содержание

ВВЕДЕНИЕ 4

1 ОПИСАНИЕ МОДЕЛИ АВТОМАТИЗАЦИИ 5

1.1 Организационная модель 5

1.2 Функциональная модель 6

1.3 Информационная модель 8

1.4 Модель бизнес-процесса объекта автоматизации 17

2 РАЗРАБОТКА БАЗЫ ДАННЫХ 18

2.1 Концептуальная модель 18

2.2 Логическая модель 21

2.3 Физическая модель 23

ЗАКЛЮЧЕНИЕ 28

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 29

СПИСОК СОКРАЩЕНИЙ 30

ПРИЛОЖЕНИЕ А НАБОР ТЕСТОВЫХ ЗАДАНИЙ ДЛЯ ПРОВЕРКИ

ПРИЛОЖЕНИЕ Б РЕЗУЛЬТАТЫ СОЗДАНИЯ, ЗАГРУЗКИ И ПРОВЕРКИ БД

# ВВЕДЕНИЕ

**Цель**: спроектировать базу данных для обеспечения хранения данных при решении задачи инвентаризации для информационной системы «Аптека». Проектирование баз данных — процесс создания схемы базы данных и определения необходимых ограничений целостности. Для проектирования базы данных нужно сначала построить модель проектируемой системы.

**Задача**: разработать модели организации для объекта автоматизации.

Модель базы данных – это тип модели данных, которая определяет логическую структуру базы данных и в корне определяет, каким образом данные могут храниться, организовываться и обрабатываться. Самым популярным примером модели базы данных является реляционная модель, которая использует табличный формат.

Модель деятельности компании – наиболее важная часть организационной модели, которая показывает процесс создания ценности в компании для всех заинтересованных сторон. В модели деятельности компании обычно выделяют следующие уровни: верхний (системный), средний (функциональный) и нижний (операционный). Таким образом, чтобы создать автоматизированную систему и базу данных, необходимо разработать модель, корректно описывающую данный объект автоматизации.

Объект автоматизации (ОА) – комплекс задач по поставкам, приему, хранению и реализации фармацевтической продукции.

**Способы**: разработать модели организации (организационную, функциональную, информационную и бизнес-процессов) можно с помощью методологии ARIS.

ARIS (Architecture of Integrated Information Systems) — методология для моделирования бизнес-процессов организаций. Любая организация в методологии ARIS рассматривается с пяти точек зрения: организационной, функциональной, обрабатываемых данных, структуры бизнес-процессов, продуктов и услуг. Каждая точка зрения представляет собой модель.

**Результат**: спроектированная база данных для выбранного объекта автоматизации с возможностью формирования, ведения и документирования справочников, а также отчетных документов.

# ОПИСАНИЕ МОДЕЛИ АВТОМАТИЗАЦИИ

**1.1 Организационная модель**

Организационная структура предприятия – состав, взаимодействие, соподчиненность, а также распределение работы по подразделениям и управленческим органам, между которыми формируются определенные отношения, связанные с реализацией властных полномочий, потоков распоряжений и информации.

Структура предприятия устанавливается исходя из объема и содержания задач, решаемых предприятием, направленности и интенсивности сложившихся на предприятии информационных и документационных потоков, а также принимая во внимание организационные и материальные возможности.

Организационная модель ОА «Инвертаризация» для ИС «Аптека» представлена органограммой «Аптека» (рисунок 1.1) и таблицей «Каталог организационных единиц» (таблица 1.1).

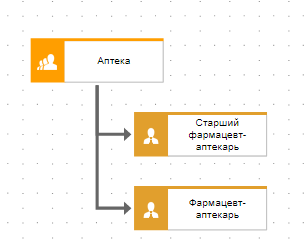


Рисунок 1.1 – Органограмма «Аптека»

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование организационной единицы** | **Расшифровка** |
| 1 | Аптека | Отделение |
| 2 | Старший аптекарь-фармацевт | Руководитель отделения |
| 3 | Аптекарь-фармацевт | Сотрудник, осуществляющий продажу товара, подчиняется руководителю |

Таблица 1.1 – Каталог организационных единиц

**1.2 Функциональная модель**

Функциональная модель объекта автоматизации – описание его на языке выполняемых функций и их отношений.

Функциональная структура – структура, элементами которой являются функции, реализуемые подразделениями предприятия, а отношениями — связи, обеспечивающие передачу между элементами предметов труда.

Функция – это предметно-ориентированное задание или действие, в результате которой выполняется одна или несколько целей, стоящих перед компанией. Функции предприятия распределяются по компонентам организационной структуры и представляют собой иерархическое дерево, строящееся от общего к частному. На самом верхнем уровне описываются самые сложные функции, которые потом детализируются через свои функциональные составляющие.

В функциональную модель ОА «Инвертаризация» для ИС «Аптека» входят следующие функции:

1. Инвентаризация фармацевтической продукции;

2. Формирование и документирование результатов инвентаризации.

Функциональная модель ОА «Инвертаризация» для ИС «Аптека» представлена на рисунке 1.2 «Функциональное дерево ОА» и таблицей «Каталог функций» (таблица 1.2).

Diagram

Description automatically generated

Рисунок 1.2 – Функциональное дерево ОА

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Наименование функции** | **Организационный элемент** | **Документы** |
| 1 | Пересчет и фиксирование товара | Комиссия из 3 работников аптеки | ОД\_ИнвОп |
| 2 | Оформление результатов инвентаризации | Старший аптекарь-фармацевт, аптекарь-фармацевт | ОД\_КартСклУч, ОД\_ЖурнУчПостТов |

Таблица 1.2 – Каталог функций

**1.3 Информационная модель**

Информационная модель — модель объекта, представленная в виде информации, описывающей существенные для данного рассмотрения параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая путём подачи на модель информации об изменениях входных величин моделировать возможные состояния объекта.

Информационная модель ОА «Инвертаризация» для ИС «Аптека» состоит из следующих документов:

1. **Справочные документы:**

Справочные документы представлены в «Каталоге справочных документов» (таблица 1.3).

