Департамент образования и науки города Москвы

Государственное бюджетное профессиональное образовательное учреждение

города Москвы

**«КОЛЛЕДЖ ЖЕЛЕЗНОДОРОЖНОГО И ГОРОДСКОГО ТРАНСПОРТА»**

**КУРСОВОЙ ПРОЕКТ**

На тему: Разработка базы данных «Банкоматы» для анализа операций обналичивания денег клиентами в банкоматах различных банков

По предмету: МДК 11.01 Технология разработки и защиты баз данных

специальность: 09.02.07 Информационные системы и программирование

Курс 2 группа 4ИС3010

**Выполнил студент: Проверил преподаватель:**

Федотов Виктор Викторович Кроттер Андрей Викторович

Москва

2022

Содержание

Введение

1. Теоретическая часть
   1. Описание предметной области

Существует несколько кредитных организаций, объединённых в одну банковскую систему. Банковская система – это совокупность нескольких банков и других кредитных учреждений.

Многие банковские системы работают таким образом, что операции производятся в специальный терминалах, именуемых банкоматами.

Банкоматы – это специальное устройства для сбора и выдачи наличных денежных средств. Каждое такое устройство имеет связь с банком, через которую в банк поступает некоторая информация о сумме снятых денежных средств и пользователе банкомата.

Каждый пользователь банкомата имеет уникальную карточку клиента банка, которая регистрируется в отделении банка и позволяет совершать операции с банкоматами и позволяет идентифицировать пользователя, который в данный момент производит операцию с банкоматом.

Операция в независимости от типа может иметь комиссию при условии того, что операция проводится в банкомате пользователем, который регистрировал свою карточку клиента в банке, к которому не относится банкомат.

Когда информация о произведённой операции передаётся в центральный банк, то она анализируется специальными сотрудниками, каждый отдел сотрудников занимается анализом своих данных, которые затем вручную заполняют отчёты.

Чтобы анализировать данные она собирается последовательно, после чего распределяется между отделами и структурируется. Такой процесс занимает достаточно большое количество времени и материальных ресурсов, таких, как бумага.

* 1. Постановка задачи

Чтобы избежать проблем с эффективностью обработки информации, которая поступает с банкоматов и повышенной траты материальных ресурсов, можно использовать базы данных.

База данных – это структурированное хранилище данных, которая может содержать данные в виде разделённой информации, которую можно использовать для обработки любых данных.

В случае с данной проблемой банковской системы преимущество использования баз данных, заключается в повышенной эффективности обработки и анализа операций. База данных способна принимать данные не только в ручном, но и в автоматическом режиме в случае подключения базы данных к системе, которая получает данные с банкоматов.

База данных будет проектироваться в СУБД MS Access, которая позволяет не только хранить данные структурированной, но и формировать отчёты в удобном для пользователя виде, которые можно распечатать в случае необходимости.

Каждый отчет может быть создан на основе как целой таблицы, так и отдельного запроса. Запросы в базы данных занимаются обработкой базы данных. Существуют такие операции, как выборка, удаление, изменение, добавление данных. Фактическая задача запроса – это действие с данными.

В данной базе данных должна содержаться информация, которая позволит анализировать операции с банкоматами, сюда можно информацию об операциях, которая включает в себя тип операции, конечную сумму операции, выданную пользователю, наличие комиссии (да или нет), указание валюты операции и даты совершения. Именно эти данные подвергаются анализу со стороны работников банковской системы.

Помимо информации об операциях нужно хранить так же информацию о банкоматах, чтобы можно было разделить операции по конкретным банкоматам.

Соответственно, банкоматы принадлежат какому-то банку, нужно хранить какую-то информацию о банках, которая позволит идентифицировать банк в базе данных.

Банк, как и банкомат имеет физическое расположение, хранить адреса так же можно для поиска уникальных идентификаторов банков и банкоматов для ускорения поиска в базе данных.

