**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КОМИ**

**Государственное профессиональное образовательное учреждение**

**«Воркутинский арктический горно-политехнический колледж»**

**ДИПЛОМНАЯ РАБОТА**

По дисциплине МДК.07.01 Управление и автоматизация баз данных

**Разработка и администрирование базы данных «ГАИ»**

Выполнил студент гр. ИСП-21 /\_\_\_\_\_\_\_/Коромыслов Ярослав Николаевич/

(подпись) (Ф.И.О.)

**ОЦЕНКА:\_\_\_\_\_\_\_\_\_\_\_\_\_**

Дата:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ПРОВЕРИЛ**

Научный руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_/ Егоров Данил Павлович/

(подпись) (Ф.И.О.)

Воркута

2025

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc198028504)

[ГЛАВА 1. ПРЕДПРОЕКТНАЯ ЧАСТЬ ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ СИСТЕМЕ 4](#_Toc198028505)

[1.1 Анализ предметной области 4](#_Toc198028506)

[1.2 Разработка технического задания 5](#_Toc198028507)

[1.3 Моделирование бизнес-процессов (IDEF0 AS IS / TO BE) 8](#_Toc198028508)

[ГЛАВА 2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ И ВЫБОР ИНСТРУМЕНТАРИЯ 12](#_Toc198028509)

[2.1 Разработка диаграммы ERD 12](#_Toc198028510)

[2.2 Словарь данных 15](#_Toc198028511)

[2.3 СУБД SQL Server 16](#_Toc198028512)

[2.4 Microsoft SQL Server Management Studio 18](#_Toc198028513)

[ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ 21](#_Toc198028514)

[3.1 Разработка базы данных 21](#_Toc198028515)

[3.2 Введение в безопасность SQL Server 29](#_Toc198028516)

[3.3 Управление безопасностью уровня сервера 30](#_Toc198028517)

[3.4 Управление участниками уровня базы данных 31](#_Toc198028518)

[3.5 Управление разрешениями уровня базы данных 32](#_Toc198028519)

[3.6 Резервное копирование баз данных 33](#_Toc198028520)

[3.7 Шифрование данных баз данных 34](#_Toc198028521)

[ЗАКЛЮЧЕНИЕ 36](#_Toc198028522)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕАРТУРЫ 37](#_Toc198028523)

# ВВЕДЕНИЕ

Безопасность дорожного движения – одна из важнейших составляющих общественной безопасности и ключевой фактор социального благополучия. В условиях быстрорастущего потока транспортных средств и увеличения количества дорожных происшествий, обеспечение порядка на дорогах стало задачей первостепенной важности. Государственная автомобильная инспекция (ГАИ) представляет собой специализированный орган, ответственный за регулирование и контроль за соблюдением правил дорожного движения, а также за профилактику дорожно-транспортных происшествий.

Бизнес-процесс — это совокупность взаимосвязанных действий и мероприятий, направленных на создание продуктов или услуг, которые обеспечивают удовлетворение потребностей клиентов и поддерживают устойчивое функционирование организации. В контексте дипломной работы, посвященной ГАИ, можно рассмотреть бизнес-процессы, связанные с обеспечением безопасности на дорожном движении, администрированием транспортных средств, проведением проверок на выявление нарушений и обучением населения правилам дорожного движения.

**Объект:** база данных «ГАИ».

Предмет: регистрация автотранспортного средства «ГАИ»

**Цель работы –** разработать базу данных для «ГАИ».

**Задачи:**

* проанализировать предметную область;
* выбрать инструментарий;
* спроектировать базу данных;
* выполнить шифрование данных;
* выполнить восстановление базы данных;
* выполнить разграничение прав доступа базы данных.

# ГЛАВА 1. ПРЕДПРОЕКТНАЯ ЧАСТЬ ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ СИСТЕМЕ

# 1.1 Анализ предметной области

Современный мир информационных технологий трудно представить себе без возможности обработки больших объёмов информации. Такие объёмы информации удобно обрабатывать с помощью баз данных. Практически все системы в той или иной степени связаны с долговременным хранением и обработкой информации. Фактически, информация становится фактором, определяющим эффективность любой сферы деятельности. Увеличились информационные потоки и повысились требования к скорости обработки данных. Большинство операций не может быть выполнено вручную. Любые административные решения требуют более чёткой и точной оценки текущей ситуации и возможных перспектив ёё изменения.

Ключевые функции и задачи ГАИ

1. **Регистрация автотранспортных средств**: ГАИ занимается регистрацией автомобилей, включая выдачу номерных знаков и внесение информации о транспортных средствах в базу данных.
2. **Безопасность дорожного движения**: ГАИ следит за соблюдением правил дорожного движения, выявляет нарушителей и применяет меры взыскания в соответствии с Кодексом об административных правонарушениях.
3. **Учет и анализ ДТП**: ГАИ регистрирует и анализирует дорожно-транспортные происшествия для улучшения безопасности на дорогах и планирования местоположения патрульных постов.
4. **Борьба с угоном транспортных средств**: ГАИ ведет статистику угонов и участвует в оперативном поиске угнанных машин.

Требования к информационным системам

1. **Хранение и обработка данных**: Системы должны обеспечивать хранение и быстрый доступ к информации о водителях, автомобилях, нарушениях и штрафах.
2. **Поиск и сортировка данных**: Возможность поиска и сортировки данных по различным критериям для эффективного анализа и принятия решений.
3. **Формирование отчетов**: Автоматизированное формирование отчетов о штрафах, страховках и других статистических данных

# 1.2 Разработка технического задания

**Назначение разработки**

Основная цель создания базы данных для ГАИ заключается в разработке системы, которая позволит эффективно хранить, управлять и обрабатывать информацию, связанную с деятельностью ГАИ. Это включает в себя данные о транспортных средствах, владельцах, нарушениях правил дорожного движения и регистрации.

В качестве пользователей данной базы данных выступают: администратор и работник ГАИ.

Инспекторы дорожно-патрульной службы (ДПС), которые выписывают штрафы и составляют протоколы о нарушениях правил дорожного движения

Сотрудники, занимающиеся регистрацией транспортных средств и выдачей водительских удостоверений

Аналитики и статистики ГАИ, использующие базу данных для составления отчетов и анализа ситуации на дорогах

Сотрудники информационных центров ГАИ, обеспечивающие техническую поддержку и обновление базы данных

Сотрудники ППС занимаются контролем соблюдения правил дорожного движения и предотвращением правонарушений на дорогах.

Водитель лицо, управляющее транспортным средством и несущие ответственность за соблюдение правил дорожного движения.

**Требования к программе**

**3.1. Требования к функциональным характеристикам**

База данных «ГАИ» должна выполнять следующие функции:

* Обеспечение безопасности данных**:** Контроль доступа к информации в зависимости от уровня полномочий пользователей системы для защиты конфиденциальной информации
* Разграничение доступа.
* Управление данными**:** Пользователи должны иметь возможность изменять, добавлять и удалять данные.

**Шифрование данных**

Авторизация**:** При входе в систему логин и пароль должны шифроваться и сравниваться с зашифрованными данными в базе.

Регистрация**:** Логин и пароль при регистрации также должны шифроваться перед добавлением в базу данных.

**Восстановление данных**

База данных должна поддерживать возможность восстановления. В случае сбоя необходимо иметь файл восстановления, позволяющий восстановить данные через запрос к базе.