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Идентификатор документа** | **Наименование документа** |
| 1 | СП\_Сотр | Сотрудники |
| 2 | СП\_ДолжнСотр | Должность сотрудника |
| 3 | СП\_Номенкл | Номенклатура |
| 4 | СП\_Производ | Производители |
| 5 | СП\_ЕдХран | Единицы хранения |
| 6 | СП\_Подразд | Подразделения |

Таблица 1.3 – Каталог справочных документов

1.1. Справочник «Сотрудники» – содержит перечень сотрудников, работающих в аптеке. Документ представлен в виде словаря данных (таблица 1.4). Эталон документа на рисунке 1.3.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код сотрудника | Числовой |
| 2 | НаимСотр | ФИО сотрудника | Текстовый |
| 3 | Должность | Должность сотрудника | Текстовый |
| 4 | Подразделение | Подразделения сотрудника | Текстовый | |

Таблица 1.4 – Словарь данных справочника «Сотрудники»

Shape, rectangle

Description automatically generated

Рисунок 1.3 – Эталон справочника «Сотрудники»

1.2. Справочник «Должность сотрудников» – содержит перечень должностей, которые могут занимать сотрудники. Документ представлен в виде словаря данных (таблица 1.5). Эталон документа на рисунке 1.4.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код сотрудника | Числовой |
| 2 | Должность | Должность сотрудника | Текстовый |

Таблица 1.5 – Словарь данных справочника «Должность сотрудников»

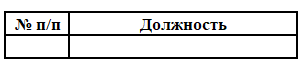


Рисунок 1.4 – Эталон справочника «Должность сотрудников»

1.3. Справочник «Номенклатура» – содержит код и наименование фармацевтической продукции, продающейся в аптеке. Документ представлен в виде словаря данных (таблица 1.6). Эталон документа на рисунке 1.5.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код фармацевтической продукции | Числовой |
| 2 | НаимФармПрод | Наименование фармацевтической продукции | Текстовый |
| 3 | Производитель | Предприятие-производитель | Текстовый |
| 6 | Описание | Дополнительное описание | Текстовый |

Таблица 1.6 – Словарь данных справочника «Номенклатура»



Рисунок 1.5 – Эталон справочника «Номенклатура»

1.4. Справочник «Производители» – содержит перечень производителей, которые изготовляют товар. Документ представлен в виде словаря данных (таблица 1.7). Эталон документа на рисунке 1.6.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код производителя | Числовой |
| 2 | НаимПроизвод | Наименование производителя | Текстовый |

Таблица 1.7 – Словарь данных справочника «Производители»

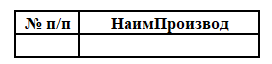


Рисунок 1.6 – Эталон справочника «Производители»

1.5. Справочник «Единицы хранения» – содержит перечень единиц, которые используются при описании товара. Документ представлен в виде словаря данных (таблица 1.8). Эталон документа на рисунке 1.7.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код единицы хранения | Числовой |
| 2 | НаимЕдХран | Наименование единицы хранения | Текстовый |

Таблица 1.8 – Словарь данных справочника «Единицы хранения»



Рисунок 1.7 – Эталон справочника «Единицы хранения»

1.6. Справочник «Единицы хранения» – содержит перечень единиц, которые используются при описании товара. Документ представлен в виде словаря данных (таблица 1.9). Эталон документа на рисунке 1.8.

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** |
| 1 | Код | Код подразделения | Числовой |
| 2 | Подразделение | Номер подраздления | Числовой |

Таблица 1.9 – Словарь данных справочника «Подразделения»

Shape, rectangle

Description automatically generated

Рисунок 1.8 – Эталон справочника «Подразделения»

1. **Оперативные документы:**

Оперативныедокументы представлены в «Каталоге оперативных документов» (таблица 1.10).

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Идентификатор документа** | **Наименование документа** |
| 1 | ОД\_КартСклУч | Карточка складского учета |
| 2 | ОД\_ИнвОп | Инвентаризационная опись |

Таблица 1.10 – Каталог оперативных документов

2.1. Оперативный документ «Карточка складского учета» – применяется для учета движения материалов на складе по каждому сорту, виду и размеру, номенклатурному номеру материалов и заполняется работником склада, являющимся материально ответственным лицом. Документ представлен в виде словаря данных (таблица 1.11), макета (рисунок 1.9) и схемы информационных связей между документами (рисунок 1.10).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** | **Источник данных** |
| 1 | НаимОрг | Наименование организации | Текстовый | СП\_Контраг |
| 2 | Дата | Дата | Дата | - |
| 3 | ДокОсн | Документ-основание | Текстовый | ОД\_ТТН, ОД\_ИнвОп |
| 4 | Производ | Производитель | Текстовый | СП\_Производ |
| 5 | Постав | Поставщик | Текстовый | СП\_Контраг |
| 6 | НаимТовара | Наименование товара | Текстовый | СП\_Номенкл |
| 7 | ЕдХран | Единица хранения | Текстовый | СП\_ЕдХран |
| 8 | НачОст | Начальный остаток | Числовой | СП\_Номенкл |
| 9 | Приход | Приход | Числовой | ОД\_ЖурнУчПрРасх |
| 10 | Расход | Расход | Числовой | ОД\_ЖурнУчПрРасх |
| 11 | КонечОст | Конечный остаток | Числовой | - |
| 12 | Должность | Должность МОЛ | Текстовый | СП\_ДолжнСотр |
| 13 | НаимСотр | ФИО МОЛ | Текстовый | СП\_Сотр |

Таблица 1.11 – Словарь данных документа «Карточка складского учета»

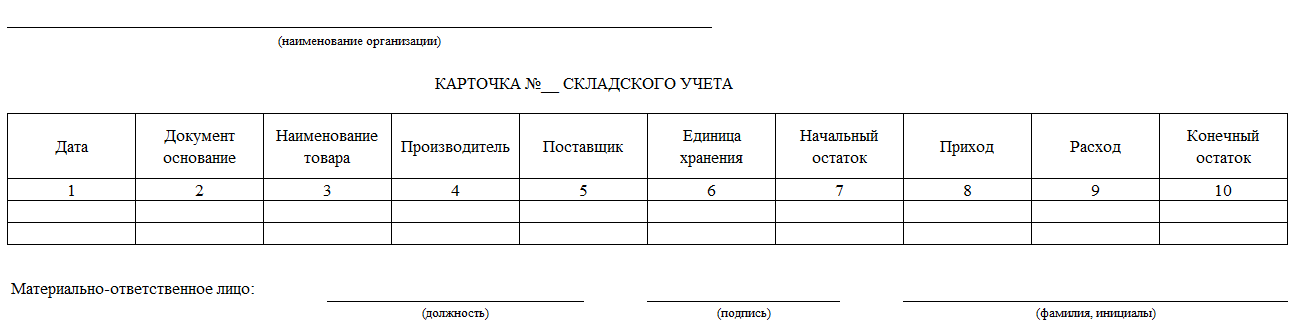


Рисунок 1.9 – Макет документа «Карточка складского учета»

