формам)

- Заказчики

- Изделия

- ТТН

- ПТР

- ОТТН

- ОПТР

Эти формы будут использоваться для просмотра и ввода данных в соответствующие таблицы.

Внешний вид формы ТТН представлен в Приложении 2.

Теперь сделаем так, чтобы кнопочная форма «Главная» автоматически загружалась во время открытия базы данных. Для этого необходимо выполнить следующие действия:

1. Откройте базу данных.

2. Открываем пункт меню «Сервис» → «Параметры запуска»

3. В выпадающем списке «Вывод форм/страниц:» выбираем таблицу «ГЛАВНАЯ» и ОК

При следующем открытии базы данных указанная кнопочная форма («ГЛАВНАЯ») откроется автоматически.

База данных Access содержит ряд объектов:

1) таблица - это основная единица хранения данных в БД

2) форма - объект БД, в котором разработчик БД размещает элементы управления, необходимые для ввода, отображения и изменения данных в полях таблиц

3) запрос - это требование на отбор данных из таблиц или выполнение определенных действий над ними. Существуют такие запросы, как запрос на выборку, на обновление, на удаление.

**4.11 Создание графиков**

Объект размещения графиков – отчет .С помощью графиков дается анализ о проделанной работе. Создание графика с помощью конструктора. Открывем окно конструктора затем нажимаем кноп «Вставка » на главной панели и вибираем оттуда Диаграмма. Выбираем место, где она будет размещена, выбираем таблицу или запрос для создания диаграммы. Выбираем поля с данными, которые необходимо отразить в диаграмме. Выбираем тип диаграммы и тип отображения данных на диаграмме. Далее даем название диаграмме и нажимает готово.

# 5. Руководство пользователя

В начале работы пользователя с базой данных автоматически на экране появляется главная кнопочная форма, содержащая основное меню. Основное меню содержит шесть кнопок для открытия таблиц: <Заказчики>, <Изделия>, <ТТН>, <ПТР>, <ОТТН>, <ОПТР>, а также по пять кнопак для просмотра и печати отчётов.

При нажатии на кнопку <Заказчики> открывается форма, содержащая сведения о заказчиках, которые отображаются из таблицы Заказчики. В открывшейся форме можно добавить запись, удалить запись. Форма также содержит кнопку «Выход».

Аналогично открываются формы, содержащие сведения из таблиц.

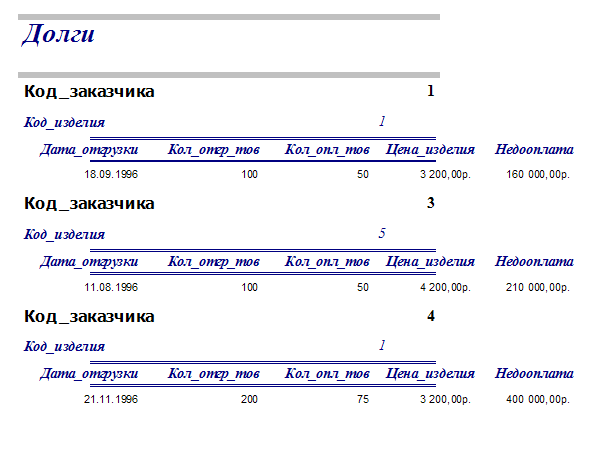
Для просмотра и печати созданных отчётов необходимо нажать на кнопки с соответствующими символами отчёта.

**6. Контрольный пример**

В качестве контрольного примера продемонстрируем решение поставленной задачи с использованием исходных данных, приведенных в приложении, подтвердив тем самым работоспособность разработанной базы данных и правильность формируемых ею выходных данных.

Из таблиц ТТН и ПТР видно, например, что по товаротранспортным накладным с номером 0024, 0025 и 0050 есть недоплата, так как общая сумма оплаты меньше суммы, на которую был отгружен товар. Значит, при правильной работе программы в запросе «Долги фирм» по номеру ТТН следует ожидать появления записей, отражающих сумму недооплаты по товаротранспортным накладным с номерами 0024, 0025 и 0050 соответственно заказчиками 1, 3 и 4. Отчёт по долгам, содержащий результаты соответствующего запроса, представлен на рис.З

.

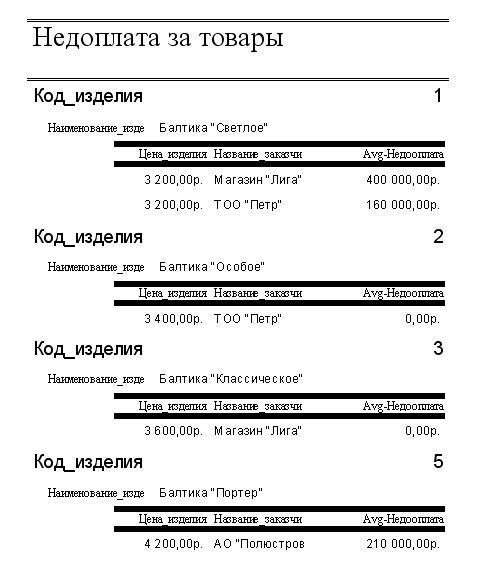


**Рис.3.** Отчёт запроса по долгам

Из рисунка видно, что заказчику 4 (магазин «Лига») 21 ноября 1996 года было отгружено 200 единиц изделия с кодом 001, а оплачено было 75. Соответственно, недоплату по данному изделию можно найти:

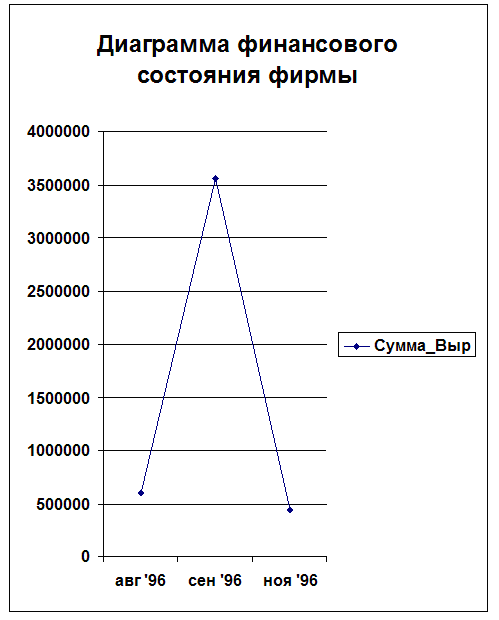
([Кол\_отг\_изделия-[Кол\_опл\_изделия])\*[Цена\_изделия]=(200-75)\*3200=400000, что совпадает с данными, предоставляемыми программой.

Информацию по недоплаченным изделиям можно получить в отчете «Недоплата за товары». Содержание соответствующего отчета представлено на рис.4.



**Рис. 4.** Список изделий, по которым имеется недоплата

Анализ состояния фирмы:



На графике отображающем состояние финансового положения фирмы по месяцам видно что с августа по сентябрь фирма заработала (3500000 – 600000=2900000) р . Значит в этом месяце были заказы с полной оплатой отгруженой продукции или заказы с небольшим кредитом. В следующем месяце заметно снижение следовательно были заказы с большим кредитом. Точка на графике в сентябре остановилась на отметке 400000 значит необходимо отменять сделки при которых существует кредит до периода возврата предыдущих долгов или до сделок с полной оплатой либо оплатой больше 50% продукции, иначе граафик пойдет ниже, и пересечет прямую 300000. Это говорит о том, что в следующем месяце у фирмы не будет средств на производство продукции, зарплату, произойдет банкротство.

# Заключение

Таким образом, поставленная задача решена. Разработанная база данных позволяет на основе хранимых данных (об отгрузках и оплатах за отгруженную продукцию) по запросу пользователя производить автоматизированный анализ оплаты для заданного изделия и выдавать соответствующий отчет. Отчет содержит сведения об оплаченной части отгруженных изделий и недооплате. Кроме того, выдается список изделий по всей номенклатуре, по которым имеется недооплата. Также для удобства пользователя программа формирует отчет, содержащий информацию о заказчиках-должниках и общей сумме недоплаты. В итоге проведенной работы был дан анализ состояния фирмы по месяцам.

