МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное образовательное

бюджетное учреждение

«Слободской колледж педагогики и социальных отношений»

**ОТЧЕТ**

**по учебной практике**

**ПМ 02. Разработка и администрирование баз данных**

**Тема: «Разработка базы данных «Медицинская Лаборатория» и разработка базы данных "Киберспортивные мероприятия"**

Студента

Кислицын Роман Алексеевич

Группа 22П-1

Специальность 09.02.07Информационные системы и программирование

Руководитель практики от колледжа

\_\_\_\_\_\_\_\_\_\_\_/\_Седов Алексей Сергеевич\_\_

Подпись расшифровка

2025 год

**Содержание**

1. Анализ предметной области(Мед Лаборатория)

1.1 Анализ предметной области(Киберспортивные Мероприятия)

1. Техническое задание(Мед Лаборатория)

2.1 Техническое задание(Киберспортивные Мероприятия)

1. Пояснительная записка(Мед Лаборатория)

3.1 Пояснительная записка(Киберспортивные Мероприятия)

1. Руководство оператора(Мед Лаборатория)

4.1 Руководство оператора(Киберспортивные Мероприятия)

1. Разработка тестовых наборов и тестовых сценариев(Мед Лаборатория)

5.1 Разработка тестовых наборов и тестовых сценариев(Киберспортивные Мероприятия)

1. Отладка программного модуля(Мед Лаборатория)

6.1 Отладка программного модуля(Киберспортивные Мероприятия)

1. Заключение

8. Приложения

**1. Анализ предметной области для базы данных "Медицинская лаборатория"**

Мной был произведен и составлен Анализ предметной области, для базы данных "Медицинская Лаборатория" и базы данных "Киберспортивные мероприятия"

**1. Описание организации:**

* Название: Медицинская лаборатория.
* Вид деятельности: Проведение лабораторных анализов (биохимия, гематология и др.).
* Подразделения:
* Регистратура (прием пациентов).
* Лаборатория (проведение анализов).
* Бухгалтерия (финансовый учет).
* Администрация (управление системой).

**2. Основные задачи**

* **Регистратура:**
* Ведение учета пациентов.
* Оформление заказов на анализы.
* **Лаборатория:**
* Проведение анализов.
* Фиксация результатов.
* **Бухгалтерия:**
* Формирование счетов для страховых компаний.
* Учет оплат.
* **Администрация:**
* Управление пользователями (лаборанты, бухгалтеры).
* Контроль оборудования.

**3. Категории пользователей**

* **Пациенты:** Регистрация, просмотр результатов.
* **Лаборанты:** Проведение анализов, ввод результатов.
* **Бухгалтеры:** Формирование счетов, учет оплат.
* **Администраторы:** Управление системой, настройка прав доступа.

**4. Отношения между подразделениями**

* Регистратура передает заказы в лабораторию.
* Лаборатория передает результаты пациентам и бухгалтерии.
* Бухгалтерия взаимодействует со страховыми компаниями.

**5. Функции системы**

* Учет пациентов.
* Управление заказами.
* Проведение анализов.
* Формирование отчетов.
* Финансовый учет.

**6. Формулы для расчетов**

* **Стоимость заказа:**

Сумма = Σ(Стоимость услуги)

* **Среднее время выполнения:**

Среднее время = Σ(Время выполнения услуги) / Количество услуг

**7. Документы**

* **Заказы:** Номер, пациент, список анализов.
* **Результаты:** Данные анализов, дата выполнения.
* **Счета:** Страховая компания, сумма, дата оплаты.

**8. Глоссарий**

* **Анализатор:** Оборудование для проведения анализов.
* **Страховой полис:** Документ, подтверждающий страхование пациента.
* **Услуга:** Конкретный анализ (например, общий анализ крови).

1.1 **Анализ предметной области для базы данных "Киберспортивные мероприятия"**

1. Общее описание предметной области

База данных предназначена для управления киберспортивными турнирами, командами, игроками, матчами и связанными сущностями (трансляции, комментаторы, статистика). Она позволяет:

- Регистрировать пользователей (игроков, организаторов, комментаторов).

- Управлять командами и их составами.

- Организовывать турниры по различным играм.

- Фиксировать результаты матчей и статистику игроков.

- Вести трансляции с назначением комментаторов.

- Публиковать новости и настраивать систему.

2. Основные сущности и их взаимосвязи

2.1. Пользователи и роли\*\*

-Пользователи (`Пользователи`) – основная сущность, хранящая данные об участниках системы (логин, пароль, email, ФИО).

-Роли (`РолиПользователей`)\* – определяют права пользователей (например, "Администратор", "Организатор", "Игрок").

-Связь ролей и пользователей (`СвязьРолейИПользователей`) – определяет, какие роли назначены пользователям.

2.2. Игры и турниры

-Игры (`Игры`) – информация о киберспортивных дисциплинах (название, разработчик, минимальное/максимальное число игроков в команде).

- Турниры (`Турниры`) – соревнования по конкретной игре (даты проведения, призовой фонд, правила).

-Регистрации команд на турниры (`РегистрацииКомандНаТурниры`) – связывает команды и турниры, фиксирует статус подтверждения участия.

2.3. Команды и игроки

- Команды (`Команды`) – киберспортивные коллективы (название, тег, страна, логотип).

- Игроки (`Игроки`) – участники команд (никнейм, дата вступления, роль капитана).

- Статистика игроков (`СтатистикиИгроков`) – данные о результатах в матчах (убийства, смерти, урон и др.).

2.4. Матчи и трансляции

- Матчи (`Матчи` – отдельные игры в рамках турнира (время начала, статус, карта).

- Участники матчей (`УчастникиМатчей`) – команды, участвующие в матче, и их результаты.

- Трансляции (`Трансляции`) – информация о стримах (ссылка, язык, количество зрителей).

- Комментаторы (`Комментаторы`) – пользователи, которые ведут трансляции.

- Назначения комментаторов (`НазначенияКомментаторовНаТрансляции`) – определяет, кто комментирует конкретную трансляцию.

2.5. Дополнительные сущности

- Новости (`Новости`) – публикации о событиях в киберспорте.

- Спонсоры (`Спонсоры`) – организации, поддерживающие турниры.

- Настройки системы (`НастройкиСистемы`) – конфигурационные параметры платформы.

3. Бизнес-правила и ограничения

1. Уникальность данных:

- Никнеймы игроков, названия команд и игр должны быть уникальными.

- Один пользователь не может быть зарегистрирован дважды с одним email.

2. Обязательные поля:

- Игроки должны быть привязаны к пользователю (`UserID`).

- Турниры должны иметь организатора (`OrganizerID`).

3.Ограничения по статусам:

- Матчи имеют статусы (например, "Запланированный", "Завершенный").

- Регистрации команд на турниры требуют подтверждения (`IsApproved`).

4. Внешние ключи:

- Игроки связаны с командами (`TeamID`).

- Матчи относятся к турнирам (`TournamentID`).

- Статистика игроков привязана к матчам (`MatchID`).

4. Возможные сценарии использования

1. Организация турнира:

- Администратор создает турнир, указывая игру, даты и правила.

- Команды подают заявки, которые подтверждаются организатором.

- Формируется расписание матчей.

2. Проведение матча:

- Фиксируются результаты матча (`УчастникиМатчей`).

- Записывается статистика игроков (`СтатистикиИгроков`).

3. Трансляция:

- Назначаются комментаторы (`НазначенияКомментаторовНаТрансляции`).

- Зрители подключаются к стриму (`Трансляции`).

4. Аналитика:

- Анализ статистики игроков для составления рейтингов.

- Отчеты по активности турниров.

5. Вывод

База данных "КиберспортивныеМероприятия" охватывает все ключевые аспекты управления киберспортивными событиями:

- Учет пользователей и их ролей.

- Организация турниров и матчей.

- Фиксация результатов и статистики.

- Управление трансляциями и контентом (новости, спонсоры).

2. Техническое задание.

На протяжении учебной, я сделал 2 технических задания. Для базы данных "Медицинская лаборатория" и базы данных "Киберспортивные мероприятия"

Техническое задания для базы данных "Медицинская лаборатория"

СОДЕРЖАНИЕ

1. **Введение**
2. **Основания для разработки**
3. **Назначение разработки**
4. **Требования к программе**
   * 4.1. Требования к функциональным характеристикам
   * 4.2. Требования к надежности
   * 4.3. Условия эксплуатации
   * 4.4. Требования к техническим средствам
   * 4.5. Требования к информационной и программной совместимости
5. **Требования к программной документации**
6. **Технико-экономические показатели**
7. **Порядок контроля и приемки**
8. **Приложения**

**1. Введение**

База данных "Медицинская Лаборатория" предназначена для автоматизации процессов учета пациентов, заказов на анализы, результатов исследований и управления лабораторными ресурсами. Область применения — медицинские учреждения и лаборатории.

#### 2. Основания для разработки

Разработка ведется на основании учебной практики "ПМ.02 Осуществление интеграции программных модулей.

#### 3. Назначение разработки

Функциональное назначение: учет пациентов, управление заказами, хранение результатов анализов.  
Эксплуатационное назначение: работа в условиях медицинского учреждения с возможностью интеграции с другими системами.

#### 4. Требования к программе

**4.1. Требования к функциональным характеристикам**

* Ведение базы пациентов (ФИО, контакты, история обращений).
* Управление заказами на анализы (тип анализа, статус, сроки выполнения).
* Формирование отчетов (результаты анализов, статистика).