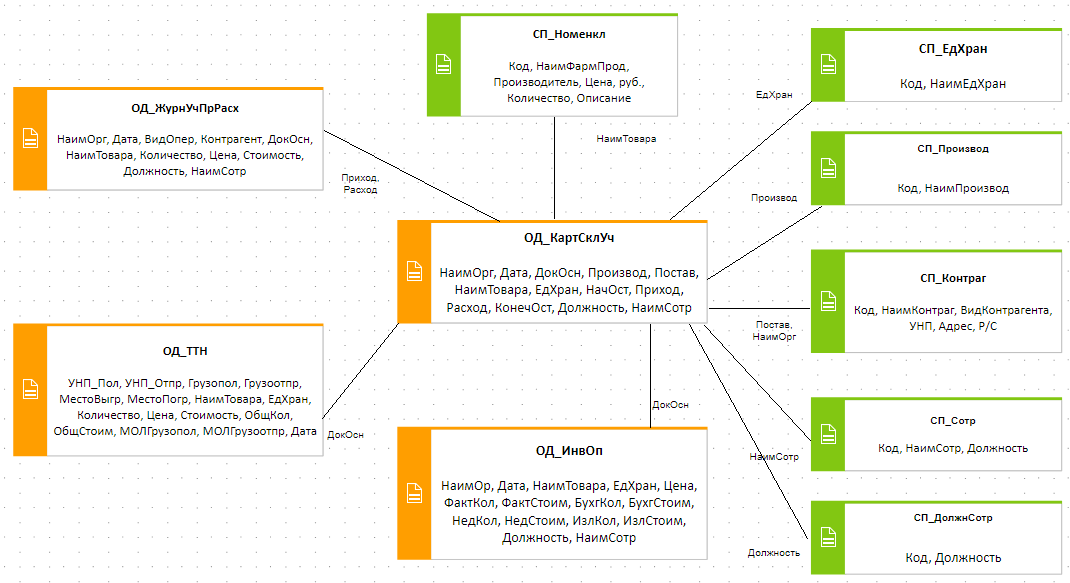


Рисунок 1.10 – Схема информационных связей документа «Карточка складского учета»

2.2. Оперативный документ «Инвентаризационная опись» – документ, в котором отражаются результаты инвентаризации. Документ представлен в виде словаря данных (таблица 1.12), макета (рисунок 1.11) и схемы информационных связей между документами (рисунок 1.12).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** | **Источник данных** |
| 1 | НаимОрг | Наименование организации | Текстовый | СП\_Контраг |
| 2 | Дата | Дата проведения | Числовой | - |
| 3 | НаимТовара | Наименование товара | Текстовый | СП\_Номенкл |
| 4 | ЕдХран | Единица хранения | Текстовый | СП\_ЕдХран |
| 5 | Цена | Цена товара | Числовой | СП\_Номенкл |
| 6 | ФактКол | Фактическое количество товара | Числовой | - |
| 7 | ФактСтоим | Фактическая стоимость товара | Числовой | - |
| 8 | БухгКол | Количество по бух. учету | Числовой | СП\_Номенкл |
| 9 | БухгСтоим | Стоимость по бух. учету | Числовой | СП\_Номенкл |
| 10 | НедКол | Количество товара в недостаче | Числовой | - |
| 11 | НедСтоим | Стоимость товара в недостаче | Числовой | - |
| 12 | ИзлКол | Количество товара в излишке | Числовой | - |
| 13 | ИзлСтоим | Стоимость товара в излишке | Числовой | - |
| 14 | Должность | Должность МОЛ | Текстовый | СП\_ДолжнСотр |
| 15 | НаимСотр | ФИО МОЛ | Текстовый | СП\_Сотр |

Таблица 1.10 – Словарь данных документа «Инвентаризационная опись»

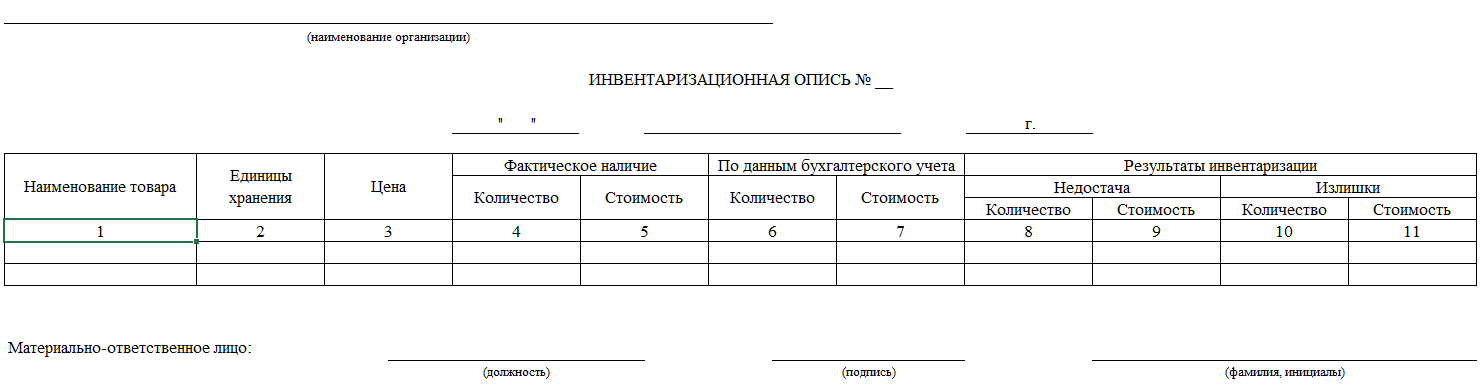


Рисунок 1.11 – Макет документа «Инвентаризационная опись»

