Государственное бюджетное профессиональное образовательное учреждение

города Москвы

«Московский государственный колледж электромеханики и

информационных технологий»

(ГБПОУ МГКЭИТ)

Отчет ВКР

Проектирование базы данных инвентаризация

Выполнил студент группы 3ИП-11-19

Насонов Д.С.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc88816885)

[1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 4](#_Toc88816886)

[1.1 Исследование предметной области 4](#_Toc88816887)

[1.2 Анализ существующих СУБД 4](#_Toc88816888)

[2 ПРАКТИЧЕСКАЯ ЧАСТЬ 5](#_Toc88816889)

[2.1 Формирование требований 5](#_Toc88816890)

[2.2 Проектирование базы данных 5](#_Toc88816891)

[2.3 Разработка базы данных 6](#_Toc88816892)

[3 ОХРАНА ТРУДА И ТЕХНИКА БЕЗОПАСНОСТИ 8](#_Toc88816893)

[ЗАКЛЮЧЕНИЕ 9](#_Toc88816894)

[СПИСОК ЛИТЕРАТУРЫ 10](#_Toc88816895)

[ПРИЛОЖЕНИЕ А. ИСХОДНЫЙ КОД ОБЪЕКТОВ БАЗЫ ДАННЫХ 11](#_Toc88816896)

# ВВЕДЕНИЕ

Целью ВКР является –

Актуальность заключается в …

Объектом исследования является …

Предметом исследования является …

Результатом окончания ВКР будет …

Какие задачи включает в себя работа над ВКР:

(Содержимое предыдущих практик вставляем в соответствующие разделы отчёта. НЕ БЕЗДУМНО, а аккуратно и (со вкусом))

# 1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

## 1.1 Исследование предметной области

(Сюда содержимое практической работы №2)

## 1.2 Анализ существующих СУБД

(Сюда содержимое практической работы №1)

**1.3 Обоснование выбора СУБД**

Эффективность работы нашей системы зависит от нашего выбора СУБД. Лучшим вариантом СУБД для нашей системы будет MySQL. Из-за ее возможностей:

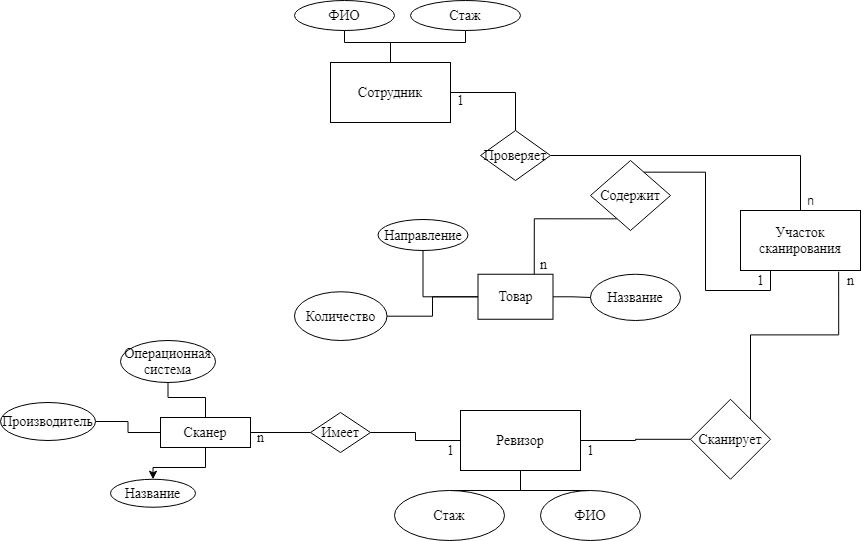
1. Поддерживается эффективная работа неограниченного количества пользователей, одновременно работающих с базой данных.
2. Количество строк в таблицах может достигать 50 млн.
3. Максимально быстрое выполнение команд. Существует мнение, что MySQL – и вовсе самый быстрый сервер из существующих.
4. Простая и эффективная система безопасности.

# 2 ПРАКТИЧЕСКАЯ ЧАСТЬ

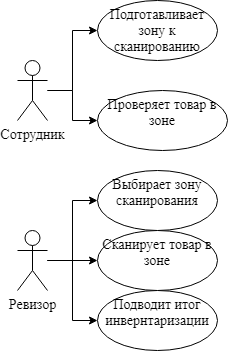
## 2.1 Формирование требований

(Сюда содержимое практической работы №3)