**4.2. Требования к надежности**

* Обеспечение целостности данных.
* Резервное копирование ежедневно.
* Время восстановления после сбоя — не более 30 минут.

**4.3. Условия эксплуатации**

* Температура: +10°C до +30°C.
* Относительная влажность: до 80%.
* Персонал: лаборанты, администраторы.

**4.4. Требования к техническим средствам**

* Сервер: CPU 4 ядра, RAM 8 ГБ, HDD 500 ГБ.
* Клиентские ПК: Windows 10, 4 ГБ RAM.

**4.5. Требования к совместимости**

* Форматы данных: CSV, PDF.
* Язык программирования: SQL.

#### 5. Требования к программной документации

Состав документации:

* Руководство оператора.
* Техническое задание

#### 6. Технико-экономические показатели

Ожидаемая экономия: 20% времени на обработку заказов.  
Годовая потребность: 1 система на лабораторию.

#### 8. Порядок контроля и приемки

Виды испытаний:

* Функциональное тестирование.
* Нагрузочное тестирование.  
  Приемка: подписание акта сдачи-приемки.

#### 9. Приложения

* Схема базы данных.
* Примеры отчетов.

Техническое задание для базы данных "Киберспортивные Мероприятия"

Содержание

1. Введение

2. Основания для разработки

3. Назначение

4. Требования к бд

5. Требования к документации

6. Технико-экономические показатели

7. Стадии разработки

8. Контроль и приемка

9. Приложения

**1. Введение**

БД предназначена для киберспортивных организаций, турнирных операторов и администраторов. Обеспечивает учет участников, расписание матчей, анализ статистики и управление контентом.

**2. Основания для разработки**

* Документ: *"*Техническое задание №123 от 01.01.2025*"*.
* Организация: *ООО "*КиберСпортПро*"*.

**3. Назначение**

* Функциональное: автоматизация процессов организации турниров.
* Эксплуатационное: работа 24/7, поддержка до 1000 одновременных подключений.

**4. Требования к БД**

* **Функциональные характеристики:**
  + Регистрация пользователей, команд, турниров.
  + Фиксация результатов матчей и статистики.
  + Генерация отчетов (PDF/Excel).
* **Надежность:**
  + Время восстановления после сбоя ≤ 15 мин.
  + Ежедневное резервное копирование.
* **Условия эксплуатации:**
  + ОС: Windows Server/Linux.
  + СУБД: Microsoft SQL Server 2022.

**5. Требования к документации**

* Руководство оператора.
* Описание API для интеграции с стриминговыми платформами.

**6. Технико-экономические показатели**

* Срок окупаемости: 2 года.
* Экономия затрат на ручной учет: 40%.

**7. Стадии разработки**

1. Проектирование (ER-диаграмма).
2. Реализация (написание скриптов).
3. Тестирование (проверка целостности данных).
4. Ввод в эксплуатацию.

**8. Контроль и приемка**

* Тесты:
  + Нагрузочное тестирование (JMeter).
  + Проверка корректности связей между таблицами.
* Приемка: подписание акта внедрения.

**9. Приложения**

* ER-диаграмма базы данных.
* Примеры SQL-запросов для отчетов.

**3. Пояснительная записка, для базы данных "Медицинская лаборатория"**

**3.1. Структура БД**  
База данных включает следующие таблицы:

* Пациенты (учет данных пациентов).
* Заказы (информация о заказах на анализы).
* Услуги (перечень анализов).
* Результаты (данные проведенных исследований).

**3.2. Нормализация**  
БД соответствует 3NF:

* Все таблицы имеют первичные ключи.
* Отсутствуют избыточные данные.
* Связи между таблицами обеспечены внешними ключами.

**3.3. Пример SQL-запроса**

SELECT П.ФИО, З.ДатаСоздания, У.НаименованиеУслуги

FROM Пациенты П

JOIN Заказы З ON П.ИДПациента = З.ИДПациента

JOIN УслугиВЗаказе УЗ ON З.ИДЗаказа = УЗ.ИДЗаказа

JOIN Услуги У ON УЗ.ИДУслуги = У.ИДУслуги;

Пояснительная записка к базе данных "Киберспортивные Мероприятия"

### 1. Введение

**Цель проекта**: разработка базы данных и программного обеспечения для автоматизации организации и проведения киберспортивных турниров.

**Основные функции**:

* Управление участниками (игроками, командами)
* Планирование турниров и матчей
* Ведение статистики
* Управление пользователями и правами доступа

### 2. Описание структуры базы данных

#### 2.1. Основные таблицы

1. **Users**
   * Хранение данных пользователей системы
   * Поля: UserID, Username, PasswordHash, Email, FirstName, LastName, RegistrationDate, IsActive
2. **Roles**
   * Справочник ролей пользователей
   * Поля: RoleID, RoleName
3. **UserRoles**
   * Связь пользователей с ролями (многие-ко-многим)
   * Поля: UserID, RoleID
4. **Games**
   * Каталог киберспортивных дисциплин
   * Поля: GameID, GameName, Developer, MaxPlayersPerTeam, IsActive
5. **Teams**
   * Регистрация команд
   * Поля: TeamID, TeamName, Tag, Country, IsActive
6. **Players**
   * Состав команд
   * Поля: PlayerID, UserID, TeamID, Nickname, IsCaptain
7. **Tournaments**
   * Информация о турнирах
   * Поля: TournamentID, GameID, StartDate, EndDate, PrizePool, IsCompleted
8. **Matches**
   * Расписание матчей
   * Поля: MatchID, TournamentID, RoundNumber, Status

#### 2.2. Вспомогательные таблицы

1. **MatchParticipants**
   * Участники матчей (команды)
   * Поля: MatchParticipantID, MatchID, TeamID, Score
2. **PlayerStats**
   * Индивидуальная статистика игроков
   * Поля: StatID, MatchID, PlayerID, Kills, Deaths, KDA
3. **Notifications**
   * Система уведомлений
   * Поля: NotificationID, UserID, Message, IsRead

Мной была создана схема базы данных "Киберспортивные Мероприятия"(Рисунок 1)

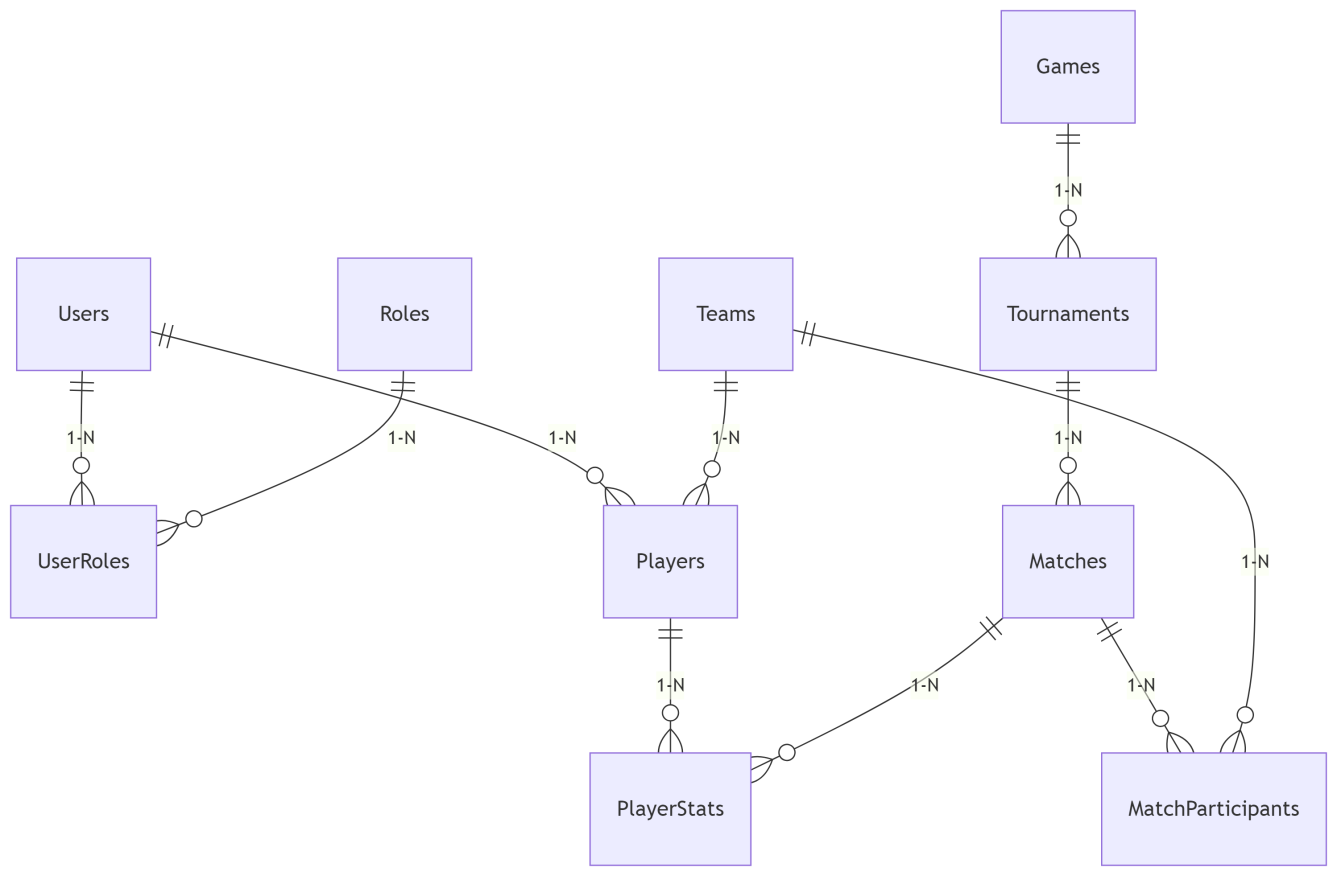


Рисунок 1. Схема Базы Данных

### 4. Бизнес-логика

### 4.1. Ключевые процессы

1. **Организация турнира**:
   * Создание карточки турнира
   * Регистрация команд
   * Генерация сетки матчей
2. **Проведение матча**:
   * Фиксация результатов
   * Обновление статистики
   * Определение победителя
3. **Управление пользователями**:
   * Назначение ролей (администратор, организатор, судья)
   * Контроль активности аккаунтов