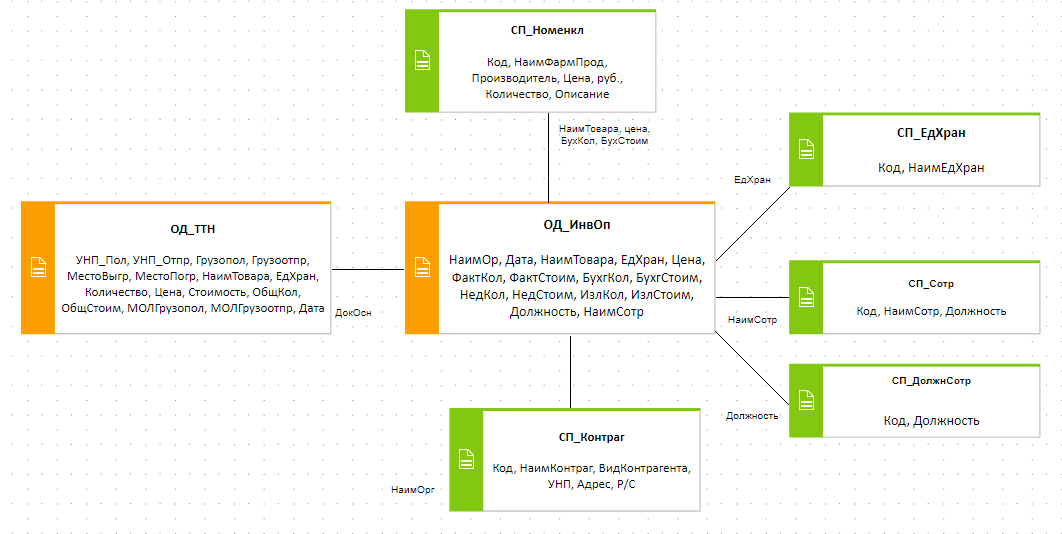


Рисунок 1.12 – Схема информационных связей документа «Инвентаризационная опись»

1. **Отчетные документы:**

Отчетный документ «Оборотно-сальдовая ведомость ТМЦ на дату» – регистр бухгалтерского учета, предназначенный для контроля операций и сальдо по счетам бухгалтерского учета и составления бухгалтерской отчетности. Документ представлен в виде словаря данных (таблица 1.10), макета (рисунок 1.13) и схемы информационных связей между документами (рисунок 1.14).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Идентификатор** | **Наименование** | **Тип данных** | **Источник данных** |
| 1 | НаимОрг | Наименование организаци | Текстовый | СП\_Контраг |
| 2 | ДатаС | Дата начала периода | Дата | - |
| 3 | ДатаПо | Дата конца периода | Дата | - |
| 3 | Дата | Дата изменения товара | Дата | - |
| 5 | НаимТовара | Наименование товара | Текстовый | СП\_Номенкл |
| 6 | Произв | Производитель товара | Текстовый | СП\_Производит |
| 7 | ЕдХран | Единица хранения | Текстовый | СП\_ЕдХран |
| 8 | НачОст | Начальный остаток | Числовой | СП\_Номенкл |
| 9 | Приход | Приход | Числовой | ОД\_ЖурнУчПрРасх |
| 10 | Расход | Расход | Числовой | ОД\_ЖурнУчПрРасх |
| 11 | КонечОст | Конечный остаток | Числовой | - |
| 12 | Должность | Должность МОЛ | Текстовый | СП\_ДолжнСотр |
| 13 | НаимСотр | ФИО МОЛ | Текстовый | СП\_Сотр |

Таблица 1.10 – Словарь данных отчета «Оборотно-сальдовая ведомость»

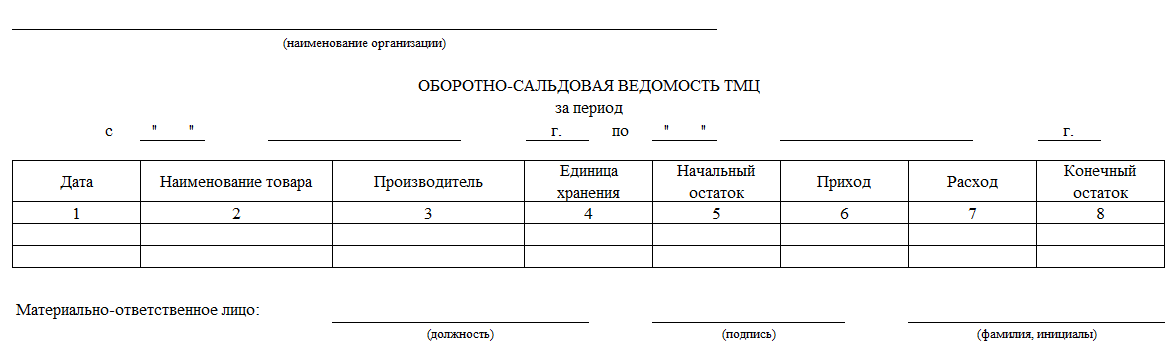


Рисунок 1.13 – Макет отчета «Оборотно-сальдовая ведомость»

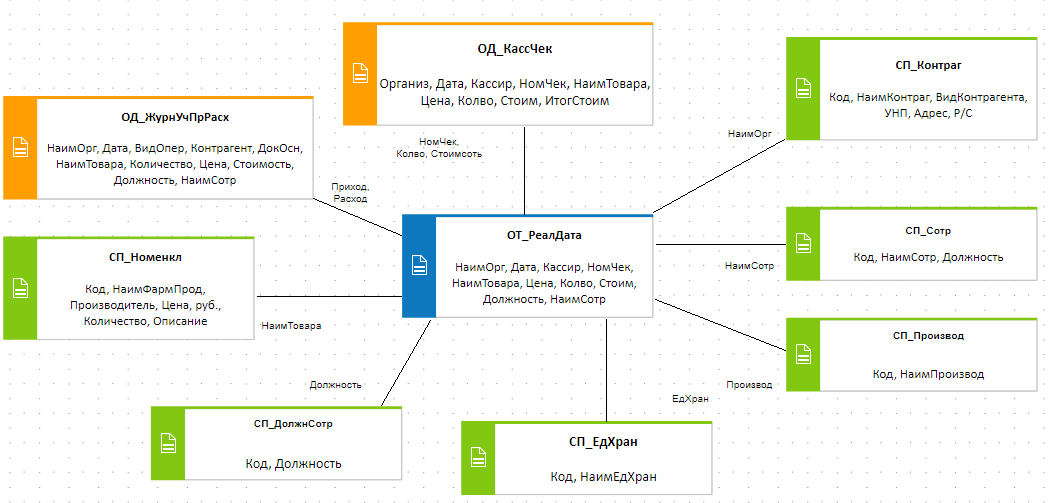


Рисунок 1.14 – Схема информационных связей отчета «Оборотно-сальдовая ведомость»

**1.4 Модель бизнес-процесса объекта автоматизации**

Процесс – любая деятельность, в которой используются ресурсы для преобразования входов в выходы. Зачастую представляет из себя совокупность взаимосвязанных и совершенных работ, в которых результаты одной работы являются началом другой работы, образуя цепочку внутренних поставщиков и потребителей.