### 2.1.1 Основные сущности базы данных

****

### 2.1.2 Варианты использования



### 2.1.3 Определение API для взаимодействия с базой данных

|  |  |
| --- | --- |
| * **Название метода** | **Описание метода** |
| Добавить сотрудника | Аргументы: ФИО(строка),стаж(временная метка) |
| Изменить данные сотрудника | Аргументы: код(число), ФИО(строка), новое ФИО(строка) стаж(временная метка), новый стаж(временная метка) |
| Изменить данные сотрудника(ФИО) | Аргументы: код(число), ФИО(строка), новое ФИО(строка) |
| Изменить данные сотрудника(стаж) | Аргументы: код(число), ФИО(строка), стаж(временная метка), новый стаж(временная метка) |
| Удалить сотрудника | Аргументы: код(число) |
| Просмотр информации о сотруднике | Аргументы: код(число), ФИО(строка), стаж(временная метка) |
| Добавить ревизора | Аргументы: ФИО(строка),стаж(временная метка) |
| Изменить данные ревизора | Аргументы: код(число), ФИО(строка), новое ФИО(строка) стаж(временная метка), новый стаж(временная метка) |
| Изменить данные ревизора(ФИО) | Аргументы: код(число), ФИО(строка), новое ФИО(строка) |
| Изменить данные ревизора(стаж) | Аргументы: код(число), ФИО(строка), стаж(временная метка), новый стаж(временная метка) |
| Удалить ревизора | Аргументы: код(число) |
| Просмотр информации о ревизоре | Аргументы: код(число), ФИО(строка), стаж(временная метка) |
| Добавить сканер | Аргументы: производитель(строка), название(строка) |
| Изменить сканер | Аргументы: код сканера(число) название(строка) производитель(строка) |
| Удалить сканер | Аргументы: код сканера(число) |
| Добавить товар | Аргументы: код товара(число), название(строка), направление(строка), количество(число) |
| Изменить товар | Аргументы: код товара(число), название(строка), новое название(строка), направление(строка), новое направление(строка), количество(число), новое количество(число) |
| Посмотреть информацию о товаре | Аргументы: код товара(число), название(строка), направление(строка), количество(число) |
| Удалить товар | Аргументы: код товара(число) |
| Изменить товар(название) | Аргументы: код товара(число), название(строка), новое название(строка) |
| Изменить товар(количество) | Аргументы: код товара(число), количество(число), новое количество(число) |
| Изменить товар(направление) | Аргументы: код товара(число), направление(строка), новое направление(строка) |
| Добавить участок сканирования | Аргументы: код участка(число) |
| Изменить участок сканирования | Аргументы: код участка(число), новый код участка(число) |
| Удалить участок сканирования | Аргументы: код участка(число) |

## 2.2 Проектирование базы данных

(Сюда содержимое практической работы №4)

### 2.2.1 Логическая схема базы данных

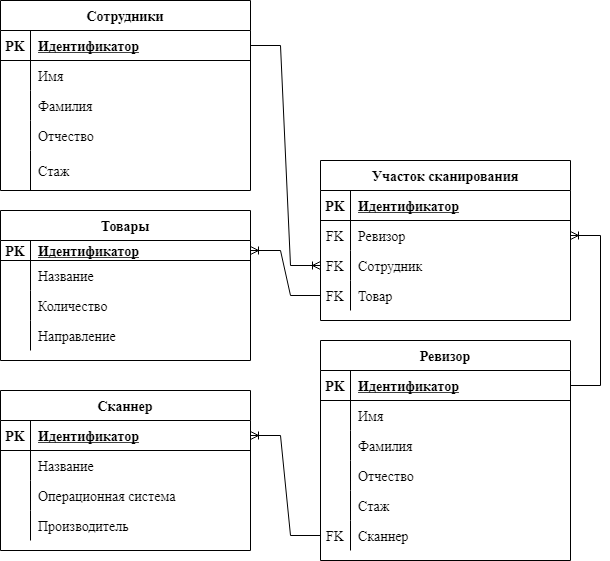


Рисунок 2. – Логическая схема базы данных

### 2.2.2 Физическая схема базы данных

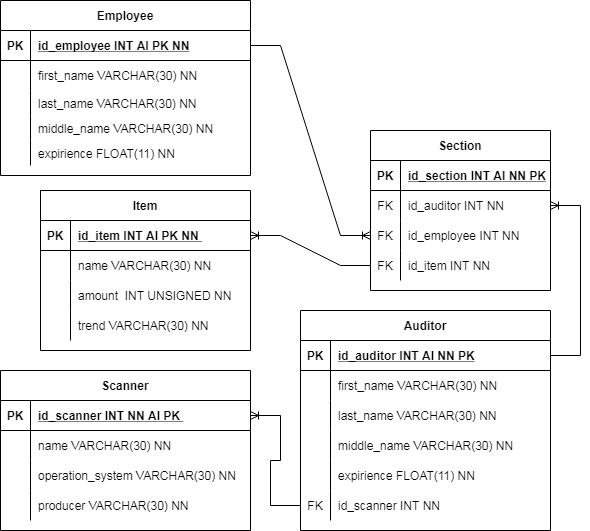


Рисунок .2 – Физическая схема базы данных

### 2.2.3 Определение типовых запросов к объектам базы данных

(Сюда содержимое практической работы №4)

### 2.2.4 Определение процедур и функций API

(Сюда содержимое практической работы №4)

## 2.3 Разработка базы данных

(Приводите пример выполнения запросов к таблицам, запросов к представлениям, вызовы процедур и функций)

### 2.3.1 Разработанные таблицы

(Приводите диаграмму таблиц из MySQL Workbench, а также скриншоты выполнения скриптов, то есть то, что они реально создаются)

### 2.3.2 Разработанные представления

(Приводите диаграмму представлений из MySQL Workbench, а также скриншоты выполнения скриптов, то есть то, что они реально создаются)

### 2.3.3 Разработанные процедуры и функции

(Приводите список созданных процедур и функций из MySQL Workbench, а также скриншоты выполнения скриптов, то есть то, что они реально создаются)

# 3 ОХРАНА ТРУДА И ТЕХНИКА БЕЗОПАСНОСТИ

(Не трогайте, вам потом самим скажут что сюда вставлять)

# ЗАКЛЮЧЕНИЕ

В результате выполнения ВКР была разработана …

В процессе выполнения ВКР были выполнены следующие задачи:

1)

2)

3)

4) …

Достоинства разработанной базы данных:

Недостатки разработанной базы данных:

# СПИСОК ЛИТЕРАТУРЫ

(Сюда вставляете все источники литературы, которые указывали в отчётах практических работ 1-4)

# ПРИЛОЖЕНИЕ А. ИСХОДНЫЙ КОД ОБЪЕКТОВ БАЗЫ ДАННЫХ

Таблица А.1 – Исходный код создания таблиц

|  |
| --- |
| **example\_base.sql** |
| CREATE DATABASE BooksBase CHARACTER SET utf8 COLLATE utf8\_general\_ci;  use `BooksBase`;  CREATE TABLE Books (  idBook int(11) PRIMARY KEY AUTO\_INCREMENT,  caption varchar(255) NOT NULL,  year int(4) NOT NULL,  countPages int(5) NOT NULL,  idGenre int(5) NOT NULL,  INDEX(idGenre)  ) ENGINE = InnoDB;  CREATE TABLE Genres (  idGenre int(11) PRIMARY KEY AUTO\_INCREMENT,  caption varchar(255) NOT NULL UNIQUE,  INDEX(idGenre)  ) ENGINE = InnoDB;  ALTER TABLE `Books`  ADD CONSTRAINT `FK1\_Books`  FOREIGN KEY (`idGenre`) REFERENCES `Genres` (`idGenre`)  ON DELETE CASCADE ON UPDATE CASCADE; |

Таблица А.2 – Исходный код создания хранимых процедур

|  |
| --- |
| **example\_procedures.sql** |
| use `BooksBase`;  DELIMITER //  CREATE PROCEDURE addBook(\_caption varchar(255), \_year int(4), \_countPages int(5), \_idGenre int(11))  BEGIN  INSERT INTO Books  (`caption`, `year`, `countPages`, `idGenre`)  VALUE  (\_caption, \_year, \_countPages, \_idGenre);  END;  CREATE PROCEDURE changeBookCaption(\_idBook int(11), newCaption varchar(255))  BEGIN  UPDATE `Books` SET `caption`=newCaption WHERE `idBook`=\_idBook;  END;  CREATE PROCEDURE changeBookYear(\_idBook int(11), newYear varchar(255))  BEGIN  UPDATE `Books` SET `year`=newYear WHERE `idBook`=\_idBook;  END;  CREATE PROCEDURE changeBookCountPages(\_idBook int(11), \_countPages int(5))  BEGIN  UPDATE `Books` SET `countPages`=\_countPages WHERE `idBook`=\_idBook;  END;  CREATE PROCEDURE removeBook(\_idBook int(11))  BEGIN  DELETE FROM `Books` WHERE `idBook`=\_idBook;  END;  CREATE PROCEDURE addGenre(\_caption varchar(255))  BEGIN  INSERT INTO Genre  (`caption`)  VALUE  (\_caption);  END;  CREATE PROCEDURE removeGenre(\_idGenre int(11))  BEGIN  DELETE FROM `Genre` WHERE `idGenre`=\_idGenre;  END;  //  /\* Для вызова процедур использовать CALL \*/  /\* CALL addBook("Эдда", 1151, 200, 2); \*/  /\* Для удаления процедур использовать DROP PROCEDURE \*/  /\* DROP PROCEDURE addBook;\*/  DELIMITER ; |

Таблица А.3 – Исходный код создания представлений

|  |
| --- |
| **example\_views.sql** |
| use `BooksBase`;  CREATE VIEW viewBooks  AS  SELECT b.caption, b.year, b.countPages, g.caption as 'Genre' FROM `Books` as b  INNER JOIN `Genres` as g ON b.idGenre=g.idGenre  ORDER BY b.caption |