Роли и права пользователей

В базе данных должны быть четко определены роли пользователей:

* **Инспекторы ДПС**: имеют доступ для внесения данных о нарушениях и штрафах.
* **Сотрудники регистрационных отделов**: могут вносить и редактировать информацию о транспортных средствах и водительских удостоверениях.
* **Аналитики и статистики**: имеют доступ к обобщенным данным для составления отчетов.
* **Технический персонал**: обеспечивает поддержку и обновление базы данных.
* **Сотрудники ППС**: занимаются контролем соблюдения правил дорожного движения и предотвращением правонарушений на дорогах.
* **Водитель**: это человек, который управляет транспортным средством и несет полную ответственность
* **Автомобиль**: средство передвижения

**3.2. Требования к надежности**

* парольная защитапри запуске программы;
* разграничение прав доступа;
* шифрование данных

**3.3. Требования к составу и параметрам технических средств**

Системные требования для работы программного продукта должны быть следующими: тактовая частота процессора -1 600 Гц; объем оперативной памяти 4 Гб; объем свободного дискового пространства 150 Мб; разрешение монитора 1 024 х 768; наличие устройства чтения USB-носители.

**3.4. Требования к информационной и программной совместимости**

Программа должна работать в операционных системах Windows 10/11. Все формируемые отчеты должны иметь возможность экспортирования в редактор электронных таблиц MS Office Excel 2016/2024.

**3.5. Требования к транспортированию и хранению**

Программа поставляется на USB-носителе информации.

Программная документация поставляется в электронном и печатном виде.

**3.6. Специальные требования**

Программное обеспечение должно иметь дружественный интерфейс, рассчитанный на пользователя средней квалификации (с точки зрения компьютерной грамотности).

Ввиду объемности проекта задачи предполагается решать поэтапно. При этом модули программного обеспечения (ПО), созданные в разное время, должны предполагать возможность наращивания системы и быть совместимы друг с другом; поэтому документация на принятое эксплуатационное ПО должна содержать полную информацию, необходимую для работы с ним программистов. Язык программирования определяется выбором исполнителя, при этом он должен обеспечивать возможность интеграции программного обеспечения с пакетом MS Office 2016/2024.

**4. Требования к программной документации**

В ходе разработки программы для базы данных ГИБДД должны быть подготовлены: текст программы и описание программы. Очевидно, что использование данной программы значительно сократит время, затрачиваемое на добавление, удаление и редактирование данных. Кроме того, в программе можно получить всю необходимую информацию о транспортных средствах, владельцах, нарушениях правил дорожного движения и пользователях системы. Для получения этих сведений вручную может потребоваться не менее 10-12 минут. С использованием программы затраты времени сокращаются до 1-2 минут.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Этап** | **Дата начала** | **Дата окончания** |
| 1. | Выдача технического задания | 25.11.2024 | 27.11.2024 |
| 2. | Разработка технического задания | 28.11.2024 | 30.11.2024 |
| 3. | Выбор инструментария | 08.12.2024 | 11.12.2024 |
| 4. | Разработка базы данных проекта | 14.12.2024 | 17.12.2024 |
| 5. | Защита курсового проекта | 21.01.2025 | 21.01.2025 |

# 1.3 Моделирование бизнес-процессов (IDEF0 AS IS / TO BE)

IDEF0 (Integration Definition for Function Modeling) — это методология функционального моделирования и графическая нотация, предназначенная для описания, анализа и проектирования бизнес-процессов, систем или организационных решений.

**AS IS** (с англ. «как есть») - это текущее состояние системы, процесса или организации. Описывает, как процессы, структуры данных, бизнес-правила и технологии функционируют в настоящий момент. Цель - понять и зафиксировать существующую ситуацию без изменений.

**TO BE** (с англ. «как должно быть») - это желаемое, целевое состояние системы или процесса после внедрения изменений, улучшений или новой системы. Отражает, как процессы и структура данных должны работать в будущем для повышения эффективности, качества или соответствия новым требованиям.

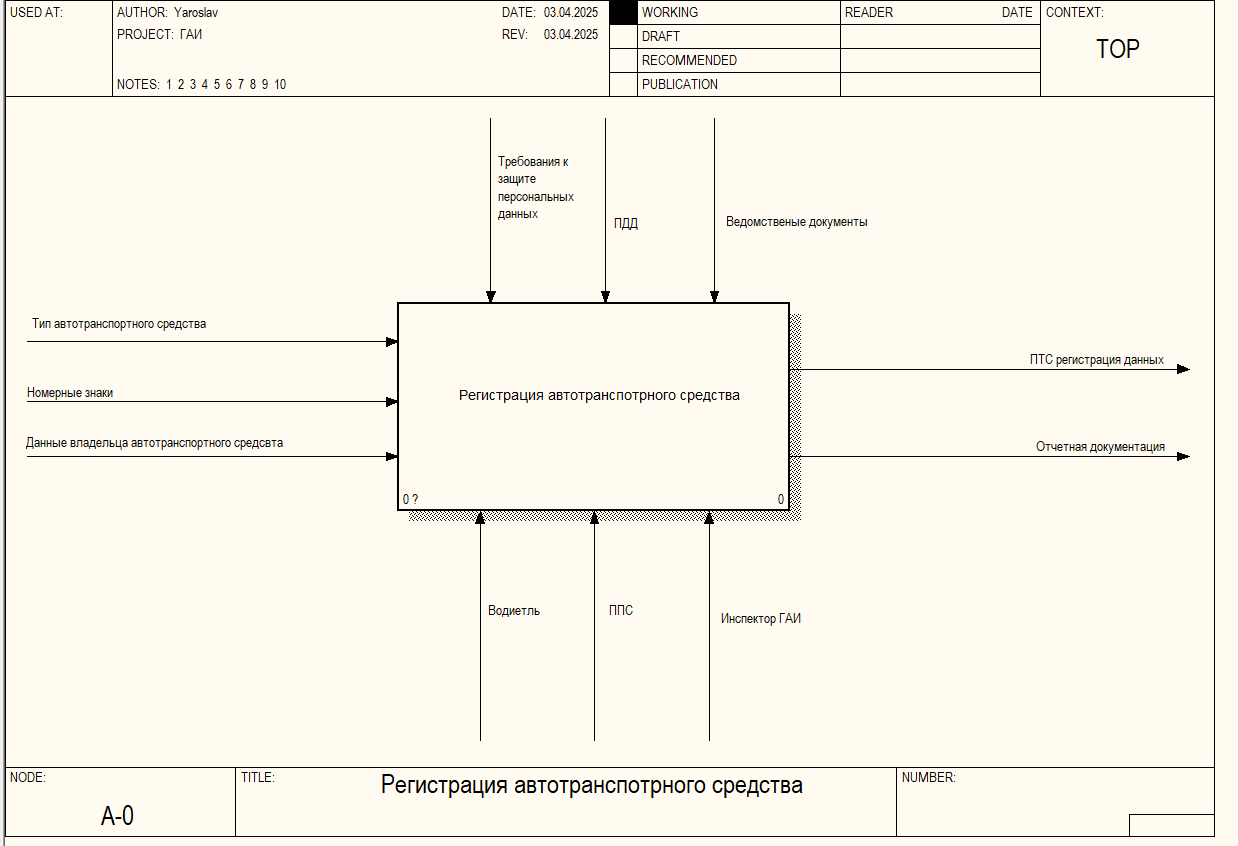
**Основные характеристики**

1. **Цель**: Моделирование сложных систем, включающих людей, машины, ресурсы, информационные системы и потоки данных. IDEF0 помогает анализировать процессы, уточнять требования и проектировать системы на уровне функций и задач.
2. **Графический язык**: Используется для визуального представления процессов в виде блоков (функциональных элементов), соединенных стрелками, которые обозначают входы, выходы, управление и механизмы.
3. **Структура модели**:

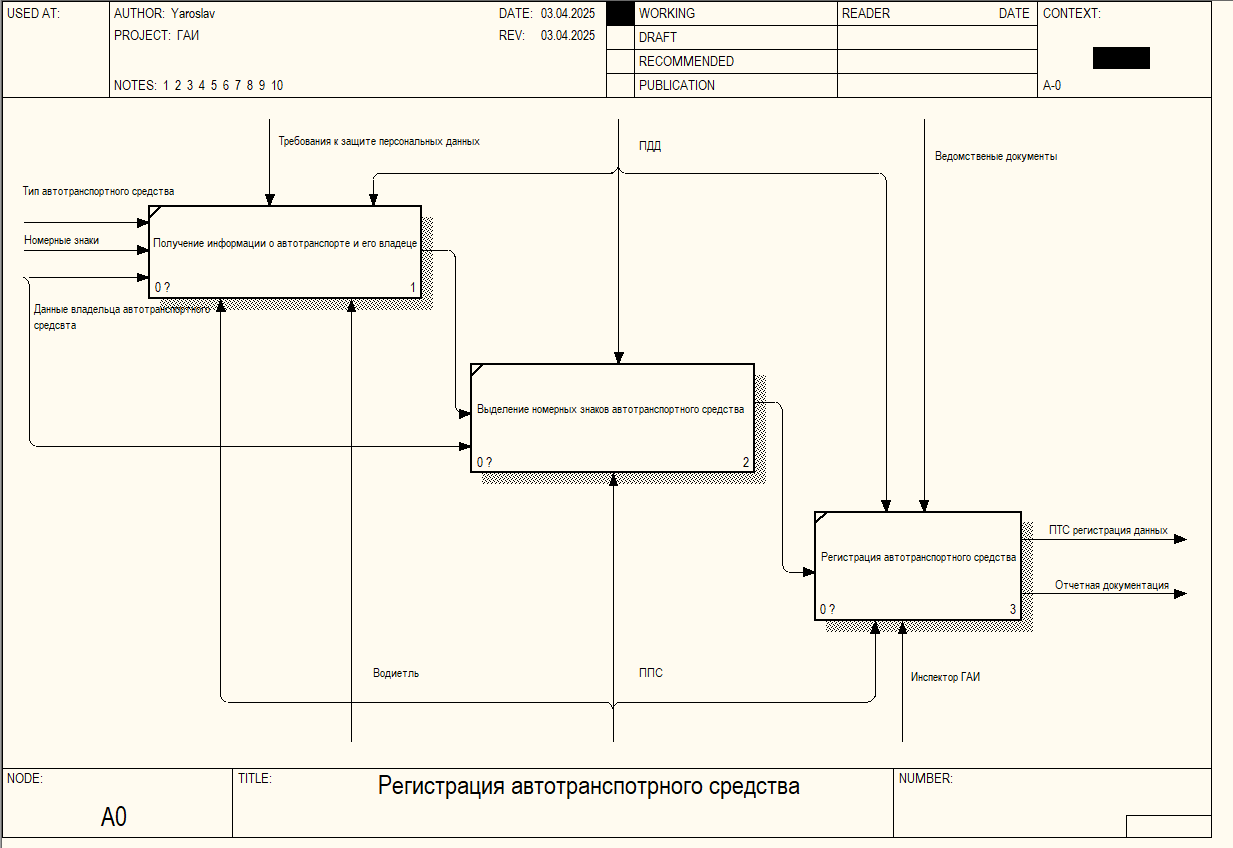
**Функциональный блок**: Представляет процесс или функцию.

**Стрелки**:

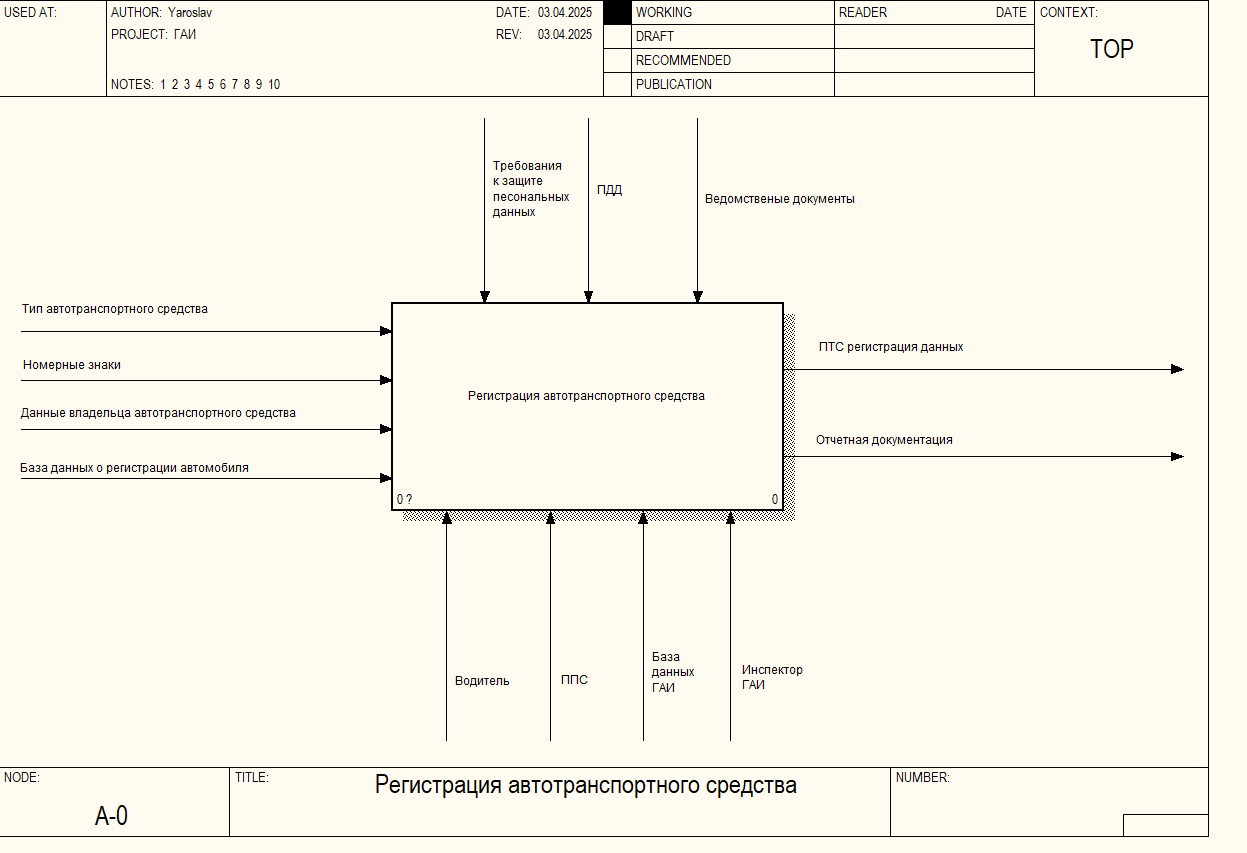
* Входы (Input) — данные или материалы, необходимые для выполнения функции.
* Управление (Control) — правила или ограничения, определяющие выполнение функции.
* Выходы (Output) — результаты выполнения функции.
* Механизмы (Mechanism) — ресурсы или инструменты, используемые для выполнения функции.