Бизнес-процесс – устойчивая и целенаправленная совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входной сигнал в выходной, представляющий ценность для потребителя.

eEPC – нотация для проектирования бизнес-процессов. Данная нотация ARIS представляет бизнес-процесс как цепочку событий и действий (функций). Каждое действие инициализируется и завершается событием.

Модель бизнес-процесса ОА «Инвертаризация» для ИС «Аптека» представлена на рисунке 1.33 с использованием нотации Business process методологии ARIS.

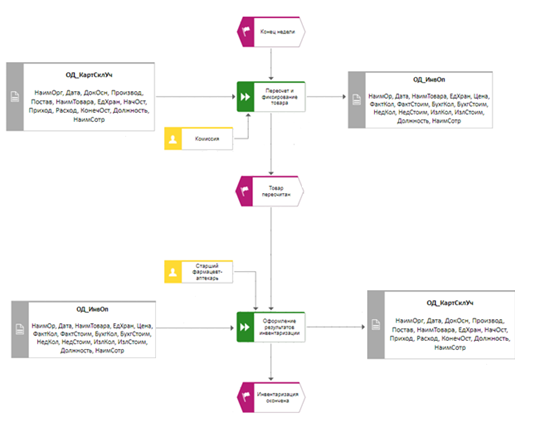


Рисунок 1.20 – Модель бизнес-процесс «Инвентаризация товара»

# 2 РАЗРАБОТКА БАЗЫ ДАННЫХ

**2.1 Концептуальная модель**

Предметная область – совокупность объектов, свойства которых и отношения между которыми рассматриваются в рамках некоторого исследования.

Модель предметной области – некоторая система, адекватно имитирующая структуру и функционирование исследуемой предметной области.

Концептуальная модель – это структура моделируемой предметной области, свойств её элементов и причинно-следственных связей, присущих системе и существенных для достижения цели моделирования. В рамках этапа концептуального моделирования выделяются основные смысловые единицы (сущности) предметной области, определяются и описываются связи между ними.

Концептуальная модель ориентирована на потенциальных пользователей базы данных, так как представляет предметную область на их уровне понимания. Этот уровень называется системно-независимым или предметно-ориентированным.

Построим локальные концептуальные модели для ОА «Инвертаризация» для ИС «Аптека» (рис. 2.1-2.5).

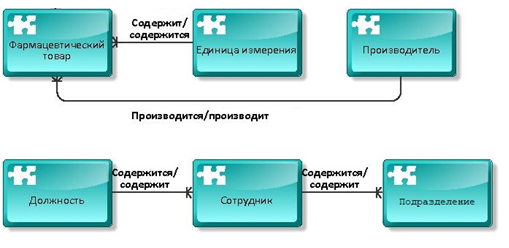


Рисунок 2.1 – Локальная концептуальная модель для справочных документов

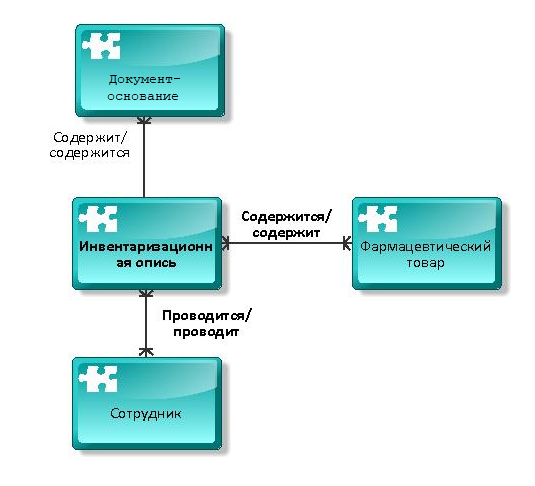


Рисунок 2.2 – Локальная концептуальная модель для оперативного документа «Инвентаризационная опись»

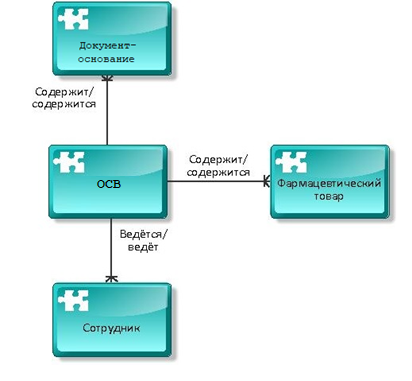


Рисунок 2.3 – Локальная концептуальная модель для оперативного документа «Оборотно-сальдовая ведомость на дату»

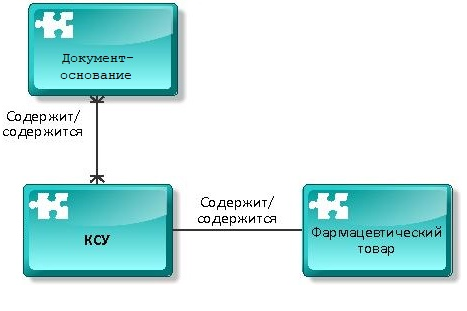


Рисунок 2.4 – Локальная концептуальная модель для оперативного документа «Карточка складского учета»

Концептуальная модель для ОА «Инвертаризация» для ИС «Аптека» представлена на рисунке 2.5 с использованием нотации Data model методологии ARIS.