При проектировании базы данных следует учитывать следующее:

* Каждый банк имеет множество банкоматов.
* Банкомат может быть закреплён только за одним банком.
* Банкомат может находиться как в банке, так и вне его.
* Банкомат может совершать операции в нескольких валютах.
* Банкомат может выполнять несколько типов операций.
  + 1. Запросы базы данных.

Для того, чтобы упростить анализ операций в базе данных нужно иметь доступ к конкретным данным по запросам. В качестве аналитических запросов могут понадобится следующие:

* Операции банкомата за промежуток – это запрос, который возвращает несколько записей, состоящих из кода банкомата, кода карты клиента, даты совершения операции, наличии комиссии (да или нет), сумму выдачи денежных средств, наименование валюты в которой была совершена операция, уникальный идентификатор операции. Параметры, требуемые от пользователя – код банкомата, а также две даты, между которыми требуется собирать информацию. Цель запроса – отбрасывание ненужных записей, создание на основе него запроса.
* Операции банкоматов за промежуток – это запрос, который возвращает несколько записей, состоящих из кода банкомата, кода операции, кода карты клиента, даты совершения операции, наличия комиссии (да или нет), сумму выдачи денежных средств клиенту, а также наименование валюты, в которой совершалась операция, и тип операции. Данный запрос отличается от предыдущего только тем, что возвращает данные не конкретного банкомата, а всех банкоматов за определенный промежуток времени. Не требует ввод кода банкомата. Не используется в отчётах.
* Операции банкоматов за промежуток в валюте – это простой запрос, который отличается от предыдущего тем, что добавляется возможность фильтровать данные по валюте в которой производится операция. Не используется в отчётах.
* Операции банкоматов за промежуток, где сумма выдачи – это запрос, который возвращает все данные запроса “Операции банкоматов за промежуток”, которые больше количества денежных средств, которое вводит пользователь. Принимает на вход даты от какой собирать информацию и до какой, а также ввод суммы от пользователя.
* Операции за промежуток определённого типа – это запрос, возвращающий несколько записей, состоящих из кода банкомата, кода карты клиента, даты совершения операции, наличии комиссии (да или нет), сумму выдачи денежных средств, наименование валюты в которой была совершена операция, уникальный идентификатор операции. Требует ввода двух дат, одна из которых является начальной, а другая конечной. Поиск осуществляется между датами, а также пользователь должен ввести наименование совершаемой операции записи которой собирается получить.
* Операции за промежуток, сумма – это запрос, возвращающий несколько записей, состоящих из кода банкомата, кода карты клиента, даты совершения операции, наличии комиссии (да или нет), сумму выдачи денежных средств, наименование валюты в которой была совершена операция, уникальный идентификатор операции. Позволяет выбрать тип операции, который требует пользователь, а также возвращает сумму выдачи всех банкоматов.
* Операции клиента за промежуток – запрос, созданный специально для вывода информации на отчёт. Возвращает код банкомата, в котором была совершена операция, дату совершения операции, наличии комиссии (да или нет), сумму выдачи клиенту, наименование выданной валюты, название операции со счетом. Требует ввод даты начала и конца сбора записей, а также серию и номер паспорта клиента.
* Поиск банка по названию и адресу – вспомогательный запрос, который возвращает код банка в системе. При этом на ввод требуется точное название банка и его расположение, которое включает в себя страну, область, город, округ, улицу, дом, корпус.
* Поиск банкомата по адресу – запрос, выводящий несколько банкоматов, расположенных по выбранному адресу. Требует ввод адреса, включающего в себя область, город, округ, улицу, дом, корпус), возвращает при этом сам адрес и код банкомата.
* Поиск клиента – запрос, который получает серию и номер паспорта клиента и возвращает его уникальный идентификатор в системе (номер карты).
  + 1. Формы базы данных.

Формы внутри баз данных существуют для удобной работы с базой данных, они могут использоваться для удобного ввода данных в запросы или редактирования таблиц в целом. Каждая форма может содержать свои элементы ввода и вывода данных, свою логику обработки. Например, возможно сделать заполнение отчёта через ввод данных на форме.