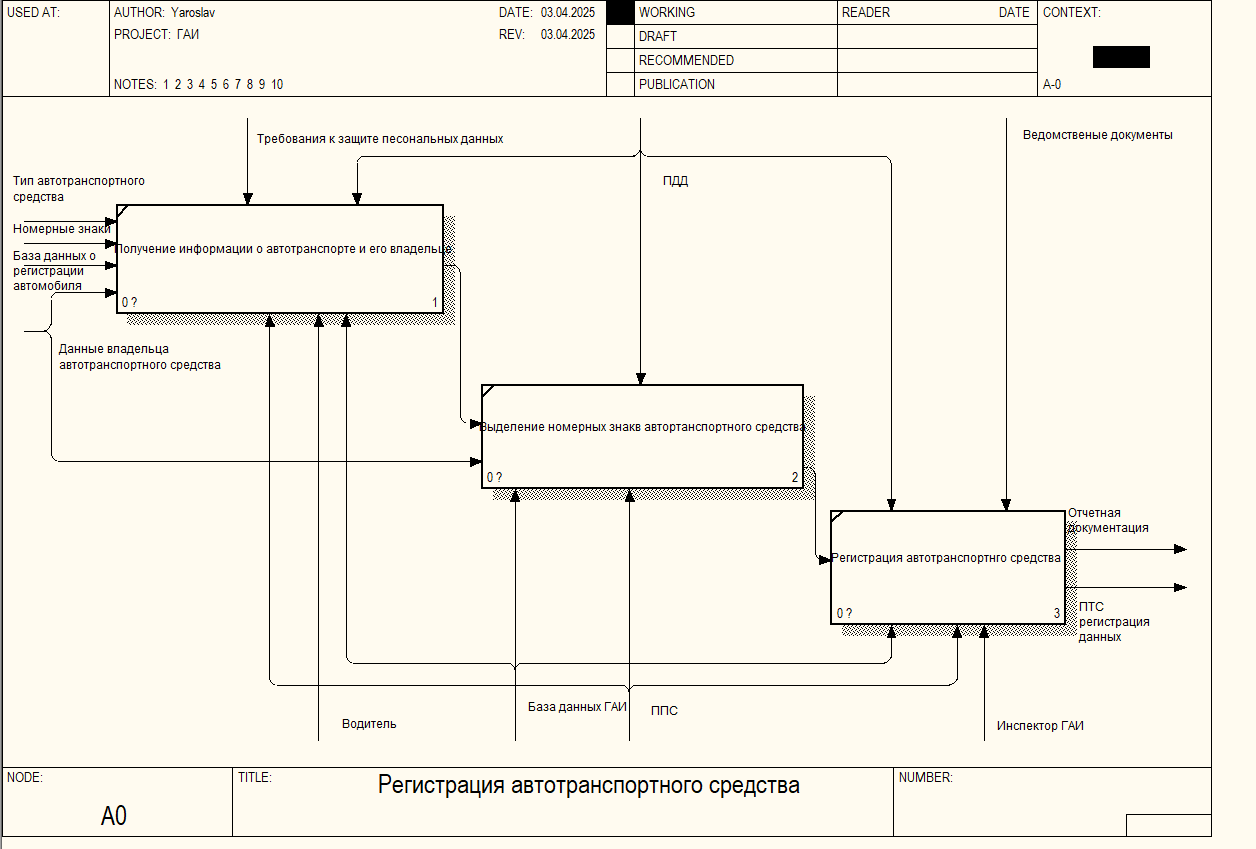
*Рис 1.1 IDEF0(AS IS)*



*Рис 1.2 IDEF3(AS IS)*



*Рис 1.3IDEF0(TO BE)*



*Рис 1.4 IDEF3(TO BE)*

# ГЛАВА 2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ И ВЫБОР ИНСТРУМЕНТАРИЯ

# 2.1 Разработка диаграммы ERD

Диаграммы «сущность-связь» (ER-диаграммы) представляют собой стандартный инструмент для описания данных и их взаимосвязей в рамках определенной предметной области. Они включают в себя сущности и связи, которые отражают ключевые бизнес-правила. В отличие от функциональных диаграмм, которые акцентируют внимание на процессах, ER-диаграммы сосредоточены на структурировании данных.

Основные элементы ER-модели включают сущности, атрибуты и связи.

Сущность представляет собой класс объектов, которые могут быть реальными или абстрактными (например, люди, события или предметы), и информация о которых важна для анализа данных. Структура данных описывает правила и ограничения, которые определяют связи между различными элементами данных.

Каждая сущность должна обладать следующими характеристиками:

* **Уникальное имя** — для четкой идентификации сущности.
* **Один или несколько атрибутов**, которые принадлежат этой сущности или могут быть унаследованы через связи с другими сущностями.
* **Один или несколько атрибутов**, которые позволяют однозначно идентифицировать каждый экземпляр данной сущности.

**Атрибут** — это любая характеристика сущности, которая имеет значение в контексте рассматриваемой области и используется для квалификации, идентификации, классификации, количественной оценки или описания состояния сущности. Таким образом, атрибут представляет собой определенный тип характеристик или свойств, связанных с множеством реальных или абстрактных объектов. Экземпляр атрибута — это конкретная характеристика определенного экземпляра сущности. Атрибуты могут быть разделены на ключевые, то есть те, которые входят в состав уникального идентификатора ключа, и описательные - прочие.

Первичный ключ — это атрибут или комбинация атрибутов, а также связей, которые служат для уникальной идентификации каждого экземпляра сущности. Он представляет собой набор признаков, позволяющих отличить один объект от другого.

Описательные атрибуты могут быть обязательными или необязательными.

Обязательные атрибуты всегда имеют определенное значение для каждой сущности, тогда как необязательные могут не иметь значения. Обязательные описательные атрибуты помечаются символом «\*», а необязательные — символом «о».

Связь — это отношение между одной сущностью и другой, или же между экземплярами одной и той же сущности. Каждая связь может иметь одну из двух модальностей. Если хотя бы один экземпляр одной сущности связан с хотя бы одним экземпляром другой сущности, то такая связь считается обязательной. Необязательная связь, в свою очередь, представляет собой условное отношение между сущностями.