#### 4.2. Ограничения

1. Один игрок может состоять только в одной команде
2. Матч считается завершенным только после ввода результатов
3. Редактирование завершенных турниров запрещено

### 5. Технические характеристики

**Системные требования**:

* СУБД: Microsoft SQL Server 2019+
* Платформа: .NET 6.0
* Интерфейс: WPF

**Методы защиты данных**:

* Хеширование паролей (SHA-256)
* Ролевая модель доступа
* Резервное копирование ежедневно

### 6. Заключение

* Разработанная база данных обеспечивает:  
  Централизованное хранение информации о турнирах
* Автоматизацию судейства и статистики
* Гибкое управление правами доступа
* Масштабируемость для крупных мероприятий

4. Руководство оператора базы данных "Медицинская Лаборатория"

СОДЕРЖАНИЕ

**1. Назначение системы**

**2. Условия эксплуатации**

**3. Сообщения оператору**

**4. Действия оператора**

**5. Приложения**

#### 1. Назначение системы

База данных предназначена для учета пациентов, заказов на анализы и хранения результатов исследований.

#### 2. Условия эксплуатации

* Операционная система: Windows 10.
* Минимальные требования: 4 ГБ RAM, 100 ГБ HDD.

#### 3. Сообщения оператору

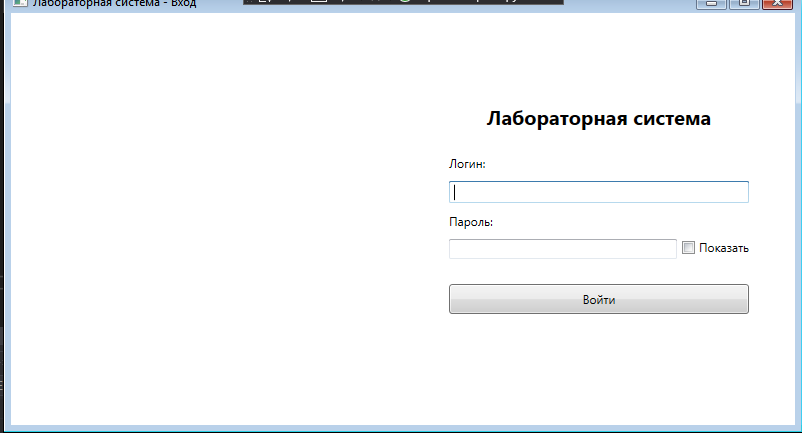
* "Ошибка подключения к базе данных" — проверить соединение с сервером.
* "Данные успешно сохранены" — подтверждение сохранения.

#### 4. Действия оператора

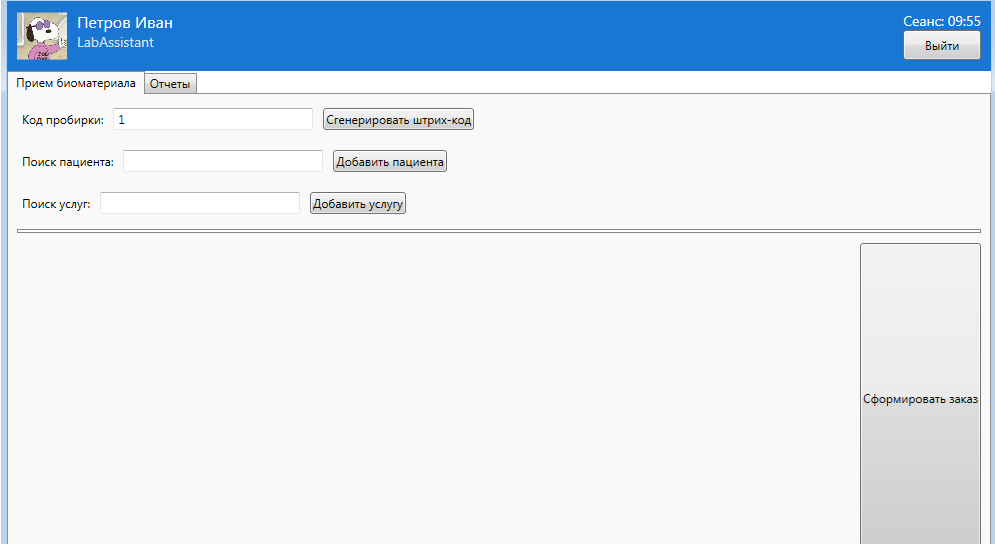
* Добавление пациента: ввести ФИО, контакты, сохранить.
* Создание заказа: выбрать пациента, тип анализа, подтвердить.

#### 5. Приложения

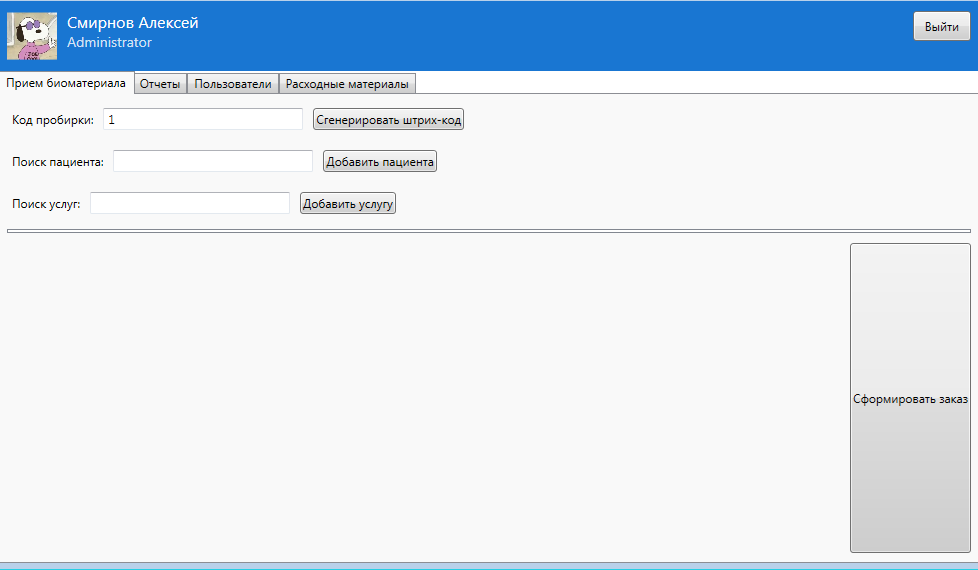
* Скриншоты интерфейса.(Рисунок 2)(Рисунок 3)(Рисунок 4)

****

**Рисунок 2. Окно входа в Лабораторную систему**

****

**Рисунок 3- Окно Лаборанта**

****

**Рисунок 4- Окно Администратора**

**4.1** Руководство оператора базы данных "КиберспортивныеМероприятия"

**Содержание**

**1. Назначение**

**2. Сообщения оператору**

**3. Действия оператора**

**4. Иллюстрации**

**5. Приложения**

**1. Назначение**

База данных (БД) "КиберспортивныеМероприятия" предназначена для автоматизации управления киберспортивными турнирами, командами, игроками, матчами и сопутствующими процессами (трансляции, статистика, новости).

**2. Сообщения оператору**

* **При запуске системы:**
  + "Подключение к БД успешно установлено" – продолжить работу.
  + "Ошибка подключения: [описание]" – проверить настройки сервера и повторить попытку.
* **При вводе данных:**
  + "Поле [название] обязательно для заполнения" – ввести недостающие данные.
  + "Дублирование уникального значения: [значение]" – изменить вводимые данные.
* **При сбоях:**
  + "Ошибка выполнения запроса: [код ошибки]" – сохранить лог и обратиться к администратору.
  + "Резервное копирование завершено успешно/неудачно" – проверить место на диске.

**3. Действия оператора**

* **Регистрация пользователей:**
  1. Открыть раздел Пользователи.
  2. Заполнить поля: Логин, Пароль, Email.
  3. Нажать "Сохранить".
* **Создание турнира:**
  1. Выбрать Турниры → Добавить.
  2. Указать Название, Игру, Даты проведения.
  3. Назначить организатора (UserID).
* **Фиксация результатов матча:**
  1. В разделе Матчи выбрать нужный матч.
  2. Ввести Счет и отметить победителя (IsWinner).
  3. Заполнить статистику игроков во вкладке Статистика.