База данных для анализа операций с денежными средствами должна содержать следующие формы:

* Ввод адресов – форма, которая используется для ввода и редактирования общих адресов в базе данных. Нужен для ввода адресов различных местоположений.
* Ввод банков – форма, используемая для ввода и редактирования информации о банках-участниках банковской системы.
* Ввод банкоматов – форма, которая используется для ввода и редактирования информации о банкоматах.
* Ввод валют – форма, используемая для ввода валют, в которых может быть совершена операция.
* Ввод клиентов – форма, для ввода и редактирования клиентов в базу данных при регистрации карт для работы с банкоматами.
* Ввод типов операции, форма для ввода и редактирования типов операций над денежными средствами.
* Формирование отчётов – специальная форма для управления отчётами.
  + 1. Отчёты базы данных.

Отчёты используются в базе данных, как лёгкий способ получения информации в определённом виде. Отчёт может быть создан, как на основе целой таблицы, так и на основе запроса.

В данной базе данных нужно создать следующие отчёты:

* Операции банкомата за промежуток – показывает все операции конкретного банкомата. Должен включать в себя номер операции, код карты клиента, дату операции, сумму выдачи, наименование валюты, которая была выдана и наименование операции с денежными средствами.
* Операции клиента за промежуток – показывает все операции конкретного клиента за определённый промежуток времени. В информацию входит название операции, код банкомата, дата выдачи, наличие комиссии и наименование валюты, в которой была совершена операция.

1. Практическая часть
   1. Разработка функциональной модели

Разработка функциональной модели – это один из процессов проектирования базы данных, который предполагает описание информации, хранящейся в базе данных.

Чтобы создать функциональную модель нужно проанализировать задачу и описать основную информацию, которая нужна для анализа и обработки данных, поступающих с банкоматов, определение наличия и расчёта комиссии при обналичивании денежных средств, а также различную дополнительную информацию.

* + 1. Информация об операциях

Так как основная задача базы данных – это анализ операций обналичивания денежных средств с банкоматов, то и информация, на которой базируется данная система должна быть об операциях банкоматов.

Любая операция в банкомате – это прежде всего операция с каким-то количеством денежных средств. Количество и будет являться основными данными при анализе.

В банкоматах проводятся различные операции, а не только операции обналичивания денежных средств, поэтому база данных должна хранить в себе информацию о типах операций.

Чтобы проанализировать данные с операциями не будет достаточно лишь типа операции и суммы выданный денежных средств. Для эффективного анализа может понадобится возможность выбора информации за конкретный промежуток времени или в конкретное время. Поэтому нужно сохранять данные о дате и времени совершения операции.

Операция может проводиться не только в рублях, но и в другой валюте. Чтобы не допустить некорректного анализа нужно будет также хранить информацию о валюте, в которой была произведена операция.

Согласно механизму работы данной банковской системы, в каждой операции также должна быть информация о наличии комиссии.

Каждая из операций имеет уникальный идентификатор и с указанием какой карты была произведена операция и какой банкомат её проводил.

* + 1. Информация о банкоматах

Каждая операция производится с использованием банкомата. Соответственно для более правильного анализа мы должны сохранять информацию о банкоматах.

Так как для анализа операций не так важно какие свойства и составляющие будет иметь банкомат, достаточно хранить данные о местоположении банкомата.

Каждый банкомат должен иметь свой уникальный идентификатор.

* + 1. Информация о банках

Чтобы определить наличие комиссии в операции, нужно понять какому банку принадлежит банкомат, поэтому мы должны хранить информацию о банках.

Для определения будет достаточно уникального идентификатора банка в системе, но для обработки данных нужно будет вести основную информацию о банке. Такой информацией будет выступать название банка и его адрес. Всё это дополнительная информация, но она не будет занимать много места в базе данных, так как фактическое количество банков-участников в банковской системе не велико.