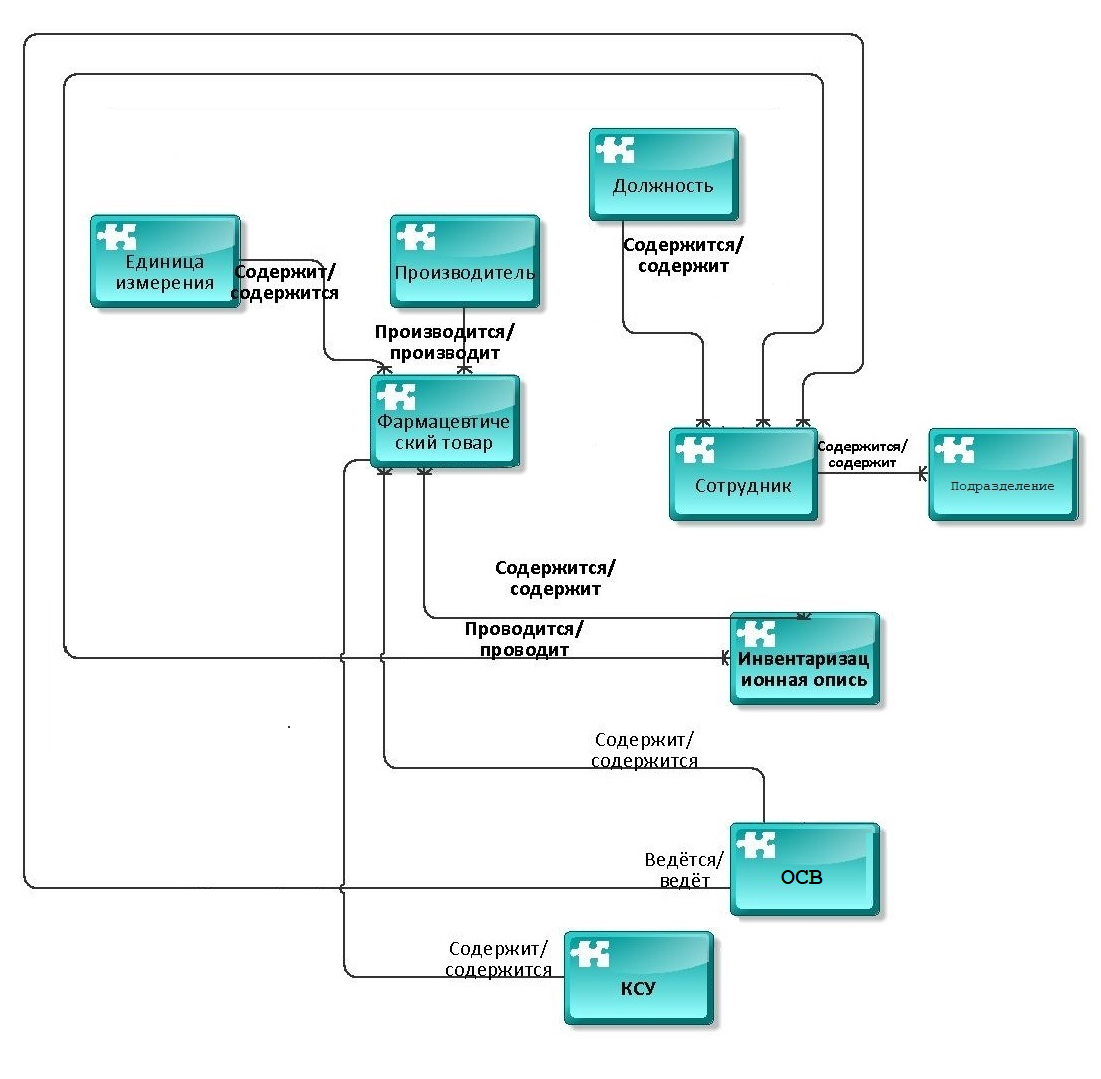


Рисунок 2.7 – Концептуальная модель для ОА «Инвертаризация» для ИС «Аптека»

**2.2. Логическая модель**

Логическая (даталогическая) модель — это схема базы данных, разработанная на основе конкретной модели данных (реляционной модели) и концептуальной модели предметной области. Преобразование концептуальной модели в логическую модель было осуществлено по следующим правилам:

* Приведение всех связей между отношениями к бинарному типу с кардинальностью 1:N.
* Описание доменов.
* Определение первичных и внешних ключей.
* Преобразования сущностей концептуальной модели в отношения логической модели путем описания их через атрибуты и механизмы целостности данных.
* Анализ связей 1:1 на предмет необходимости их присутствия в логической модели.
* Нормализация отношений.

Бинарные связи между сущностями вида «многие-ко-многим» при преобразовании концептуальной модели в логическую удаляются путем введения ассоциативной (промежуточной) сущности, которая буде состоять в связи «один-ко-многим» с каждой из сущностей преобразуемой бинарной связи.

Для представления «Аптека» были введены домены, описание которых представлено в таблице 2.1.

|  |  |  |
| --- | --- | --- |
| **Имя домена** | **Характеристики домена** | **Пример допустимых значений** |
| Код | Положительный целочисленный | 0 |
| Строка | Текстовый | Строка Тест |
| Число | Положительный целочисленный | 1 |
| Дата | Дата | 01.01.2022 |

Таблица 2.1 – Описание доменов

Логическая модель была построена с помощью Online Data Modeler и представлена на рисунке 2.8.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 2.8 – Логическая модель для ОА «Инвертаризация» для ИС «Аптека»

**2.3. Физическая модель**

Физическая модель данных описывает реализацию объектов логической модели на уровне объектов конкретной СУБД. Для создания БД была выбрана сетевая реляционная СУБД MySQL, которая предоставляет всю современную функциональность, располагающуюся на стороне сервера. После завершения этапа логического моделирования был проведен переход к физической модели БД, а также сгенерированы скрипты для создания объектов базы данных «Аптека». Физическая модель представлена на рисунке 2.9.