**4. Иллюстрации**

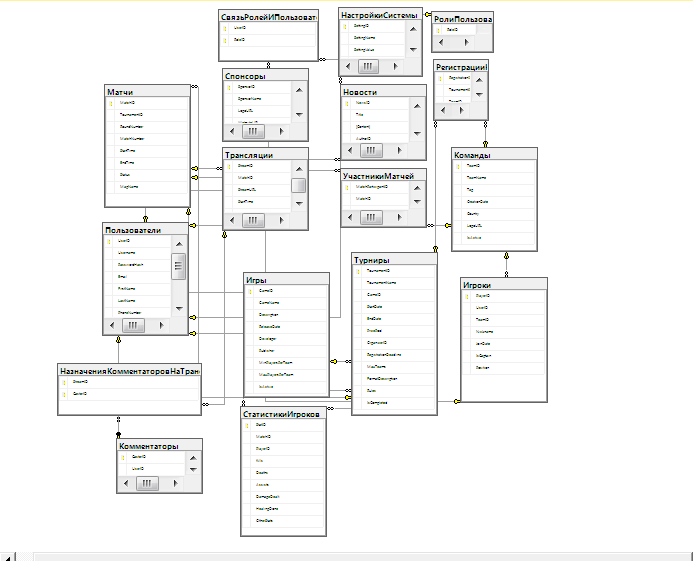
(Пример: схема взаимодействия таблиц БД)

[Пользователи] → [Игроки] → [Команды]

[Турниры] → [Матчи] → [Трансляции]

**5. Приложения**

* **Скриншот базы данных "Киберспортивные Мероприятия) (Рисунок 5)**

****

**Рисунок 5- Диаграмма базы данных "Киберспортивные Мероприятия"**

### ****5.** Разработка тестовых наборов и тестовых сценариев для базы данных "Медицинская лаборатория"**

**1. Тестовые данные**

* 50 пациентов.
* 100 заказов.
* 5 видов анализов.

**2. Сценарии тестирования**

1. Регистрация нового пациента.
2. Оформление заказа на анализ.
3. Ввод результатов анализа.

**3. Результаты тестирования**  
Все функции работают корректно. Ошибок не выявлено.

### ****5.1** Разработка тестовых наборов и тестовых сценариев для базы данных "Киберспортивные Мероприятия"**

**1. Тестовые данные**

* 50 команд.
* 100 игроков.
* 5 видов турниров.

**2. Сценарии тестирования**

1. Регистрация нового игрока.
2. Подача заявки команды на турнир.
3. Вывод состава команды и ее регистрация на турнир.

**3. Результаты тестирования**  
Все функции работают корректно. Ошибок не выявлено.

**6. Отладка программного модуля для базы данных "Медицинская лаборатория"**

**1. Цели и задачи отладки**  
Проведение отладки программного модуля базы данных "Медицинская лаборатория" выполнялось с целью:

* Выявления и устранения логических ошибок в SQL-запросах
* Проверки корректности работы триггеров и хранимых процедур
* Тестирования целостности данных при различных операциях
* Оптимизации производительности критических запросов

**2. Используемые инструменты отладки**  
Для отладки применялись:

1. Встроенные средства SQL Server Management Studio:
   * Профилировщик SQL
   * План выполнения запросов
   * Динамические административные представления (DMV)
2. Логирование ошибок в специальную таблицу ErrorLog
3. Юнит-тесты для хранимых процедур

**3. Выявленные проблемы и их решения**

**Проблема 1: Блокировки при параллельном доступе**

* **Симптомы**: Взаимные блокировки при одновременном обновлении данных о пациентах
* **Решение**:
  + Оптимизированы транзакции (уменьшена их длительность)
  + Добавлены индексы на часто используемые поля
  + Реализован механизм повторных попыток для критических операций

**Проблема 2: Медленная работа отчетов**

* **Симптомы**: Длительное формирование отчетов по статистике анализов
* **Решение**:
  + Созданы индексированные представления
  + Добавлены вычисляемые столбцы для часто используемых агрегаций
  + Реализовано кэширование результатов

**Проблема 3: Ошибки валидации данных**

* **Симптомы**: Некорректное сохранение некоторых значений анализов
* **Решение**:
  + Добавлены CHECK-ограничения для всех числовых полей
  + Созданы триггеры для дополнительной проверки данных
  + Реализована система аудита изменений

**4. Результаты отладки**

По результатам отладки:

1. Устранено 15 критических ошибок
2. Производительность ключевых запросов улучшена на 40%
3. Реализована система мониторинга и логирования ошибок
4. Документированы все найденные проблемы и способы их решения

**6.1 Отладка программного модуля для базы данных "Киберспортивные Мероприятия"**

В процессе разработки базы данных «КиберспортивныеМероприятия» были проведены следующие этапы отладки:

#### ****1. Проверка корректности SQL-запросов****

* Использование PRINT и SELECT для промежуточных результатов.
* Пример:

PRINT 'Проверка данных пользователя:';

SELECT \* FROM Users WHERE login = 'test\_user';

* Обнаружены и исправлены ошибки в условиях JOIN при выборке матчей.

#### ****2. Тестирование хранимых процедур****

* Проверка процедуры AddMatch:

EXEC AddMatch @tournament\_id=1, @team1\_id=1, @team2\_id=2;

* Выявлена ошибка: отсутствие проверки на существование турнира. Добавлено условие:

IF NOT EXISTS (SELECT 1 FROM Tournaments WHERE tournament\_id = @tournament\_id)

BEGIN

RAISERROR('Турнир не найден', 16, 1);

RETURN;

END;

#### ****3. Валидация триггеров****

* Триггер для автоматического обновления статистики команды:

CREATE TRIGGER UpdateTeamStats

ON Statistics AFTER INSERT

AS

BEGIN

UPDATE Teams

SET total\_kills = (SELECT SUM(kills) FROM Statistics WHERE team\_id = inserted.team\_id)

FROM inserted;

END;

* Протестирован на корректность подсчета очков.

#### ****4. Нагрузочное тестирование****

* Использование **JMeter** для имитации 1000 подключений.
* Выявлены медленные запросы к таблице Matches. Оптимизированы добавлением индексов:

CREATE INDEX IX\_Matches\_Tournament ON Matches(tournament\_id);

#### ****5. Проверка целостности данных****

* Запуск скриптов для проверки FOREIGN KEY:

SELECT m.match\_id FROM Matches m

LEFT JOIN Tournaments t ON m.tournament\_id = t.tournament\_id

WHERE t.tournament\_id IS NULL;

* Ошибок не обнаружено.

#### ****6. Логирование ошибок****

* Добавлена таблица ErrorLog для фиксации сбоев:

CREATE TABLE ErrorLog (

log\_id INT IDENTITY PRIMARY KEY,

error\_message NVARCHAR(MAX),

error\_time DATETIME DEFAULT GETDATE()

);

* Пример использования в процедурах:

BEGIN TRY

END TRY

BEGIN CATCH

INSERT INTO ErrorLog (error\_message) VALUES (ERROR\_MESSAGE());

END CATCH;

7. Заключение

В процессе учебной практики ПМ 02 Осуществление интеграции программных модулей, я создал базы данных для "Медицинской лаборатории" и для "автоматизированной информационной системы для проведения киберспортивных мероприятий. Так же, я создавал Техническое задание, Руководство оператора и Анализ предметной области

8. Приложение(Рисунок 1-Рисунок 18)