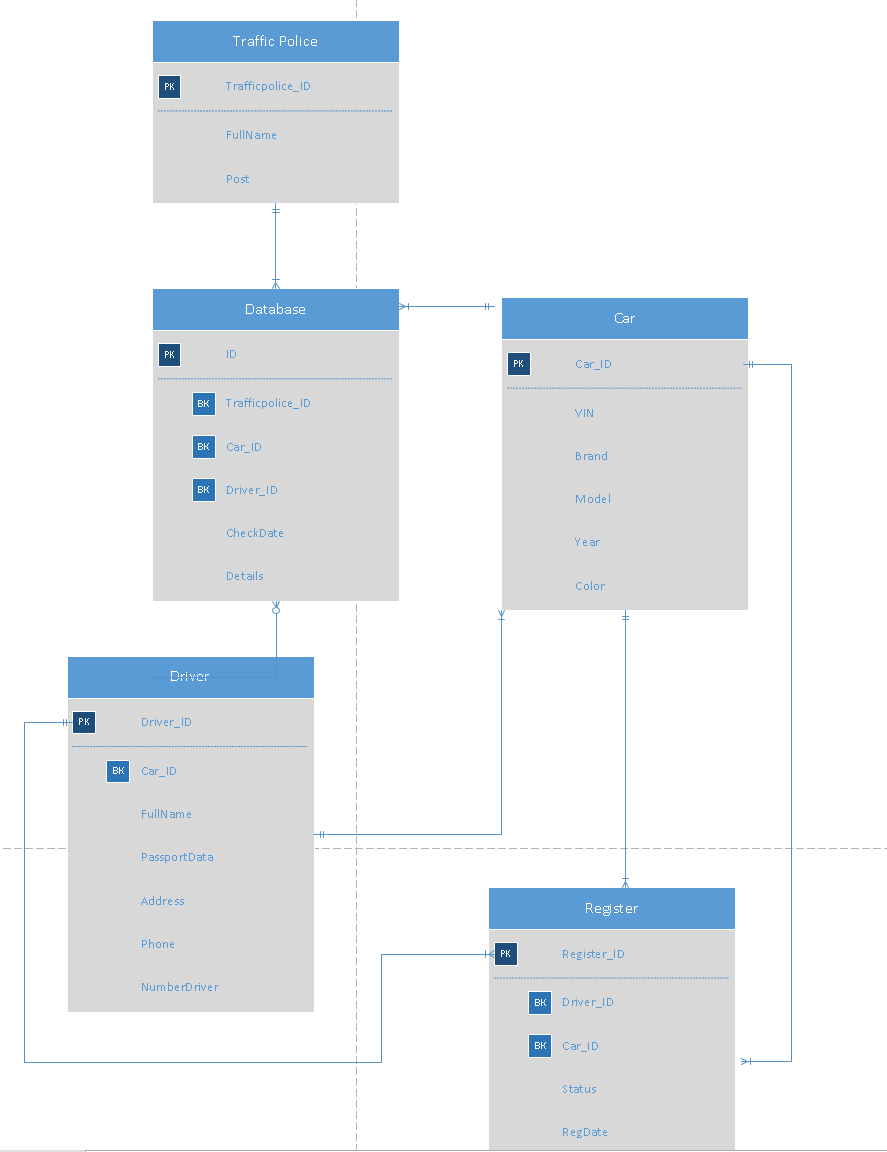


Рис 2.1 Диаграмма ERD (Entity Relationship Diagram)!!!!!

# 2.2 Словарь данных

Словарь данных – это структура данных, которая представляет собой набор пар ключ-значение. Каждый ключ уникален в словаре данных, и ему соответствует определенное значение. Словари данных используются для хранения и организации информации, а также для быстрого доступа к данным по ключу. Они широко применяются в программировании для решения различных задач, таких как хранение конфигурационных данных, управление ресурсами и т.д. Словари данных также удобны для хранения и передачи структурированных данных между различными частями программы.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Словарь данных | | | | |
| **Driver** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **Driver\_ID** | INTEGER | Y | Первичный ключ |
| BK | **Car\_ID** | INTEGER | Y | Ссылка на сущность Car |
|  | **Fullname** | VARCHAR(255) | Y | Полное имя водителя |
|  | **Аdres** | VARCHAR(255) | Y | Полный адрес проживания |
|  | **Phone** | VARCHAR(15) | Y | Контактный телефон |
|  | **NumberDrive** | VARCHAR(20) | Y | Уникальный номер водительского удостоверения |
|  | **PassportData** | VARCHAR(20) | Y | Серия и номер паспорта |

*Таблица 1. Словарь данных*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Car** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **Car\_ID** | INTEGER | Y | Первичный ключ |
|  | **VIN** | VARCHAR(17) | Y | Уникальный идентификатор автомобиля |
|  | **Brand** | VARCHAR(50) | Y | Марка автомобиля |
|  | **Model** | VARCHAR(50) | N | Модель автомобиля |
|  | **Year** | INTEGER | Y | Год выпуска |
|  | **Color** | VARCHAR(30) | N | Цвет автомобиля |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Register** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **Register\_ID** | INTEGER | Y | Первичный ключ |
| BK | **Driver\_ID** | INTEGER | Y | Ссылка на сущность Driver |
| BK | **Car\_ID** | INTEGER | Y | Ссылка на сущность Car |
|  | **Status** | FLOAT | Y | Статус |
|  | **RegDate** | VARCHAR(100) | Y | ДатаРегистрации |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Database** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | INTEGER | Y | Первичный ключ |
| BK | **Trafficpolice\_ID** | INTEGER | Y | Ссылка на сущность Trafficpolice |
| BK | **Car\_ID** | INTEGER | Y | Ссылка на сущность Car |
| BK | **Driver\_ID** | INTEGER | N | Ссылка на сущность Driver |
|  | **Checkdate** | Date | Y | Дата запроса |
|  | **Details** | VARCHAR(100) | Y | Детали |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Traffic Police** | | | | |
| **KEY** | **FIELD NAME** | **DATA TYPE / FIELD SIZE** | **REQUIRED?** | **NOTES** |
| PK | **ID** | INTEGER | Y | Первичный ключ |
|  | **Fullname** | VARCHAR(255) | Y | ФИО |
|  | **Post** | DECIMAL(10) | N | Должность |

# 2.3 СУБД SQL Server

**Microsoft SQL Server** — система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

Особенности функционирования SQL SERVER:

СУБД SQL server используются для создания, размещения, хранения и управления реляционными (табличными) базами данных на специальных серверах или в облаке. Они работают через настольные приложения и web- сайты. К основным преимуществам их функционирования относятся:

* высокоскоростной доступ к данным, обеспечиваемый надежной клиент-серверной архитектурой СУБД;
* простота работы и администрирования, обусловленные понятной структурой языка программирования SQL;
* безопасность хранения информации в БД – благодаря возможности шифрования данных и резервного копирования.
* Специфика работы сервера базы данных SQL server заключается в транзакционной обработке данных. Это означает, что по каждому запросу от СУБД обрабатывается и сохраняется небольшое количество информации.

Применение SQL server позволяет автоматизировать решение различных бизнес-задач, поддерживать проведение аналитики данных в режиме онлайн, отслеживать направление ресурсов СУБД, управлять транзакциями (операциями по обработке данных).

Виды SQL-server:

Для реализации функций СУБД на сегодняшний день чаще всего используются следующие SQL-серверы:

MS SQL server - многопользовательский программный продукт, разработанный компаний Microsoft, обладающий высокой производительностью и отказоустойчивостью, тесно интегрированный с ОС Windows. Этот сервер поддерживает удаленные подключения, работает с многими популярными типами данных, дает возможность создавать триггеры и хранимые данные, имеет практичные и удобные утилиты для настройки;

Oracle Database server СУБД, предназначенная для создания, консолидации и управления базами данных в облачной среде. Используя этот сервер, можно как автоматизировать обычные бизнес-операции, так и выполнять динамический многомерный анализ данных (OLAP), проводить операции с документами xml-формата и управлять разделенной и локальной информацией;

ІВМ DB2 - семейство СУБД для работы с реляционными базами данных, признанное самым производительным, имеющим высокие технические показатели и возможности масштабирования. SQL-серверы этой группы характеризуются мультиплатформенностью, способностью к мгновенному созданию резервных копий и восстановлению БД, реорганизации таблиц в онлайн-режиме, разбиению баз данных, определению пользователями новых типов данных; +

MySQL - СУБД, разработанная и поддерживаемая компанией Oracle. В основном она используется локальными или удаленными клиентами, позволяя им работать с таблицами разных типов, поддерживающих полнотекстовый поиск или выполняющих транзакции на уровне отдельных записей;

PostgreSQL – СУБД с открытым исходным кодом, работающая с объектно-реляционными базами данных. Также PostgreSQL предназначена для создания, хранения и извлечения сложных структур данных. Она поддерживает самые различные типы данных.