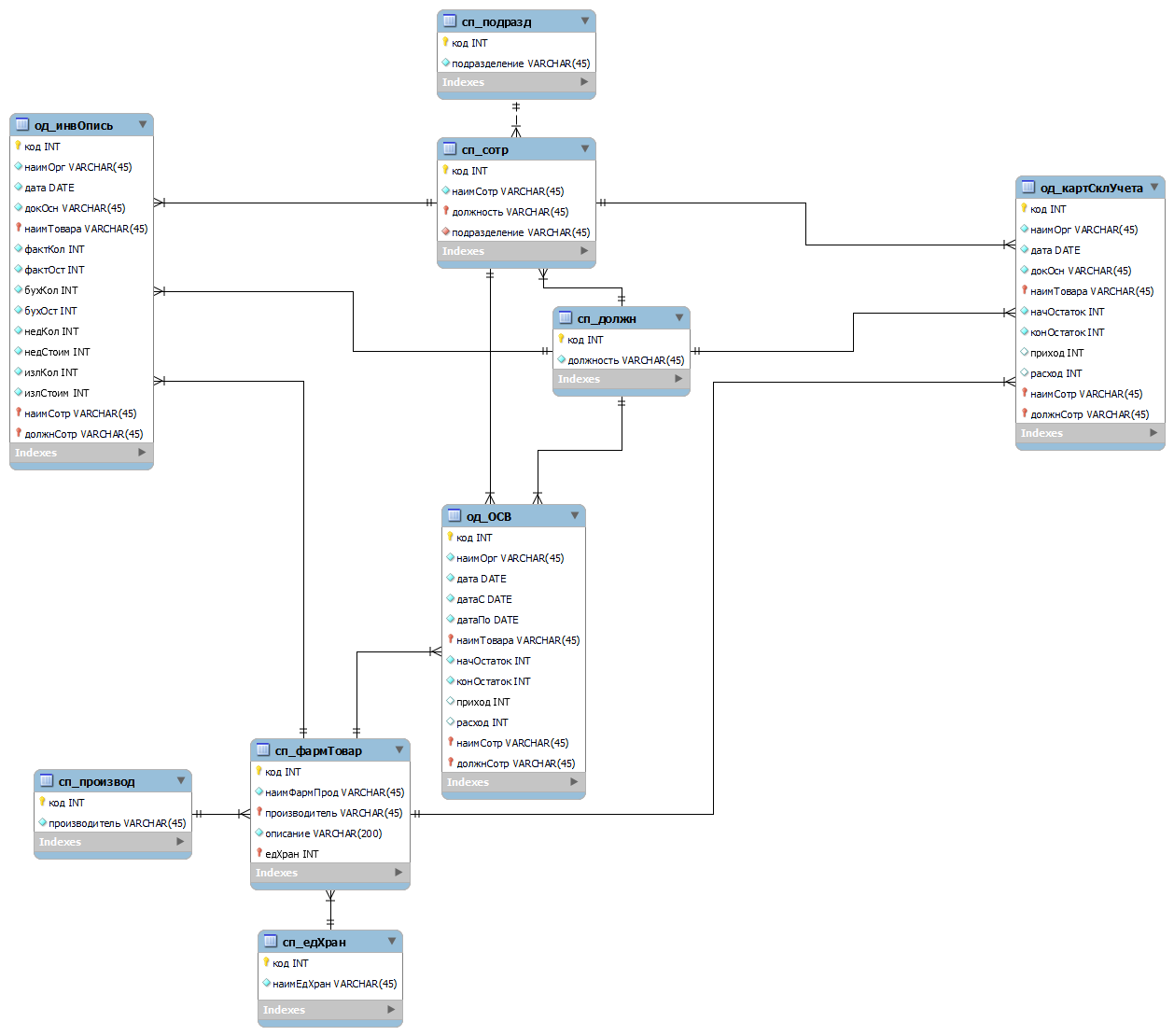


Рисунок 2.9 – Физическая модель для ОА «Инвертаризация» для ИС «Аптека»

Далее приведен скрипт для создания базы данных «Аптека».

-- -----------------------------------------------------

-- Schema apteka

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema apteka

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `apteka` DEFAULT CHARACTER SET utf8 COLLATE utf8\_czech\_ci ;

USE `apteka` ;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_должн`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_должн` (

`код` INT NOT NULL AUTO\_INCREMENT,

`должность` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`),

UNIQUE INDEX `должность\_UNIQUE` (`должность` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_подразд`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_подразд` (

`код` INT NOT NULL,

`подразделение` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`),

UNIQUE INDEX `подразделение\_UNIQUE` (`подразделение` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_сотр`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_сотр` (

`код` INT NOT NULL,

`наимСотр` VARCHAR(45) NOT NULL,

`должность` VARCHAR(45) NOT NULL,

`подразделение` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`, `должность`),

UNIQUE INDEX `наимСотр\_UNIQUE` (`наимСотр` ASC) VISIBLE,

INDEX `подразделение\_idx` (`подразделение` ASC) VISIBLE,

INDEX `должность\_idx` (`должность` ASC) VISIBLE,

CONSTRAINT `подразделение`

FOREIGN KEY (`подразделение`)

REFERENCES `apteka`.`сп\_подразд` (`подразделение`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `должность`

FOREIGN KEY (`должность`)

REFERENCES `apteka`.`сп\_должн` (`должность`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_едХран`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_едХран` (

`код` INT NOT NULL,

`наимЕдХран` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_производ`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_производ` (

`код` INT NOT NULL,

`производитель` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`),

UNIQUE INDEX `производитель\_UNIQUE` (`производитель` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`сп\_фармТовар`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`сп\_фармТовар` (

`код` INT NOT NULL,

`наимФармПрод` VARCHAR(45) NOT NULL,

`производитель` VARCHAR(45) NOT NULL,

`описание` VARCHAR(200) NOT NULL,

`едХран` INT NOT NULL,

PRIMARY KEY (`код`, `едХран`, `производитель`),

INDEX `производитель\_idx` (`производитель` ASC) VISIBLE,

INDEX `fk\_сп\_фармТовар\_сп\_едХран1\_idx` (`едХран` ASC) VISIBLE,

UNIQUE INDEX `наимФармПрод\_UNIQUE` (`наимФармПрод` ASC) VISIBLE,

CONSTRAINT `производитель`

FOREIGN KEY (`производитель`)

REFERENCES `apteka`.`сп\_производ` (`производитель`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_сп\_фармТовар\_сп\_едХран1`

FOREIGN KEY (`едХран`)

REFERENCES `apteka`.`сп\_едХран` (`код`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`од\_картСклУчета`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`од\_картСклУчета` (