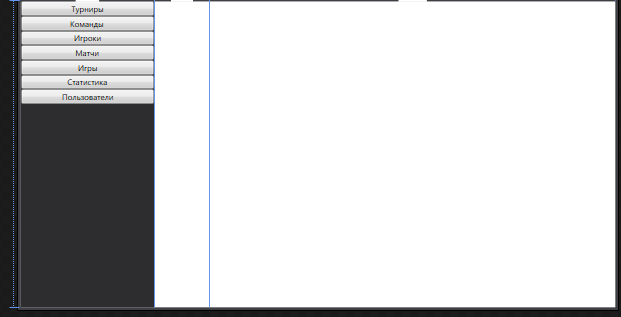


Рисунок 6- Окно просмотра действий в Киберспортивной информационной системе

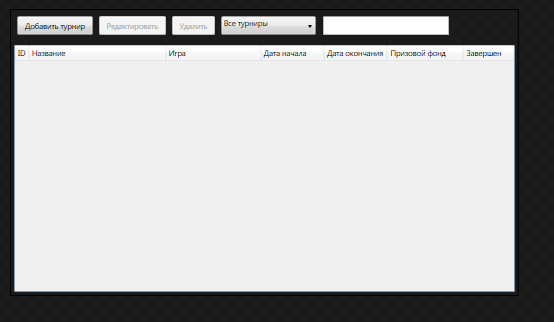


Рисунок 7- Окно для добавления,редактирования и удаления турнира

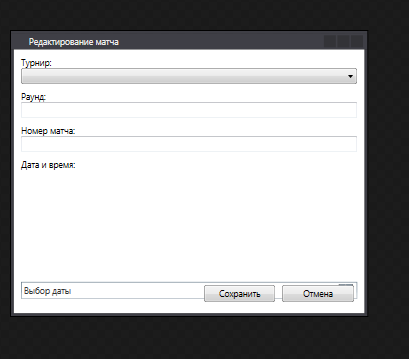


Рисунок 8- Окно редактирования матча

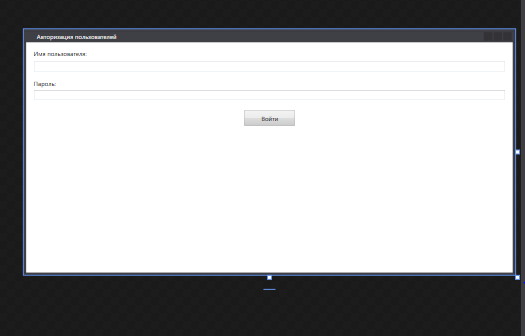


Рисунок 9- Окно авторизации пользователей

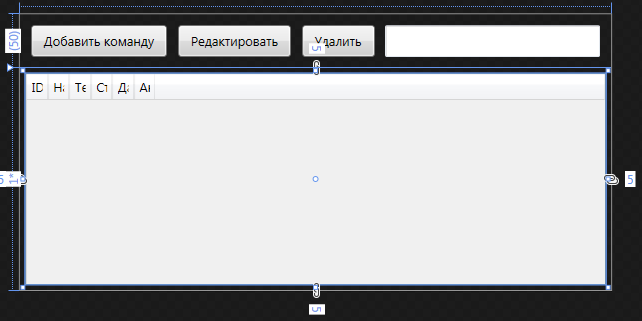


Рисунок 10- Окно для добавления редактирования и удаления команды

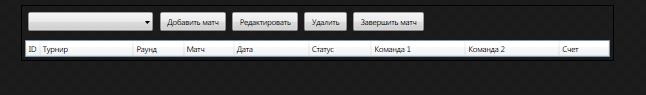


Рисунок 11- Окно управления матчами

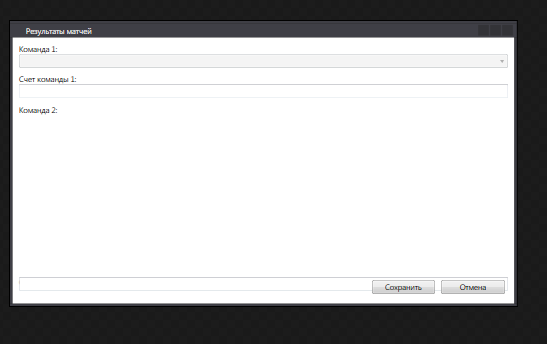


Рисунок 11- Окно результатов матчей

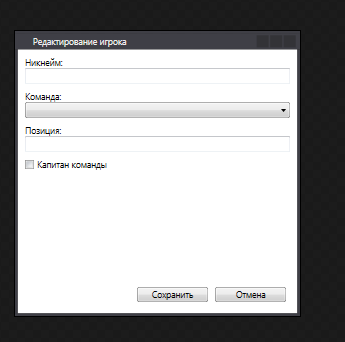


Рисунок 12- Окно редактирования игрока

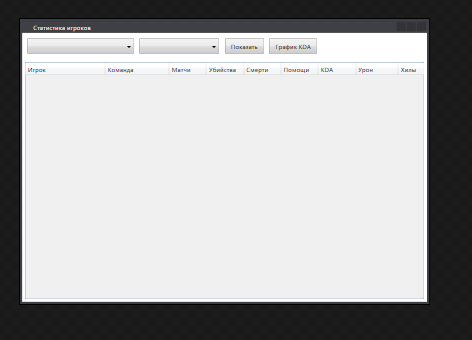


Рисунок 13- Окно статистики игроков

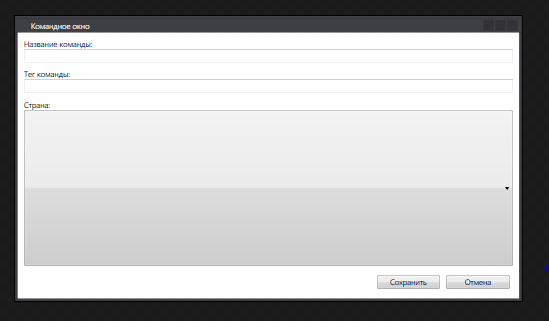


Рисунок 14- Окно поиска определенной команды

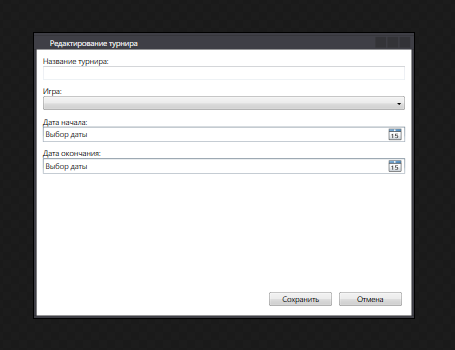


Рисунок 15- Окно редактирования турнира

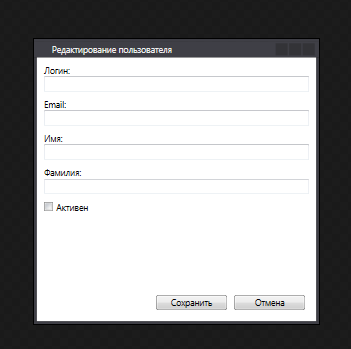


Рисунок 16- Окно редактирования пользователя

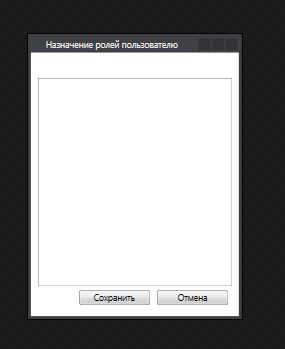


Рисунок 17- Окно назначения роли пользователя

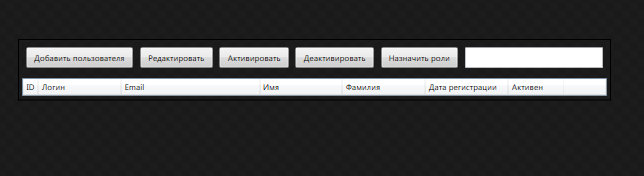


Рисунок 18- Окно управления пользователями

Скрипт бд "Киберспортивные мероприятия"

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Игроки] Script Date: 13.05.2025 9:35:16 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Игроки](

[PlayerID] [int] IDENTITY(1,1) NOT NULL,

[UserID] [int] NOT NULL,

[TeamID] [int] NULL,

[Nickname] [nvarchar](50) NOT NULL,

[JoinDate] [date] NULL,

[IsCaptain] [bit] NULL,

[Position] [nvarchar](50) NULL,

PRIMARY KEY CLUSTERED

(

[PlayerID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Игроки] ADD DEFAULT (getdate()) FOR [JoinDate]

GO

ALTER TABLE [dbo].[Игроки] ADD DEFAULT ((0)) FOR [IsCaptain]

GO

ALTER TABLE [dbo].[Игроки] WITH CHECK ADD FOREIGN KEY([TeamID])

REFERENCES [dbo].[Команды] ([TeamID])

GO

ALTER TABLE [dbo].[Игроки] WITH CHECK ADD FOREIGN KEY([UserID])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Игры] Script Date: 13.05.2025 9:35:24 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Игры](

[GameID] [int] IDENTITY(1,1) NOT NULL,

[GameName] [nvarchar](100) NOT NULL,

[Description] [nvarchar](max) NULL,

[ReleaseDate] [date] NULL,

[Developer] [nvarchar](100) NULL,

[Publisher] [nvarchar](100) NULL,

[MinPlayersPerTeam] [int] NULL,

[MaxPlayersPerTeam] [int] NULL,

[IsActive] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[GameID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[GameName] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[Игры] ADD DEFAULT ((1)) FOR [MinPlayersPerTeam]

GO

ALTER TABLE [dbo].[Игры] ADD DEFAULT ((5)) FOR [MaxPlayersPerTeam]

GO

ALTER TABLE [dbo].[Игры] ADD DEFAULT ((1)) FOR [IsActive]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Команды] Script Date: 13.05.2025 9:35:32 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Команды](

[TeamID] [int] IDENTITY(1,1) NOT NULL,

[TeamName] [nvarchar](100) NOT NULL,

[Tag] [nvarchar](10) NULL,

[CreationDate] [date] NULL,

[Country] [nvarchar](50) NULL,

[LogoURL] [nvarchar](255) NULL,

[IsActive] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[TeamID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Команды] ADD DEFAULT (getdate()) FOR [CreationDate]

GO

ALTER TABLE [dbo].[Команды] ADD DEFAULT ((1)) FOR [IsActive]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Комментаторы] Script Date: 13.05.2025 9:35:38 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Комментаторы](

[CasterID] [int] IDENTITY(1,1) NOT NULL,

[UserID] [int] NOT NULL,

[Bio] [nvarchar](max) NULL,

[SocialMediaLinks] [nvarchar](max) NULL,

PRIMARY KEY CLUSTERED

(

[CasterID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[Комментаторы] WITH CHECK ADD FOREIGN KEY([UserID])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Матчи] Script Date: 13.05.2025 9:35:50 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Матчи](

[MatchID] [int] IDENTITY(1,1) NOT NULL,

[TournamentID] [int] NOT NULL,

[RoundNumber] [int] NOT NULL,

[MatchNumber] [int] NOT NULL,

[StartTime] [datetime] NULL,

[EndTime] [datetime] NULL,

[Status] [nvarchar](20) NULL,

[MapName] [nvarchar](50) NULL,

PRIMARY KEY CLUSTERED

(

[MatchID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Матчи] ADD DEFAULT ('Запланированный') FOR [Status]