* + 1. Информация о клиентах

Каждая операция относится к конкретному клиенту банка. Хранение информации о клиентах банка, которые используют систему безналичной оплаты позволяет делать отчёты об операциях конкретного клиента.

О клиенте должны храниться базовые данные, такие как паспортная информация (имя, фамилия, отчество, серия и номер паспорта), а также код банка, к которому привязана карта. Данная информация преимущественно нужна для отчетов и выборки операций определённых клиентов.

* 1. Инфологическое проектирование

Следующий процесс разработки базы данных – это инфологическое проектирование. Оно предполагает структурирование информации на некие группы данных (сущности) и их атрибуты (части сущностей, которые имеют данные определённого типа внутри) в базе данных таким образом, чтобы можно было максимально эффективно работать с базой данных.

Помимо основной информации сущность должна иметь дополнительную информацию для обеспечения связей между сущностями при логическом проектировании.

Можно выделить несколько групп:

* Операции – это основные данные для анализа, которые были получены с банкоматов. Является основной группой для выполнения анализа данных. В группу входят атрибуты: код операции (уникальный код операции), код карты (код клиента или его карточки), код банкомата (идентификатор банкомата в котором была совершена операция), дата (дата совершения операции), комиссия (наличие комиссии, должен быть выбор из вариантов: да, нет), сумма выдачи (сумма, выданная клиенту, либо сумма, которая была полученная от клиента), код валюты (идентификатор валюты), код типа (идентификатор типа операции).
* Валюты, содержит в себе наименование валюты и знак валюты, которые можно использовать в отчётах. Так же использование валют, как сущности позволит без вмешательства разработчиков базы данных добавлять новые валюты при их возможном изменении в банкоматах.
* Банки – это сущность, которая содержит всю основную информацию о банках из базы данных, выделение данной сущности позволяет использовать данные о банках более удобно. Например, найти банк в базе данных по названию. Содержит в себе идентификатор банка, название банка и код адреса.
* Банкоматы – фактически является промежуточной сущностью, которая позволяет анализировать данные более конкретно, например выбрав только тот банкомат, который требует анализа операций. Хранит в себе только идентификаторы банкомата, адреса и банка.
* Адреса – сущность, которую стоит выделить только для того, чтобы не повторять данные о местоположении, так как по условию задачи банкомат и банк могут находится в одном здании. Данная сущность содержит в себе следующие атрибуты, которые являются элементом полного адреса: страна, город, округ, округ, улица, дом, корпус. И последним атрибутом становится код адреса (уникальный идентификатор адреса), который позволяет указывать адрес, не прибегая к избыточности данных в сущностях банки и банкоматы.
* Клиенты – сущность, выделяемая для хранения минимальных данных для отчётов и запросов в базе данных. В данную сущность входят следующие атрибуты: код карты (уникальный код карты клиента), имя, фамилия, отчество, код банка (идентификатор банка из сущности банки) в котором была совершена регистрация карты клиента, серия паспорта и его номер соответственно. Серия и номер паспорта позволяет определить уникальный идентификатор клиента в базе данных. Имя, фамилия, отчество нужны для формирования отчётов.
* Типы операций – это сущность создана для возможности добавления в базу данных новых операций для вывода их на отчётах. Содержит только код типа (идентификатор типа операции) и название типа.

Каждая из сущностей должна иметь свой уникальный первичный ключ, поэтому все атрибуты, которые имеют наименование, начинающееся с “код”, будут считаться первичными ключами.

Результатом инфологического проектирования будет ER модель. Чтобы спроектировать данную модель нужно выписать сущности и атрибуты в отдельные элементы, схожие с таблицами. В качестве заголовка данных таблиц нужно указать наименование сущности, а содержимым наименования атрибутов. ER модель приведена в приложении 1.