# 2.4 Microsoft SQL Server Management Studio

SQL Server Management Studio (SSMS) — это интегрированная среда для управления любой инфраструктурой SQL, от SQL Server до баз данных SQL Azure. SSMS предоставляет средства для настройки, наблюдения и администрирования экземпляров SQL Server и баз данных. Используйте SSMS для развертывания, мониторинга и обновления компонентов уровня данных, используемых приложениями, и создания запросов и скриптов.

Используйте SSMS для запроса, проектирования и управления базами данных и хранилищами данных, где бы они ни находились на локальном компьютере или в облаке.

SQL Server Management Studio (SSMS) утилита из Microsoft SQL Server 2005 и более поздних версий для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server. Утилита включает скриптовый редактор и графическую программу, которая работает с объектами и настройками сервера.

Главным инструментом SQL Server Management Studio является Object Explorer, который позволяет пользователю просматривать, извлекать объекты сервера, а также полностью ими управлять.

Также есть SQL Server Management Studio Express для Express версии сервера, которая является бесплатной. Однако в ней нет поддержки ряда компонентов (Analysis Services, Integration Services, Notification Services, Reporting Services) и SQL Server 2005 Mobile Edition.

Среда SQL Server Management Studio предназначена для доступа к службам Службы Analysis Services, Integration Services и Reporting Services, a также для их настройки, администрирования и управления ими. Хотя все три технологии бизнес-аналитики полагаются на среду SQL Server Management Studio, административные задачи, связанные с каждой из этих технологий, несколько отличаются.

При использовании последнего пакета обновления текущий выпуск SSMS поддерживает следующие 64-разрядные платформы:

Поддерживаемые операционные системы:

* Windows 11 (x64)
* Windows 10 (x64) версии 1607 (10.0.14393) и более поздних версий
* Windows Server 2022 (x64)
* Windows Server Core 2022 (x64)
* Windows Server 2019 (x64)
* Windows Server Core 2019 (x64)
* Windows Server 2016 (x64) 1
* Для 1 SSMS требуется платформа .NET Framework 4.7.2.

Дополнительные требования:

Права администратора необходимы для установки или обновления SSMS.

Контейнеры Windows не поддерживаются.

SSMS не поддерживается в решениях виртуализации приложений, таких как Microsoft App-V или MSIX для Windows, или сторонние технологии виртуализации приложений.

ЅЅМЅ работает только в Windows (AMD или Intel). Если вам требуется от Windows, средство, которое работает на платформах, отличных — это кроссплатформенное рассмотрите Azure Data Studio. Azure Data Studio средство для macOS, Linux и Windows.

Среда SQL Server Management Studio позволяет включать компоненты служб Reporting Services, администрировать серверы и базы данных, управлять ролями и заданиями.

В целом, SQL Server Management Studio (SSMS) — это мощный инструмент, предоставляющий полную среду для управления базами данных SQL Server. Благодаря набору инструментов и функций SSMS упрощает разработку, управление и поддержку баз данных, позволяя пользователям более эффективно работать с экземплярами SQL Server.

# ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ

# 3.1 Разработка базы данных

-- Создаём базу данных GAI

CREATE DATABASE GAI;

GO

USE GAI;

GO

-- Таблица автомобилей

CREATE TABLE Car (

Car\_ID INT PRIMARY KEY,

VIN VARCHAR(17) NOT NULL,

Brand VARCHAR(50),

Model VARCHAR(50),

Year INT,

Color VARCHAR(30)

);

GO

-- Таблица водителей

CREATE TABLE Driver (

Driver\_ID INT PRIMARY KEY,

Fullname NVARCHAR(100),

Address NVARCHAR(100),

Phone VARCHAR(20),

NumberDrive VARCHAR(20),

PassportData NVARCHAR(50)

);

GO

-- Таблица инспекторов и сотрудников (сотрудники с должностями)

CREATE TABLE TrafficPolice (

ID INT PRIMARY KEY,

Fullname NVARCHAR(100),

Post INT

);

GO

-- Таблица ролей

CREATE TABLE Role (

RoleID INT PRIMARY KEY,

RoleName NVARCHAR(100)

);

GO

-- Таблица проверок (инспекций)

CREATE TABLE DatabaseTable (

ID INT PRIMARY KEY,

TrafficPolice\_ID INT,

Car\_ID INT,

Driver\_ID INT,

CheckDate DATE,

Details NVARCHAR(255),

FOREIGN KEY (Car\_ID) REFERENCES Car(Car\_ID),

FOREIGN KEY (Driver\_ID) REFERENCES Driver(Driver\_ID)

);

GO

-- Таблица штрафов

CREATE TABLE Fines (

Fine\_ID INT PRIMARY KEY,

Driver\_ID INT,

Car\_ID INT,

Sum\_fine DECIMAL(18,2),

Date\_Violation DATE,

Date\_Payment DATE NULL,

FOREIGN KEY (Driver\_ID) REFERENCES Driver(Driver\_ID),

FOREIGN KEY (Car\_ID) REFERENCES Car(Car\_ID)

);

GO

-- Таблица пробега

CREATE TABLE MileageHistory (

Mileage\_ID INT PRIMARY KEY,

Car\_ID INT,

Mileage INT,

Date DATE,

FOREIGN KEY (Car\_ID) REFERENCES Car(Car\_ID)

);

GO

-- Таблица владения автомобилем

CREATE TABLE OwnershipHistory (

Ownership\_ID INT PRIMARY KEY,

Car\_ID INT,

Driver\_ID INT,

StartDate DATE,

EndDate DATE NULL,

FOREIGN KEY (Car\_ID) REFERENCES Car(Car\_ID),

FOREIGN KEY (Driver\_ID) REFERENCES Driver(Driver\_ID)

);

GO

-- Таблица регистрации

CREATE TABLE Register (

Register\_ID INT PRIMARY KEY,

Driver\_ID INT,

Car\_ID INT,

Status INT,

RegDate DATE,

FOREIGN KEY (Driver\_ID) REFERENCES Driver(Driver\_ID),

FOREIGN KEY (Car\_ID) REFERENCES Car(Car\_ID)

);

GO

-- Вставляем данные в Car

INSERT INTO Car (Car\_ID, VIN, Brand, Model, Year, Color) VALUES

(1, '1HGCM82633A123456', 'Toyota', 'Camry', 2020, 'Black'),

(2, '2HGCM82633A123457', 'Honda', 'Civic', 2018, 'White'),

(3, '3HGCM82633A123458', 'Nissan', 'Qashqai', 2022, 'Gray');

GO

-- Вставляем данные в Driver

INSERT INTO Driver (Driver\_ID, Fullname, Address, Phone, NumberDrive, PassportData) VALUES

(1, N'Иванов Иван Иванович', N'Москва', '1234567890', '1234567890', '1234 123456'),

(2, N'Петров Петр Петрович', N'Санкт-Петербург', '9876543210', '9876543210', '5678 987654'),