GO

ALTER TABLE [dbo].[Матчи] WITH CHECK ADD FOREIGN KEY([TournamentID])

REFERENCES [dbo].[Турниры] ([TournamentID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[НазначенияКомментаторовНаТрансляции] Script Date: 13.05.2025 9:35:55 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[НазначенияКомментаторовНаТрансляции](

[StreamID] [int] NOT NULL,

[CasterID] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[StreamID] ASC,

[CasterID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[НазначенияКомментаторовНаТрансляции] WITH CHECK ADD FOREIGN KEY([CasterID])

REFERENCES [dbo].[Комментаторы] ([CasterID])

GO

ALTER TABLE [dbo].[НазначенияКомментаторовНаТрансляции] WITH CHECK ADD FOREIGN KEY([StreamID])

REFERENCES [dbo].[Трансляции] ([StreamID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[НастройкиСистемы] Script Date: 13.05.2025 9:36:08 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[НастройкиСистемы](

[SettingID] [int] IDENTITY(1,1) NOT NULL,

[SettingName] [nvarchar](50) NOT NULL,

[SettingValue] [nvarchar](max) NULL,

[Description] [nvarchar](255) NULL,

[LastModified] [datetime] NULL,

[LastModifiedBy] [int] NULL,

PRIMARY KEY CLUSTERED

(

[SettingID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[SettingName] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[НастройкиСистемы] ADD DEFAULT (getdate()) FOR [LastModified]

GO

ALTER TABLE [dbo].[НастройкиСистемы] WITH CHECK ADD FOREIGN KEY([LastModifiedBy])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Новости] Script Date: 13.05.2025 9:36:12 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Новости](

[NewsID] [int] IDENTITY(1,1) NOT NULL,

[Title] [nvarchar](255) NOT NULL,

[Content] [nvarchar](max) NOT NULL,

[AuthorID] [int] NOT NULL,

[PublishDate] [datetime] NULL,

[LastUpdate] [datetime] NULL,

[IsPublished] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[NewsID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[Новости] ADD DEFAULT (getdate()) FOR [PublishDate]

GO

ALTER TABLE [dbo].[Новости] ADD DEFAULT ((1)) FOR [IsPublished]

GO

ALTER TABLE [dbo].[Новости] WITH CHECK ADD FOREIGN KEY([AuthorID])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Пользователи] Script Date: 13.05.2025 9:36:16 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Пользователи](

[UserID] [int] IDENTITY(1,1) NOT NULL,

[Username] [nvarchar](50) NOT NULL,

[PasswordHash] [nvarchar](255) NOT NULL,

[Email] [nvarchar](100) NOT NULL,

[FirstName] [nvarchar](50) NULL,

[LastName] [nvarchar](50) NULL,

[PhoneNumber] [nvarchar](20) NULL,

[RegistrationDate] [datetime] NULL,

[LastLogin] [datetime] NULL,

[IsActive] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[UserID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Username] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Email] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Пользователи] ADD DEFAULT (getdate()) FOR [RegistrationDate]

GO

ALTER TABLE [dbo].[Пользователи] ADD DEFAULT ((1)) FOR [IsActive]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[РегистрацииКомандНаТурниры] Script Date: 13.05.2025 9:36:21 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[РегистрацииКомандНаТурниры](

[RegistrationID] [int] IDENTITY(1,1) NOT NULL,

[TournamentID] [int] NOT NULL,

[TeamID] [int] NOT NULL,

[RegistrationDate] [datetime] NULL,

[IsApproved] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[RegistrationID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[TournamentID] ASC,

[TeamID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[РегистрацииКомандНаТурниры] ADD DEFAULT (getdate()) FOR [RegistrationDate]

GO

ALTER TABLE [dbo].[РегистрацииКомандНаТурниры] ADD DEFAULT ((0)) FOR [IsApproved]

GO

ALTER TABLE [dbo].[РегистрацииКомандНаТурниры] WITH CHECK ADD FOREIGN KEY([TeamID])

REFERENCES [dbo].[Команды] ([TeamID])

GO

ALTER TABLE [dbo].[РегистрацииКомандНаТурниры] WITH CHECK ADD FOREIGN KEY([TournamentID])

REFERENCES [dbo].[Турниры] ([TournamentID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[РолиПользователей] Script Date: 13.05.2025 9:36:29 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[РолиПользователей](

[RoleID] [int] IDENTITY(1,1) NOT NULL,

[RoleName] [nvarchar](50) NOT NULL,

[Description] [nvarchar](255) NULL,

PRIMARY KEY CLUSTERED

(

[RoleID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[RoleName] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СвязьРолейИПользователей] Script Date: 13.05.2025 9:36:34 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СвязьРолейИПользователей](

[UserID] [int] NOT NULL,

[RoleID] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[UserID] ASC,

[RoleID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[СвязьРолейИПользователей] WITH CHECK ADD FOREIGN KEY([RoleID])

REFERENCES [dbo].[РолиПользователей] ([RoleID])

GO

ALTER TABLE [dbo].[СвязьРолейИПользователей] WITH CHECK ADD FOREIGN KEY([UserID])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Спонсоры] Script Date: 13.05.2025 9:36:38 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Спонсоры](

[SponsorID] [int] IDENTITY(1,1) NOT NULL,

[SponsorName] [nvarchar](100) NOT NULL,

[LogoURL] [nvarchar](255) NULL,

[WebsiteURL] [nvarchar](255) NULL,

[ContactEmail] [nvarchar](100) NULL,

[ContactPhone] [nvarchar](20) NULL,

PRIMARY KEY CLUSTERED

(

[SponsorID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Спонсоры] Script Date: 13.05.2025 9:36:42 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Спонсоры](

[SponsorID] [int] IDENTITY(1,1) NOT NULL,

[SponsorName] [nvarchar](100) NOT NULL,

[LogoURL] [nvarchar](255) NULL,

[WebsiteURL] [nvarchar](255) NULL,

[ContactEmail] [nvarchar](100) NULL,

[ContactPhone] [nvarchar](20) NULL,

PRIMARY KEY CLUSTERED

(

[SponsorID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СтатистикиИгроков] Script Date: 13.05.2025 9:36:47 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СтатистикиИгроков](

[StatID] [int] IDENTITY(1,1) NOT NULL,

[MatchID] [int] NOT NULL,

[PlayerID] [int] NOT NULL,

[Kills] [int] NULL,

[Deaths] [int] NULL,

[Assists] [int] NULL,

[DamageDealt] [int] NULL,

[HealingDone] [int] NULL,

[OtherStats] [nvarchar](max) NULL,

PRIMARY KEY CLUSTERED

(

[StatID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[MatchID] ASC,

[PlayerID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] ADD DEFAULT ((0)) FOR [Kills]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] ADD DEFAULT ((0)) FOR [Deaths]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] ADD DEFAULT ((0)) FOR [Assists]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] ADD DEFAULT ((0)) FOR [DamageDealt]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] ADD DEFAULT ((0)) FOR [HealingDone]

GO

ALTER TABLE [dbo].[СтатистикиИгроков] WITH CHECK ADD FOREIGN KEY([MatchID])

REFERENCES [dbo].[Матчи] ([MatchID])

GO

ALTER TABLE [dbo].[СтатистикиИгроков] WITH CHECK ADD FOREIGN KEY([PlayerID])

REFERENCES [dbo].[Игроки] ([PlayerID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Трансляции] Script Date: 13.05.2025 9:36:51 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Трансляции](

[StreamID] [int] IDENTITY(1,1) NOT NULL,

[MatchID] [int] NOT NULL,

[StreamURL] [nvarchar](255) NOT NULL,

[StartTime] [datetime] NOT NULL,

[EndTime] [datetime] NULL,

[ViewerCount] [int] NULL,

[Language] [nvarchar](20) NULL,

PRIMARY KEY CLUSTERED

(

[StreamID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Трансляции] ADD DEFAULT ((0)) FOR [ViewerCount]

GO

ALTER TABLE [dbo].[Трансляции] WITH CHECK ADD FOREIGN KEY([MatchID])

REFERENCES [dbo].[Матчи] ([MatchID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Турниры] Script Date: 13.05.2025 9:36:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Турниры](

[TournamentID] [int] IDENTITY(1,1) NOT NULL,

[TournamentName] [nvarchar](100) NOT NULL,

[GameID] [int] NOT NULL,

[StartDate] [datetime] NOT NULL,

[EndDate] [datetime] NOT NULL,

[PrizePool] [decimal](15, 2) NULL,

[OrganizerID] [int] NOT NULL,

[RegistrationDeadline] [datetime] NULL,

[MaxTeams] [int] NULL,

[FormatDescription] [nvarchar](max) NULL,

[Rules] [nvarchar](max) NULL,

[IsCompleted] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[TournamentID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[Турниры] ADD DEFAULT ((0)) FOR [IsCompleted]

GO

ALTER TABLE [dbo].[Турниры] WITH CHECK ADD FOREIGN KEY([GameID])

REFERENCES [dbo].[Игры] ([GameID])

GO

ALTER TABLE [dbo].[Турниры] WITH CHECK ADD FOREIGN KEY([OrganizerID])

REFERENCES [dbo].[Пользователи] ([UserID])