* 1. Логическое проектирование

Логическое проектирование – это процесс проектирования базы данных, который предполагает построение логики базы данных. Логическое проектирование предполагает собой представление логической таблицы как объекта реляционной базы данных, описание типов атрибутов, размера полей, а также установка ключевых полей.

Таблица 1 - Сущность “Адреса”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодАдреса | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| Страна | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| Область | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| Город | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| Округ | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| Улица | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| Дом | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Нет |
| Корпус | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Нет |

Таблица 2 - Сущность “Банки”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодБанка | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| НазваниеБанка | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Нет |
| КодАдреса | Да | Экономия памяти, нет названия, которое занимает больше. | Текстовый | 25 символов | Внешний |

Таблица 3 - Сущность “Банкоматы”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодБанкомата | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| КодАдреса | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Внешний |
| КодБанка | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Внешний |

Таблица 4 - Сущность “Валюты”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодВалюты | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| Наименование валюты | Да | Экономия памяти, нет наименования, которое занимает больше. | Текстовый | 25 символов | Нет |
| ЗнакВалюты | Да | Нет названий знаков, которые могут быть больше | Текстовый | 25 символов | Нет |

Таблица 5 - Сущность “Операции”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодОперации | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| КодКарты | Да | Экономия памяти, нет наименования, которое занимает больше. | Текстовый | 25 символов | Внешний |
| КодБанкоматы | Да | Нет названий знаков, которые могут быть больше | Текстовый | 25 символов | Внешний |
| Дата | Да | Ограничивается СУБД | Дата и время | Неизвестно | Нет |
| Комиссия | Да | Вариант “нет” занимает не более 3-х символов. | Текстовый | 3 символа, выбор да или нет | Нет |
| Сумма выдачи | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Нет |
| КодВалюты | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Внешний |
| КодТипа | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Внешний |

Таблица 4 - Сущность “Типы операций”

| Атрибут | Ввод | Причина ограничения | Тип данных | Длина | Ключ |
| --- | --- | --- | --- | --- | --- |
| КодТипа | Да | Ограничивается СУБД | Числовой | 2 147 483 647 | Да |
| НазваниеОперации | Да | Экономия памяти, нет наименования, которое занимает больше. | Текстовый | 25 символов | Нет |

* 1. Физическое проектирование

Разработка базы данных будет проводиться в СУБД MS Access. Данная СУБД подходит для этой задачи и при этом разработка в ней происходит легче, чем в альтернативных СУБД.

Перенос базы данных из логической в реляционную, которой является MS Access происходит в определённой последовательности. Сначала создаются сущности в MS Access они имеют название “таблицы”, затем в данных таблицах создаются атрибуты, которым указываются типы данных и свойства, так же в MS Access необходимо указать необходимые первичные ключи базе атрибутам, которые этого требуют. Внешние ключи не указываются. При переносе данной логической схемы в MS Access стоит учитывать, что все свойства, которые не затронуты остаются стандартными.

После настройки свойств атрибутов производится связывание таблиц в схеме данных – это интерфейс пользователя, в котором пользователь может проводить связи между таблицами и указывать типы обеспечения целостности данных. Указывая целостность данных для данной базы данных, выбирается каскадное удаление связанных полей и каскадное обновление.

Данная база данных имеет лишь небольшие отличия атрибутов от стандартных. Каждое поле, которое указано как числовое в логическом проектировании имеет размер поля в MS Access “Длинное целое”. Каждое текстовое поле имеет размер поля, указанный в логическом проектировании. Все поля, которые обязательны имеют свойство обязательное поле, установленное, как “да” в другом случае “нет”.

После физического проектирования формируются дополнительные элементы, такие как запросы, отчёты и формы.

* 1. Формирование запросов и отчётов

Так как данная база данных формируется больше для анализа, чем для хранения данных, то без запросов нельзя обойтись. На основе запросов должны формироваться отчёты.

* + 1. Запросы

Из постановки задачи собираются следующие запросы

* Операции банкомата за промежуток. Схема работы запроса указана на рисунке 1.