(3, N'Иванова Мария Ивановна', N'Москва', '5551234567', '5551234567', '9012 345678');

GO

-- Вставляем данные в Inspection

INSERT INTO DataDatabaseTable(ID, TrafficPolice\_ID, Car\_ID, Driver\_ID, CheckDate, Details) VALUES

(1, 1, 1, 1, '2025-04-07', N'Проверка документов'),

(2, 1, 2, 2, '2025-04-08', N'Проверка документов'),

(3, 2, 3, 3, '2025-04-09', N'Проверка технического состояния');

GO

-- Вставляем данные в Fine

INSERT INTO Fines(Fine\_ID, Driver\_ID, Car\_ID, Sum\_fine, Date\_Violation, Date\_Payment) VALUES

(1, 1, 1, 5000.00, '2025-04-07', NULL),

(2, 2, 2, 3000.00, '2025-04-08', NULL),

(3, 3, 3, 2000.00, '2025-04-09', NULL);

GO

-- Вставляем данные в Mileage

INSERT INTO MileageHistory(Mileage\_ID, Car\_ID, Mileage, Date) VALUES

(2, 2, 80000, '2025-04-08'),

(3, 2, 70000, '2024-01-01'),

(4, 2, 90000, '2025-01-01'),

(5, 3, 20000, '2024-06-01'),

(6, 3, 30000, '2025-03-01');

GO

-- Вставляем данные в Ownership

INSERT INTO OwnershipHistory (Ownership\_ID, Car\_ID, Driver\_ID, StartDate, EndDate) VALUES

(3, 2, 2, '2018-01-01', '2020-01-01'),

(4, 2, 3, '2020-01-01', NULL),

(5, 3, 4, '2022-01-01', NULL);

GO

-- Вставляем данные в Register

INSERT INTO Register (Register\_ID, Driver\_ID, Car\_ID, Status, RegDate) VALUES

(1, 1, 1, 2, '2025-04-07'),

(2, 2, 2, 1, '2025-04-08'),

(3, 3, 3, 1, '2025-04-09'),

(5, 1, 2, 3, '2025-04-10');

GO

-- Вставляем данные в Role

INSERT INTO Role (RoleID, RoleName) VALUES

(1, N'Инспекторы ДПС'),

(2, N'Сотрудники регистрационных отделов'),

(3, N'Аналитики и статистики'),

(4, N'Технический персонал'),

(5, N'Сотрудники ППС');

GO

-- Вставляем данные в Employee

INSERT INTO TrafficPolice(ID, Fullname, Post) VALUES

(1, N'Петров Петр Петрович', 101),

(2, N'Сидоров Сергей Сергеевич', 102),

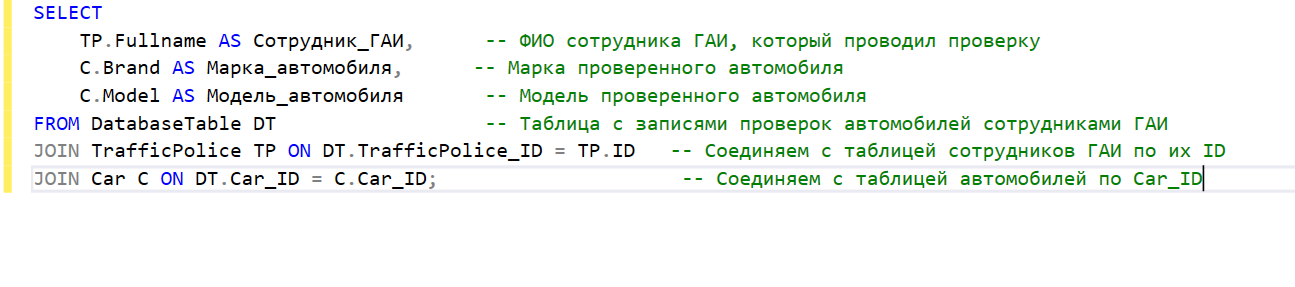
(3, N'Кузнецов Иван Иванович', 103),

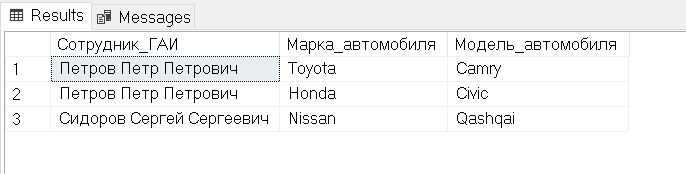
(4, N'Николаев Николай Николаевич', 104),

(5, N'Анатольевич Попов Владимир', 104);

GO

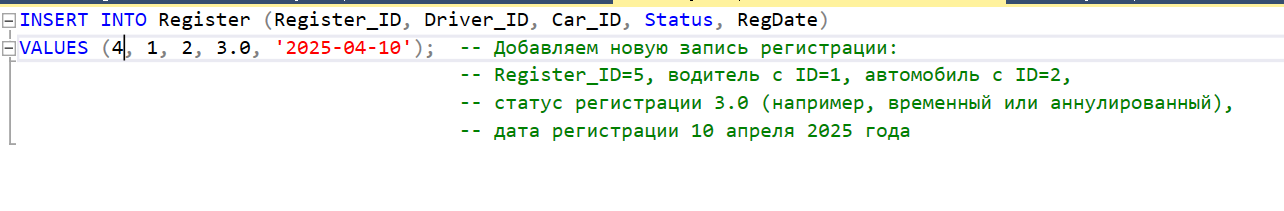
1. Какой сотрудник ГАИ проверил какой автомобиль

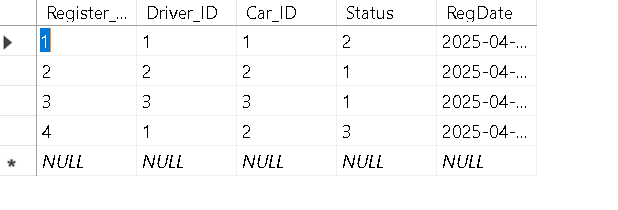




*Рис3.1 Сотрудник ГАИ проверил какой автомобиль*

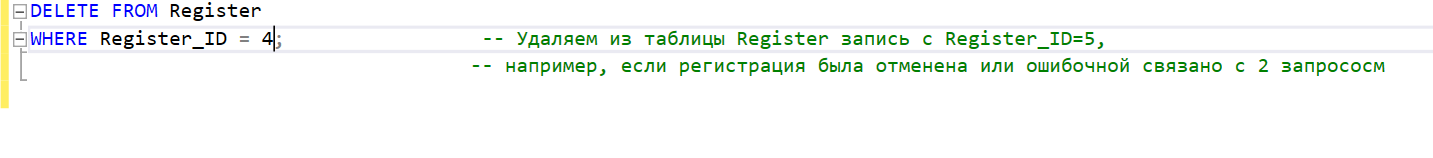
1. Добавить новую регистрацию автомобиля с новым статусом