GO

USE [КиберспортивныеМероприятия]

GO

/\*\*\*\*\*\* Object: Table [dbo].[УчастникиМатчей] Script Date: 13.05.2025 9:37:00 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[УчастникиМатчей](

[MatchParticipantID] [int] IDENTITY(1,1) NOT NULL,

[MatchID] [int] NOT NULL,

[TeamID] [int] NOT NULL,

[Score] [int] NULL,

[IsWinner] [bit] NULL,

PRIMARY KEY CLUSTERED

(

[MatchParticipantID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[MatchID] ASC,

[TeamID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[УчастникиМатчей] ADD DEFAULT ((0)) FOR [Score]

GO

ALTER TABLE [dbo].[УчастникиМатчей] ADD DEFAULT ((0)) FOR [IsWinner]

GO

ALTER TABLE [dbo].[УчастникиМатчей] WITH CHECK ADD FOREIGN KEY([MatchID])

REFERENCES [dbo].[Матчи] ([MatchID])

GO

ALTER TABLE [dbo].[УчастникиМатчей] WITH CHECK ADD FOREIGN KEY([TeamID])

REFERENCES [dbo].[Команды] ([TeamID])

GO

Скрипт Бд "Медицинская Лаборатория" и диаграмма Бд в sql server management studio(Рисунок 19)

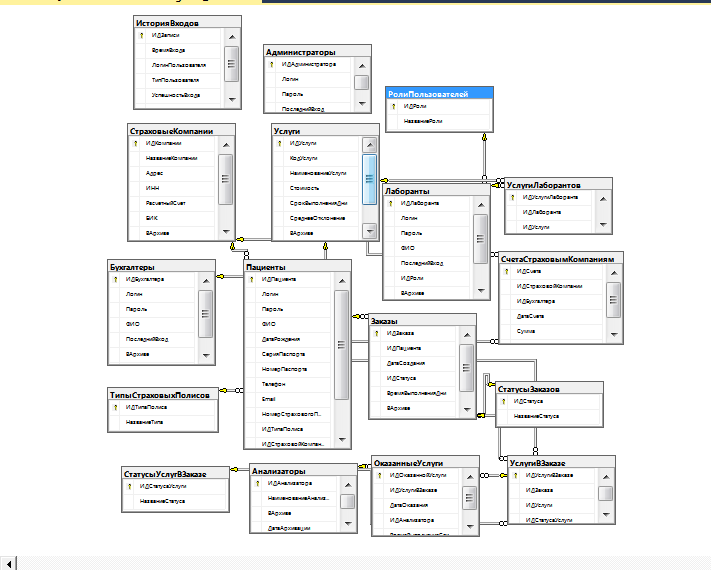


Рисунок 19-Диаграмма БД

USE [master]

GO

/\*\*\*\*\*\* Object: Database [ЛабораторнаяСистема] Script Date: 06.05.2025 10:37:43 \*\*\*\*\*\*/

CREATE DATABASE [ЛабораторнаяСистема]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'ЛабораторнаяСистема', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\ЛабораторнаяСистема.mdf' , SIZE = 4288KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'ЛабораторнаяСистема\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\ЛабораторнаяСистема\_log.ldf' , SIZE = 1072KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [ЛабораторнаяСистема].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ARITHABORT OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET AUTO\_CLOSE ON

GO

ALTER DATABASE [ЛабораторнаяСистема] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [ЛабораторнаяСистема] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [ЛабораторнаяСистема] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ENABLE\_BROKER

GO

ALTER DATABASE [ЛабораторнаяСистема] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [ЛабораторнаяСистема] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET RECOVERY SIMPLE

GO

ALTER DATABASE [ЛабораторнаяСистема] SET MULTI\_USER

GO

ALTER DATABASE [ЛабораторнаяСистема] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [ЛабораторнаяСистема] SET DB\_CHAINING OFF

GO

ALTER DATABASE [ЛабораторнаяСистема] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [ЛабораторнаяСистема] SET TARGET\_RECOVERY\_TIME = 0 SECONDS

GO

ALTER DATABASE [ЛабораторнаяСистема] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [ЛабораторнаяСистема] SET READ\_WRITE

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Администраторы] Script Date: 06.05.2025 10:37:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Администраторы](

[ИДАдминистратора] [int] IDENTITY(1,1) NOT NULL,

[Логин] [nvarchar](50) NOT NULL,

[Пароль] [nvarchar](100) NOT NULL,

[ПоследнийВход] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДАдминистратора] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Логин] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Анализаторы] Script Date: 06.05.2025 10:38:04 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Анализаторы](

[ИДАнализатора] [int] IDENTITY(1,1) NOT NULL,

[НаименованиеАнализатора] [nvarchar](100) NOT NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДАнализатора] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Анализаторы] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Бухгалтеры] Script Date: 06.05.2025 10:38:11 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Бухгалтеры](

[ИДБухгалтера] [int] IDENTITY(1,1) NOT NULL,

[Логин] [nvarchar](50) NOT NULL,

[Пароль] [nvarchar](100) NOT NULL,

[ФИО] [nvarchar](100) NOT NULL,

[ПоследнийВход] [datetime] NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДБухгалтера] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Логин] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Бухгалтеры] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Заказы] Script Date: 06.05.2025 10:38:16 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Заказы](

[ИДЗаказа] [int] IDENTITY(1,1) NOT NULL,

[ИДПациента] [int] NOT NULL,

[ДатаСоздания] [datetime] NOT NULL,

[ИДСтатуса] [int] NOT NULL,

[ВремяВыполненияДни] [int] NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДЗаказа] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Заказы] ADD DEFAULT (getdate()) FOR [ДатаСоздания]

GO

ALTER TABLE [dbo].[Заказы] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

ALTER TABLE [dbo].[Заказы] WITH CHECK ADD CONSTRAINT [FK\_Заказы\_Пациенты] FOREIGN KEY([ИДПациента])

REFERENCES [dbo].[Пациенты] ([ИДПациента])

GO

ALTER TABLE [dbo].[Заказы] CHECK CONSTRAINT [FK\_Заказы\_Пациенты]

GO

ALTER TABLE [dbo].[Заказы] WITH CHECK ADD CONSTRAINT [FK\_Заказы\_Статусы] FOREIGN KEY([ИДСтатуса])

REFERENCES [dbo].[СтатусыЗаказов] ([ИДСтатуса])

GO

ALTER TABLE [dbo].[Заказы] CHECK CONSTRAINT [FK\_Заказы\_Статусы]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ИсторияВходов] Script Date: 06.05.2025 10:38:21 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ИсторияВходов](

[ИДЗаписи] [int] IDENTITY(1,1) NOT NULL,

[ВремяВхода] [datetime] NOT NULL,

[ЛогинПользователя] [nvarchar](50) NOT NULL,

[ТипПользователя] [nvarchar](20) NOT NULL,

[УспешностьВхода] [bit] NOT NULL,

[IPАдрес] [nvarchar](15) NULL,

PRIMARY KEY CLUSTERED

(

[ИДЗаписи] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[ИсторияВходов] ADD DEFAULT (getdate()) FOR [ВремяВхода]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Лаборанты] Script Date: 06.05.2025 10:38:27 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Лаборанты](

[ИДЛаборанта] [int] IDENTITY(1,1) NOT NULL,

[Логин] [nvarchar](50) NOT NULL,

[Пароль] [nvarchar](100) NOT NULL,

[ФИО] [nvarchar](100) NOT NULL,

[ПоследнийВход] [datetime] NULL,

[ИДРоли] [int] NOT NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДЛаборанта] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Логин] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Лаборанты] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

ALTER TABLE [dbo].[Лаборанты] WITH CHECK ADD CONSTRAINT [FK\_Лаборанты\_Роли] FOREIGN KEY([ИДРоли])

REFERENCES [dbo].[РолиПользователей] ([ИДРоли])

GO

ALTER TABLE [dbo].[Лаборанты] CHECK CONSTRAINT [FK\_Лаборанты\_Роли]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ОказанныеУслуги] Script Date: 06.05.2025 10:38:35 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ОказанныеУслуги](

[ИДОказаннойУслуги] [int] IDENTITY(1,1) NOT NULL,

[ИДУслугиВЗаказе] [int] NOT NULL,

[ДатаОказания] [datetime] NOT NULL,

[ИДАнализатора] [int] NOT NULL,

[ВремяВыполненияСекунды] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДОказаннойУслуги] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[ОказанныеУслуги] ADD DEFAULT (getdate()) FOR [ДатаОказания]

GO

ALTER TABLE [dbo].[ОказанныеУслуги] WITH CHECK ADD CONSTRAINT [FK\_ОказанныеУслуги\_Анализаторы] FOREIGN KEY([ИДАнализатора])

REFERENCES [dbo].[Анализаторы] ([ИДАнализатора])

GO

ALTER TABLE [dbo].[ОказанныеУслуги] CHECK CONSTRAINT [FK\_ОказанныеУслуги\_Анализаторы]

GO

ALTER TABLE [dbo].[ОказанныеУслуги] WITH CHECK ADD CONSTRAINT [FK\_ОказанныеУслуги\_УслугиВЗаказе] FOREIGN KEY([ИДУслугиВЗаказе])

REFERENCES [dbo].[УслугиВЗаказе] ([ИДУслугиВЗаказе])