`код` INT NOT NULL,

`наимОрг` VARCHAR(45) NOT NULL,

`дата` DATE NOT NULL,

`докОсн` VARCHAR(45) NOT NULL,

`наимТовара` VARCHAR(45) NOT NULL,

`начОстаток` INT NOT NULL,

`конОстаток` INT NOT NULL,

`приход` INT NULL,

`расход` INT NULL,

`наимСотр` VARCHAR(45) NOT NULL,

`должнСотр` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`, `наимТовара`, `должнСотр`, `наимСотр`),

UNIQUE INDEX `наимТовара\_UNIQUE` (`наимТовара` ASC) VISIBLE,

UNIQUE INDEX `наимСотр\_UNIQUE` (`наимСотр` ASC) VISIBLE,

INDEX `наимДолж\_idx` (`должнСотр` ASC) VISIBLE,

CONSTRAINT `наимСотр`

FOREIGN KEY (`наимСотр`)

REFERENCES `apteka`.`сп\_сотр` (`наимСотр`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `должнСотр`

FOREIGN KEY (`должнСотр`)

REFERENCES `apteka`.`сп\_должн` (`должность`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `товар`

FOREIGN KEY (`наимТовара`)

REFERENCES `apteka`.`сп\_фармТовар` (`наимФармПрод`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`од\_инвОпись`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`од\_инвОпись` (

`код` INT NOT NULL,

`наимОрг` VARCHAR(45) NOT NULL,

`дата` DATE NOT NULL,

`докОсн` VARCHAR(45) NOT NULL,

`наимТовара` VARCHAR(45) NOT NULL,

`фактКол` INT NOT NULL,

`фактОст` INT NOT NULL,

`бухКол` INT NOT NULL,

`бухОст` INT NOT NULL,

`недКол` INT NOT NULL,

`недСтоим` INT NOT NULL,

`излКол` INT NOT NULL,

`излСтоим` INT NOT NULL,

`наимСотр` VARCHAR(45) NOT NULL,

`должнСотр` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`, `наимСотр`, `должнСотр`, `наимТовара`),

UNIQUE INDEX `наимТовара\_UNIQUE` (`наимТовара` ASC) VISIBLE,

UNIQUE INDEX `наимСотр\_UNIQUE` (`наимСотр` ASC) VISIBLE,

INDEX `наимДолж\_idx` (`должнСотр` ASC) VISIBLE,

CONSTRAINT `наимСотр2`

FOREIGN KEY (`наимСотр`)

REFERENCES `apteka`.`сп\_сотр` (`наимСотр`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `должнСотр2`

FOREIGN KEY (`должнСотр`)

REFERENCES `apteka`.`сп\_должн` (`должность`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `товар2`

FOREIGN KEY (`наимТовара`)

REFERENCES `apteka`.`сп\_фармТовар` (`наимФармПрод`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `apteka`.`од\_ОСВ`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `apteka`.`од\_ОСВ` (

`код` INT NOT NULL,

`наимОрг` VARCHAR(45) NOT NULL,

`дата` DATE NOT NULL,

`датаС` DATE NOT NULL,

`датаПо` DATE NOT NULL,

`наимТовара` VARCHAR(45) NOT NULL,

`начОстаток` INT NOT NULL,

`конОстаток` INT NOT NULL,

`приход` INT NULL,

`расход` INT NULL,

`наимСотр` VARCHAR(45) NOT NULL,

`должнСотр` VARCHAR(45) NOT NULL,

PRIMARY KEY (`код`, `наимСотр`, `должнСотр`, `наимТовара`),

UNIQUE INDEX `наимТовара\_UNIQUE` (`наимТовара` ASC) VISIBLE,

UNIQUE INDEX `наимСотр\_UNIQUE` (`наимСотр` ASC) VISIBLE,

INDEX `наимДолж\_idx` (`должнСотр` ASC) VISIBLE,

CONSTRAINT `наимСотр1`

FOREIGN KEY (`наимСотр`)

REFERENCES `apteka`.`сп\_сотр` (`наимСотр`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `должнСотр1`

FOREIGN KEY (`должнСотр`)

REFERENCES `apteka`.`сп\_должн` (`должность`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `товар1`

FOREIGN KEY (`наимТовара`)

REFERENCES `apteka`.`сп\_фармТовар` (`наимФармПрод`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

# Заключение

В ходе выполнения данного курсового проекта была разработана база данных для обеспечения хранения данных при решении комплекса задач для подсистемы «Инвентаризация товара» информационной системы «Аптека». Цели и задачи были реализованы полностью.

Данный курсовой проект можно было бы усовершенствовать, добавив поддержку триггеров, что обеспечило бы частичную автоматизацию заполнения некоторых полей таблиц, а также обеспечило более действенное выполнение условия целостностности данных.

При выполнении применялась методология моделирования бизнес-процессов ARIS, использовался инструмент моделирования ARIS Express, программа для проектирования и документирования баз данных Online Data Modeler и СУБД MySQL для создания, проверки и загрузки базы данных.

# Список используемых источников

1. ARIS Cloud - [Электронный ресурс]. – Режим доступа: https://ariscloud.com/ - Дата доступа: 08.10.2021
2. Организационная структура - [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Организационная\_структура- Дата доступа: 08.10.2021
3. Информационная модель - [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/ Информационная\_модель - Дата доступа: 08.10.2021
4. Online Data Modeler - [Электронный ресурс]. – Режим доступа: https://lucid.app/documents#/ - Дата доступа: 08.10.2021
5. MySQL Documentation - [Электронный ресурс]. – Режим доступа: https://dev.mysql.com/doc/ - Дата доступа: 08.10.2021

# Список сокращений

БД – база данных.

ОА – объект автоматизации.

ИС – информационная система.

МОЛ – материально-ответственное лицо.

СУБД – система управления базами данных.

ТМЦ – товарно-материальные ценности.

ARIS – Architecture of Integrated Information Systems.

СП – справочный документ.

ОД – оперативный документ.

ОТ – отчетный документ.

СП\_инвОпис – оперативный документ «Инвентаризационная опись».

СП\_картСклУч – оперативный документ «Карточка складского учета».

СП\_ОСВ – отчетный документ «Оборотно-сальдовая ведомость».

СП\_подраздел – справочный документ «Подразделение».

СП\_должнСотр – справочный документ «Должности сотрудников».

СП\_сотр – справочный документ «Сотрудники».

СП\_едХран – справочный документ «Единицы хранения».

СП\_фармТовар – справочный документ «Фармацевтический товар».

СП\_производ – справочный документ «Производитель».