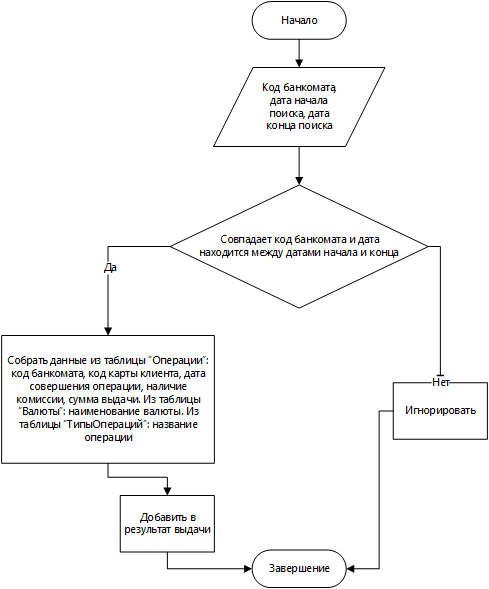


Рисунок 1 - Операции банкомата за промежуток

* Операции банкоматов за промежуток. Схема работы указана на рисунке 2.

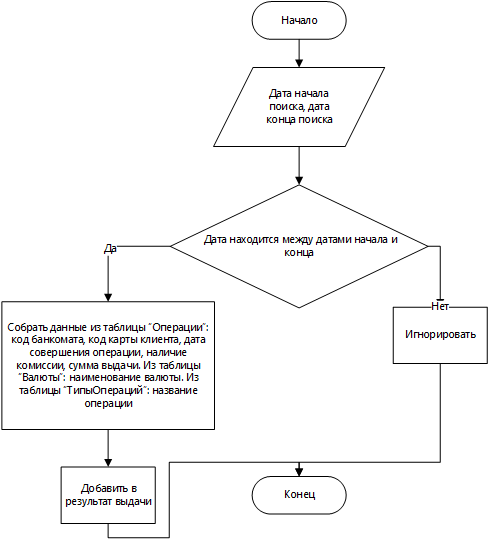


Рисунок 2 - Операции банкоматов за промежуток

* Операции банкомата за промежуток в валюте. Схема работы указана на рисунке 3.

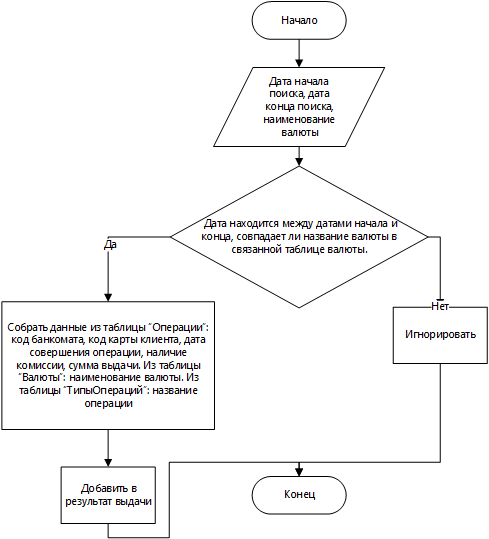


Рисунок 3 - Операции банкоматов за промежуток в валюте

* Операции банкомата за промежуток где сумма выдачи. Схема работы указана на рисунке 4.

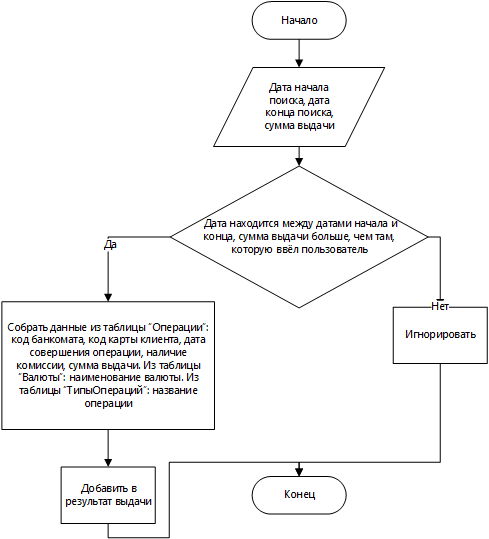


Рисунок 4 - Операции банкоматов за промежуток где сумма выдачи

* Операции банкомата, где промежуток определённого типа. Схема работы указана на рисунке 5.