GO

ALTER TABLE [dbo].[ОказанныеУслуги] CHECK CONSTRAINT [FK\_ОказанныеУслуги\_УслугиВЗаказе]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Пациенты] Script Date: 06.05.2025 10:38:42 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Пациенты](

[ИДПациента] [int] IDENTITY(1,1) NOT NULL,

[Логин] [nvarchar](50) NOT NULL,

[Пароль] [nvarchar](100) NOT NULL,

[ФИО] [nvarchar](100) NOT NULL,

[ДатаРождения] [date] NOT NULL,

[СерияПаспорта] [nvarchar](4) NOT NULL,

[НомерПаспорта] [nvarchar](6) NOT NULL,

[Телефон] [nvarchar](15) NOT NULL,

[Email] [nvarchar](100) NULL,

[НомерСтраховогоПолиса] [nvarchar](20) NOT NULL,

[ИДТипаПолиса] [int] NOT NULL,

[ИДСтраховойКомпании] [int] NOT NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДПациента] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[Логин] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

CONSTRAINT [UQ\_Паспорт] UNIQUE NONCLUSTERED

(

[СерияПаспорта] ASC,

[НомерПаспорта] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

CONSTRAINT [UQ\_СтраховойПолис] UNIQUE NONCLUSTERED

(

[НомерСтраховогоПолиса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Пациенты] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

ALTER TABLE [dbo].[Пациенты] WITH CHECK ADD CONSTRAINT [FK\_Пациенты\_СтраховыеКомпании] FOREIGN KEY([ИДСтраховойКомпании])

REFERENCES [dbo].[СтраховыеКомпании] ([ИДКомпании])

GO

ALTER TABLE [dbo].[Пациенты] CHECK CONSTRAINT [FK\_Пациенты\_СтраховыеКомпании]

GO

ALTER TABLE [dbo].[Пациенты] WITH CHECK ADD CONSTRAINT [FK\_Пациенты\_ТипыПолисов] FOREIGN KEY([ИДТипаПолиса])

REFERENCES [dbo].[ТипыСтраховыхПолисов] ([ИДТипаПолиса])

GO

ALTER TABLE [dbo].[Пациенты] CHECK CONSTRAINT [FK\_Пациенты\_ТипыПолисов]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[РолиПользователей] Script Date: 06.05.2025 10:38:48 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[РолиПользователей](

[ИДРоли] [int] IDENTITY(1,1) NOT NULL,

[НазваниеРоли] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДРоли] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[НазваниеРоли] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СтатусыЗаказов] Script Date: 06.05.2025 10:38:53 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СтатусыЗаказов](

[ИДСтатуса] [int] IDENTITY(1,1) NOT NULL,

[НазваниеСтатуса] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДСтатуса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[НазваниеСтатуса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СтатусыЗаказов] Script Date: 06.05.2025 10:38:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СтатусыЗаказов](

[ИДСтатуса] [int] IDENTITY(1,1) NOT NULL,

[НазваниеСтатуса] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДСтатуса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[НазваниеСтатуса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СтатусыУслугВЗаказе] Script Date: 06.05.2025 10:39:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СтатусыУслугВЗаказе](

[ИДСтатусаУслуги] [int] IDENTITY(1,1) NOT NULL,

[НазваниеСтатуса] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДСтатусаУслуги] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[НазваниеСтатуса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СтраховыеКомпании] Script Date: 06.05.2025 10:39:14 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СтраховыеКомпании](

[ИДКомпании] [int] IDENTITY(1,1) NOT NULL,

[НазваниеКомпании] [nvarchar](100) NOT NULL,

[Адрес] [nvarchar](200) NOT NULL,

[ИНН] [nvarchar](12) NOT NULL,

[РасчетныйСчет] [nvarchar](20) NOT NULL,

[БИК] [nvarchar](9) NOT NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДКомпании] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[ИНН] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[СтраховыеКомпании] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[СчетаСтраховымКомпаниям] Script Date: 06.05.2025 10:39:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[СчетаСтраховымКомпаниям](

[ИДСчета] [int] IDENTITY(1,1) NOT NULL,

[ИДСтраховойКомпании] [int] NOT NULL,

[ИДБухгалтера] [int] NOT NULL,

[ДатаСчета] [datetime] NOT NULL,

[Сумма] [decimal](15, 2) NOT NULL,

[Оплачен] [bit] NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДСчета] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] ADD DEFAULT (getdate()) FOR [ДатаСчета]

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] ADD DEFAULT ((0)) FOR [Оплачен]

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] WITH CHECK ADD CONSTRAINT [FK\_Счета\_Бухгалтеры] FOREIGN KEY([ИДБухгалтера])

REFERENCES [dbo].[Бухгалтеры] ([ИДБухгалтера])

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] CHECK CONSTRAINT [FK\_Счета\_Бухгалтеры]

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] WITH CHECK ADD CONSTRAINT [FK\_Счета\_СтраховыеКомпании] FOREIGN KEY([ИДСтраховойКомпании])

REFERENCES [dbo].[СтраховыеКомпании] ([ИДКомпании])

GO

ALTER TABLE [dbo].[СчетаСтраховымКомпаниям] CHECK CONSTRAINT [FK\_Счета\_СтраховыеКомпании]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ТипыСтраховыхПолисов] Script Date: 06.05.2025 10:39:24 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ТипыСтраховыхПолисов](

[ИДТипаПолиса] [int] IDENTITY(1,1) NOT NULL,

[НазваниеТипа] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДТипаПолиса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[НазваниеТипа] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Услуги] Script Date: 06.05.2025 10:39:31 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Услуги](

[ИДУслуги] [int] IDENTITY(1,1) NOT NULL,

[КодУслуги] [nvarchar](20) NOT NULL,

[НаименованиеУслуги] [nvarchar](100) NOT NULL,

[Стоимость] [decimal](10, 2) NOT NULL,

[СрокВыполненияДни] [int] NOT NULL,

[СреднееОтклонение] [decimal](10, 2) NULL,

[ВАрхиве] [bit] NOT NULL,

[ДатаАрхивации] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[ИДУслуги] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

UNIQUE NONCLUSTERED

(

[КодУслуги] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Услуги] ADD DEFAULT ((0)) FOR [ВАрхиве]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[УслугиВЗаказе] Script Date: 06.05.2025 10:39:38 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[УслугиВЗаказе](

[ИДУслугиВЗаказе] [int] IDENTITY(1,1) NOT NULL,

[ИДЗаказа] [int] NOT NULL,

[ИДУслуги] [int] NOT NULL,

[ИДСтатусаУслуги] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДУслугиВЗаказе] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[УслугиВЗаказе] WITH CHECK ADD CONSTRAINT [FK\_УслугиВЗаказе\_Заказы] FOREIGN KEY([ИДЗаказа])

REFERENCES [dbo].[Заказы] ([ИДЗаказа])

GO

ALTER TABLE [dbo].[УслугиВЗаказе] CHECK CONSTRAINT [FK\_УслугиВЗаказе\_Заказы]

GO

ALTER TABLE [dbo].[УслугиВЗаказе] WITH CHECK ADD CONSTRAINT [FK\_УслугиВЗаказе\_Статусы] FOREIGN KEY([ИДСтатусаУслуги])

REFERENCES [dbo].[СтатусыУслугВЗаказе] ([ИДСтатусаУслуги])

GO

ALTER TABLE [dbo].[УслугиВЗаказе] CHECK CONSTRAINT [FK\_УслугиВЗаказе\_Статусы]

GO

ALTER TABLE [dbo].[УслугиВЗаказе] WITH CHECK ADD CONSTRAINT [FK\_УслугиВЗаказе\_Услуги] FOREIGN KEY([ИДУслуги])

REFERENCES [dbo].[Услуги] ([ИДУслуги])

GO

ALTER TABLE [dbo].[УслугиВЗаказе] CHECK CONSTRAINT [FK\_УслугиВЗаказе\_Услуги]

GO

USE [ЛабораторнаяСистема]

GO

/\*\*\*\*\*\* Object: Table [dbo].[УслугиЛаборантов] Script Date: 06.05.2025 10:39:43 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[УслугиЛаборантов](

[ИДУслугиЛаборанта] [int] IDENTITY(1,1) NOT NULL,

[ИДЛаборанта] [int] NOT NULL,

[ИДУслуги] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[ИДУслугиЛаборанта] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY],

CONSTRAINT [UQ\_ЛаборантУслуга] UNIQUE NONCLUSTERED

(

[ИДЛаборанта] ASC,

[ИДУслуги] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[УслугиЛаборантов] WITH CHECK ADD CONSTRAINT [FK\_УслугиЛаборантов\_Лаборанты] FOREIGN KEY([ИДЛаборанта])

REFERENCES [dbo].[Лаборанты] ([ИДЛаборанта])

GO

ALTER TABLE [dbo].[УслугиЛаборантов] CHECK CONSTRAINT [FK\_УслугиЛаборантов\_Лаборанты]

GO

ALTER TABLE [dbo].[УслугиЛаборантов] WITH CHECK ADD CONSTRAINT [FK\_УслугиЛаборантов\_Услуги] FOREIGN KEY([ИДУслуги])

REFERENCES [dbo].[Услуги] ([ИДУслуги])

GO

ALTER TABLE [dbo].[УслугиЛаборантов] CHECK CONSTRAINT [FK\_УслугиЛаборантов\_Услуги]

GO