# Список литературы

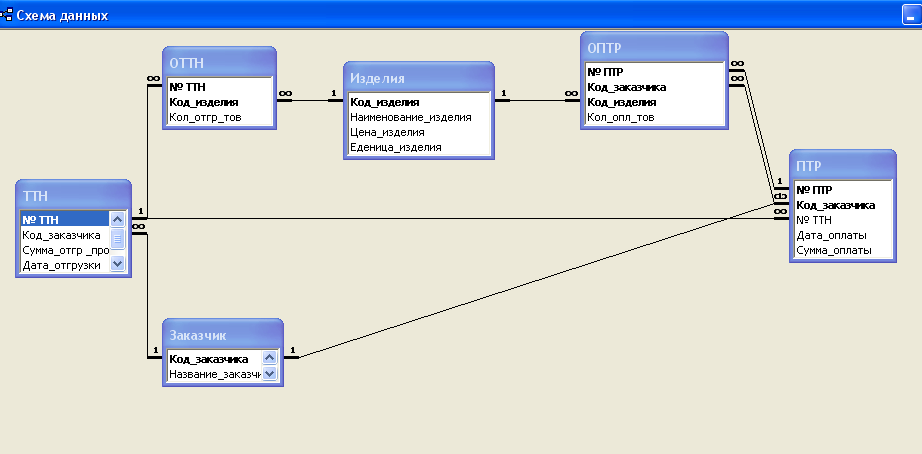
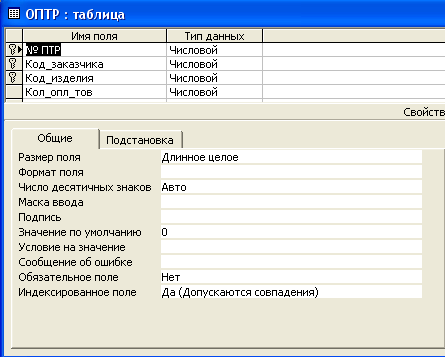
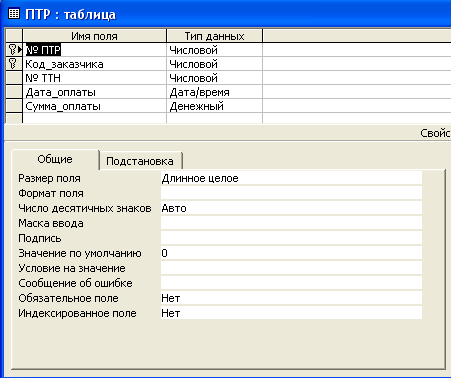
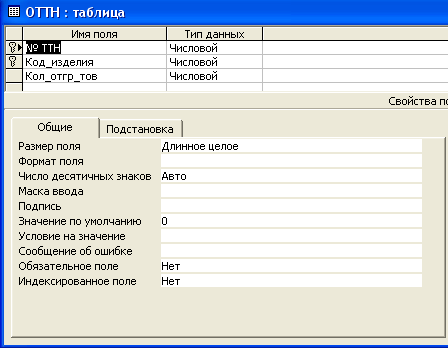
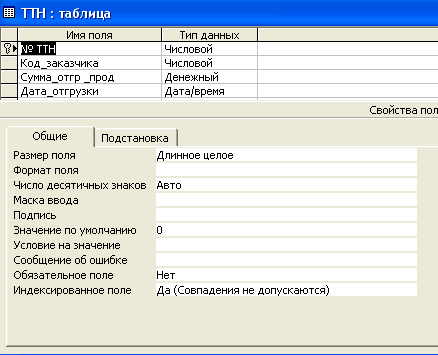
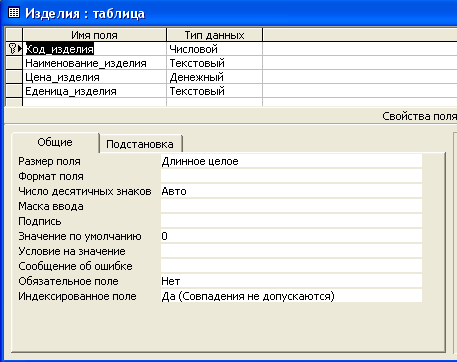
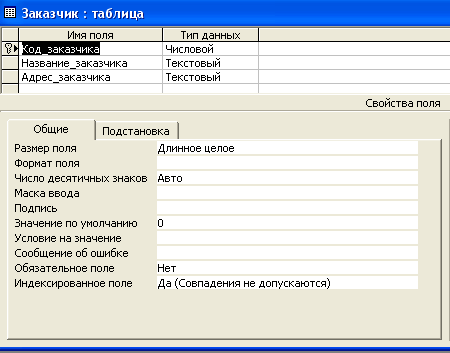
1) Каратыгин С. Microsoft Access 2000 на примерах. Руководство пользователя; М.: Лаборатория Базовых знаний, 2000;

2) Справочная система: Microsoft Access 2000;

3) «Экономическая информатика». Учебник для вузов, под редакцией Евдокимова В.В., С-Пб.: 1997;

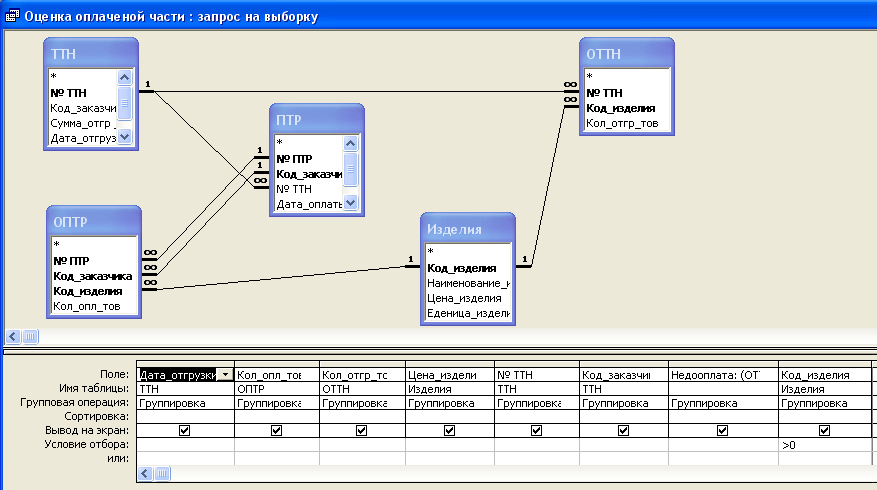
# Приложение 1

Структуры таблиц базы данных:



# Приложение 2

Структура запроса на выборку сведений об оценке оплаченной части :



Формы:

