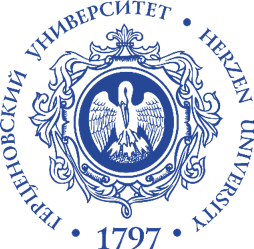
МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»



Направление подготовки   
09.03.01 Информатика и вычислительная техника

Направленность (профиль)

«Технологии разработки программного обеспечения и обработки больших данных»

**ОТЧЁТ**

по реализации проекта для дисциплины «Базы данных»

Разработка системы электронного документооборота в МЧС по Санкт-Петербургу

Преподаватель: к.ф-м.н., доцент кафедры ИТиЭО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Жуков Н. Н.)

Студенты 2 курса:

Иванова О. Д. \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Вяткина А. П. \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Санкт-Петербург  
2025

Оглавление

[Ответственные 3](#_Toc201043876)

[Предметная область 3](#_Toc201043877)

[Ход выполнения нормализации 4](#_Toc201043878)

[Объяснение выбранной СУБД 5](#_Toc201043879)

[ER–диаграмма 5](#_Toc201043880)

[Исходный текст запросов 6](#_Toc201043881)

[По созданию таблиц 6](#_Toc201043882)

[По созданию индексов 11](#_Toc201043883)

# 

# Ответственные

Иванова О. Д. – разработчик проекта. В обязанности Ивановой О. Д. входили анализ предметной области, нормализация данных, создание ER-диаграммы и моделей SQLModel.

Вяткина А. П. – разработчик проекта. В обязанности Вяткиной А. П. входили настройка подключения к базе данных, реализация авторизации и разграничения ролей.

Оба разработали основные API-эндпоинты на FastAPI, писали сервисы для работы с пользователями и сотрудниками, а также вместе тестировали систему и оформляли отчёт.

# Предметная область

Предметной областью проекта является автоматизация электронного документооборота в рамках работы территориального управления МЧС по Санкт-Петербургу.

Система предназначена для хранения, просмотра и частичного редактирования информации, связанной с профессиональной деятельностью сотрудников. Основная цель проекта — обеспечить централизованный и безопасный доступ к данным о сотрудниках, их обучении, аттестации и задействованном оборудовании.

В системе предусмотрено два типа пользователей:

* Администратор — имеет полный доступ ко всем функциям: просмотр, редактирование, добавление и удаление записей в системе.
* Пользователь — получает доступ только к просмотру информации, без возможности внесения изменений.

Хранимая информация:

* данные о сотрудниках (ФИО, дата рождения, звание, должность),
* данные об аттестации и её статусе,
* данные о занятиях,
* данные об используемом оборудовании.

# 

# Ход выполнения нормализации

После выделения основных сущностей предметной области была выполнена поэтапная нормализация данных от 1НФ до 3НФ. Ниже представлен список сущностей и их атрибутов:

**Сотрудник (employee):**

Каждый сотрудник имеет фамилию, имя, отчество, дату рождения, а также связан с определённой должностью и званием. В качестве ключевого атрибута используется surrogate key — id. Атрибуты position и rank являются внешними ключами на соответствующие справочники должностей и званий. Все атрибуты обязательны к заполнению.

**Аттестация (attestation):**

Каждая аттестация относится к сотруднику (employee\_id), имеет тип (type — внешний ключ на справочник типов аттестаций), статус, дату проведения аттестации, а также поле причины, если аттестация не была пройдена. Ключевой атрибут — id.

**Занятия (exercise):**

Каждое занятие связано с сотрудником (employee\_id), содержит дату, тип (exercise\_type — внешний ключ) и адрес проведения. Используется surrogate key id.

**Отчёт по занятиям (exercises\_report):**

Отдельная сущность, отражающая плановые и фактические показатели: даты начала и окончания, запланированное и фактическое количество проведенных занятий, а также комментарии. Имеет собственный id.

**Должность (position):**

Справочник должностей, включает наименование и группу. Первичный ключ — id.

**Звание (rank):**

Справочник званий, включает наименование и срок подготовки. Первичный ключ — id.

**Типы аттестаций и занятий (attestation\_type, exercise\_type):**

Отдельные справочники, каждый содержит уникальный идентификатор и наименование.

В ходе нормализации:

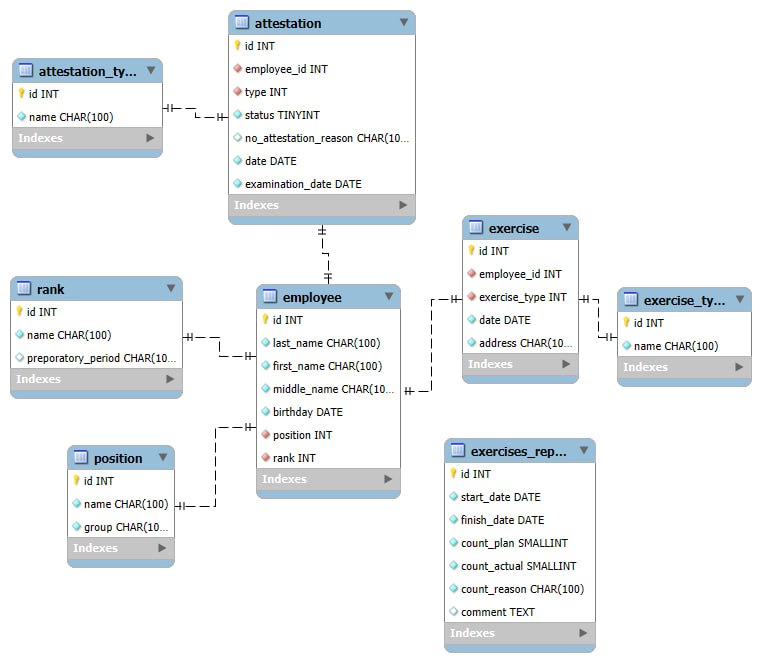
* Устранены дублирующиеся данные за счёт вынесения повторяющихся атрибутов в отдельные справочники;
* Введены surrogate-ключи для обеспечения уникальности;
* Реализованы связи между сущностями по внешним ключам;
* Таблицы приведены к 3НФ, транзитивные зависимости устранены.

На основе нормализованных сущностей построена ER-диаграмма, отражающая структуру базы данных и взаимосвязи между объектами системы.

# Объяснение выбранной СУБД

Для реализации проекта была выбрана СУБД MySQL в связке с MySQL Workbench. Данный выбор был обусловлен тем, что Workbench используется для удобного проектирования ER-диаграмм и генерации SQL-кода, а также простотой, надёжностью и широким распространением MySQL в учебных и рабочих проектах.

# ER–диаграмма



# Исходный текст запросов

## По созданию таблиц

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

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

-- Schema mchs\_project

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

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

-- Schema mchs\_project

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

CREATE SCHEMA IF NOT EXISTS mchs\_project DEFAULT CHARACTER SET utf8mb3 ;

USE mchs\_project ;

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

-- Table mchs\_project.attestation\_type

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

CREATE TABLE IF NOT EXISTS mchs\_project.attestation\_type (

id INT NOT NULL AUTO\_INCREMENT,

name CHAR(100) NOT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.position

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

CREATE TABLE IF NOT EXISTS mchs\_project.position (

id INT NOT NULL AUTO\_INCREMENT,

name CHAR(100) NOT NULL,

group CHAR(100) NOT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.rank

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

CREATE TABLE IF NOT EXISTS mchs\_project.rank (

id INT NOT NULL AUTO\_INCREMENT,

name CHAR(100) NOT NULL,

preporatory\_period CHAR(100) NULL DEFAULT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.employee

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

CREATE TABLE IF NOT EXISTS mchs\_project.employee (

id INT NOT NULL AUTO\_INCREMENT,

last\_name CHAR(100) NOT NULL,

first\_name CHAR(100) NOT NULL,

middle\_name CHAR(100) NOT NULL,

birthday DATE NOT NULL,

position INT NOT NULL,

rank INT NOT NULL,

PRIMARY KEY (id),

INDEX employee\_to\_position\_idx (position ASC) VISIBLE,

INDEX employee\_to\_rank\_idx (rank ASC) VISIBLE,

CONSTRAINT employee\_to\_position

FOREIGN KEY (position)

REFERENCES mchs\_project.position (id),

CONSTRAINT employee\_to\_rank

FOREIGN KEY (rank)

REFERENCES mchs\_project.rank (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.attestation

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

CREATE TABLE IF NOT EXISTS mchs\_project.attestation (

id INT NOT NULL AUTO\_INCREMENT,

employee\_id INT NOT NULL,

type INT NOT NULL,

status TINYINT NOT NULL,

no\_attestation\_reason CHAR(100) NULL DEFAULT NULL,

date DATE NOT NULL,

examination\_date DATE NOT NULL,

PRIMARY KEY (id),

INDEX attestation\_to\_employee\_idx (employee\_id ASC) VISIBLE,

INDEX attestation\_to\_attestation\_type\_idx (type ASC) VISIBLE,

CONSTRAINT attestation\_to\_attestation\_type

FOREIGN KEY (type)

REFERENCES mchs\_project.attestation\_type (id),

CONSTRAINT attestation\_to\_employee

FOREIGN KEY (employee\_id)

REFERENCES mchs\_project.employee (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.exercise\_type

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

CREATE TABLE IF NOT EXISTS mchs\_project.exercise\_type (

id INT NOT NULL AUTO\_INCREMENT,

name CHAR(100) NOT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.exercise

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

CREATE TABLE IF NOT EXISTS mchs\_project.exercise (

id INT NOT NULL AUTO\_INCREMENT,

employee\_id INT NOT NULL,

exercise\_type INT NOT NULL,

date DATE NOT NULL,

address CHAR(100) NOT NULL,

PRIMARY KEY (id),

INDEX exercise\_to\_employee\_idx (employee\_id ASC) VISIBLE,

INDEX exercise\_to\_exercise\_type\_idx (exercise\_type ASC) VISIBLE,

CONSTRAINT exercise\_to\_employee

FOREIGN KEY (employee\_id)

REFERENCES mchs\_project.employee (id),

CONSTRAINT exercise\_to\_exercise\_type

FOREIGN KEY (exercise\_type)

REFERENCES mchs\_project.exercise\_type (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

-- Table mchs\_project.exercises\_report

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

CREATE TABLE IF NOT EXISTS mchs\_project.exercises\_report (

id INT NOT NULL AUTO\_INCREMENT,

start\_date DATE NOT NULL,

finish\_date DATE NOT NULL,

count\_plan SMALLINT NOT NULL,

count\_actual SMALLINT NOT NULL,

count\_reason CHAR(100) NOT NULL,

comment TEXT NULL DEFAULT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

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

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

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