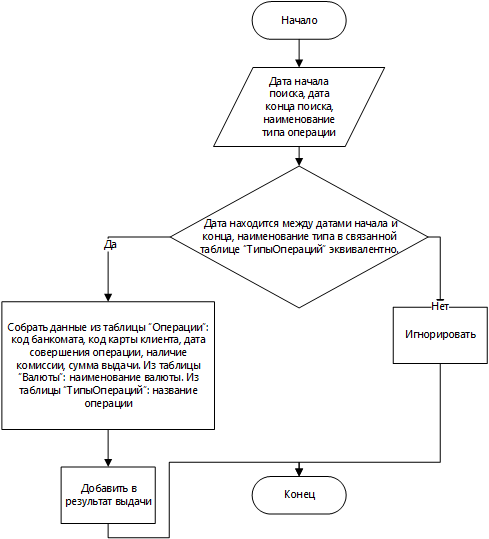


Рисунок 5 - Операции банкоматов за промежуток определённого типа

* Операции за промежуток, сумма. Схема работы указана на рисунке 6.

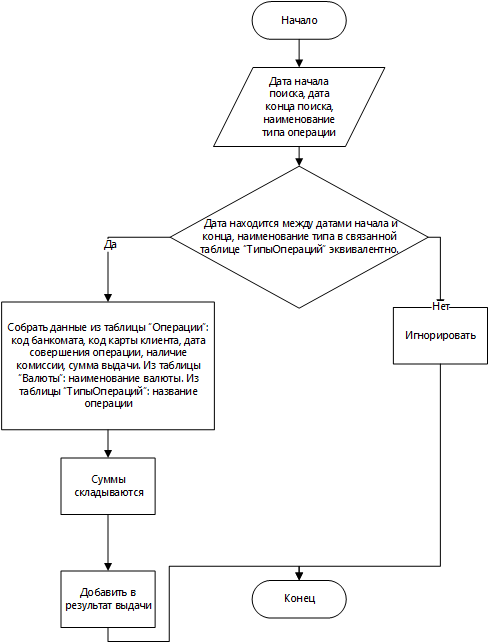


Рисунок 6 - Операции банкоматов за промежуток определённого типа

* Операции клиента за промежуток. Схема работы указана на рисунке 7.

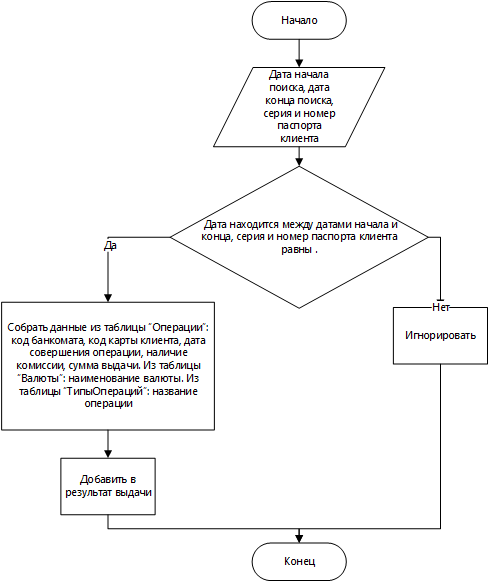


Рисунок 7 - Операции клиента за промежуток

* + 1. Отчёты
  1. Разработка интерфейса. Создание экранных форм.

В данной базе данных все данные вводятся сотрудниками, поэтому база данных содержит экранные формы для заполнения каждой из таблиц.