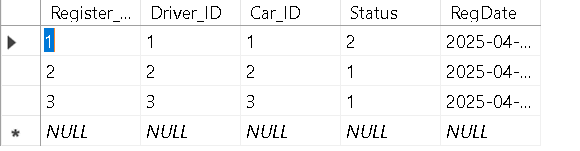




*Рис3.2 Добавили новый статус в очереди*

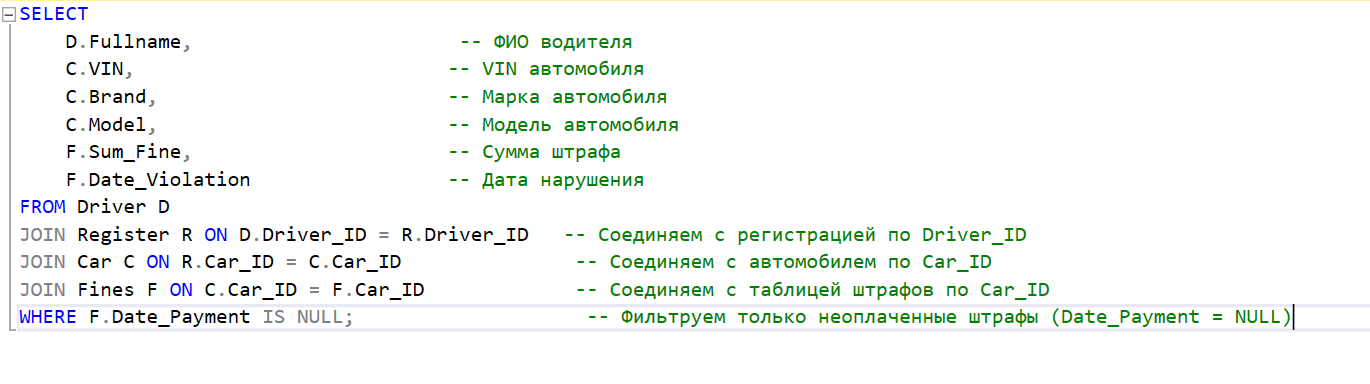
1. Удалить регистрацию автомобиля по ID

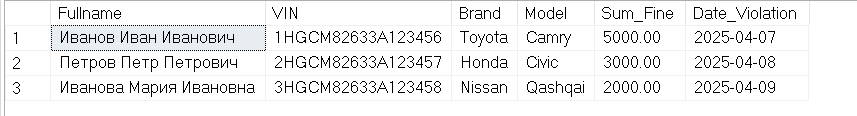




*Рис3.3 Удален статус из очереди*

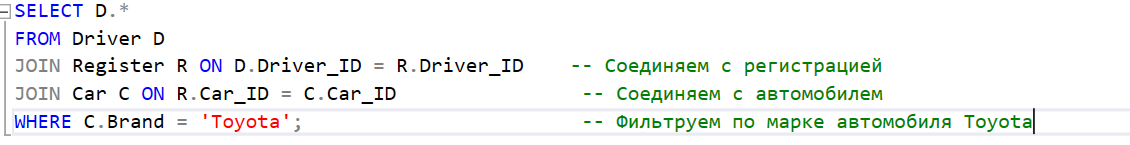
1. Найти всех водителей с неоплаченными штрафами

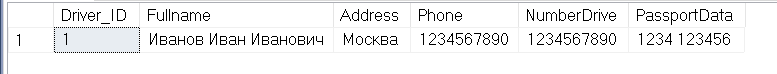




*Рис3.4 Список водителей с неоплаченными штрафами*

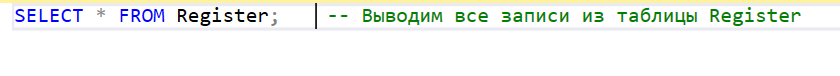
1. Получить информацию о водителях, владеющих автомобилями марки Toyota

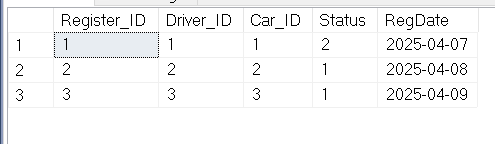




*Рис3.5 все водители владеющей марка Toyota*

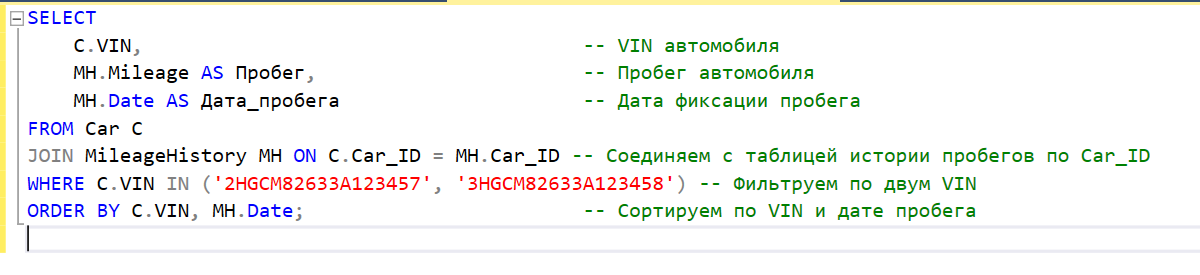
1. Показать все регистрационные записи автомобилей

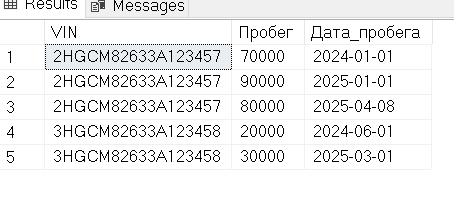




*Рис3.6 Все зарегистрированные автомобили*

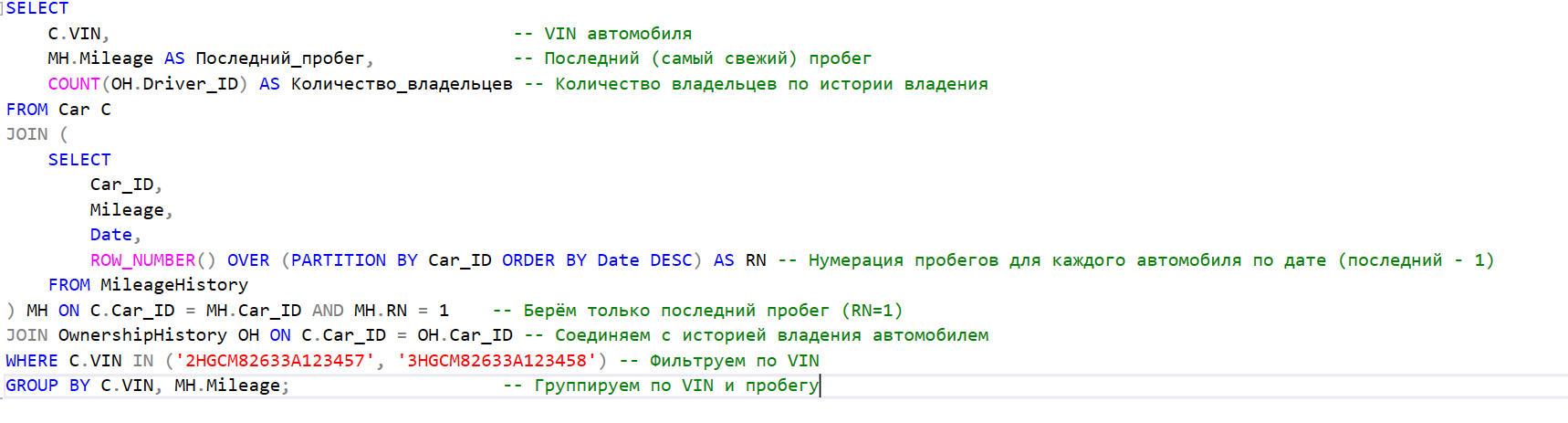
1. Показать всю историю пробегов для двух автомобилей по VIN

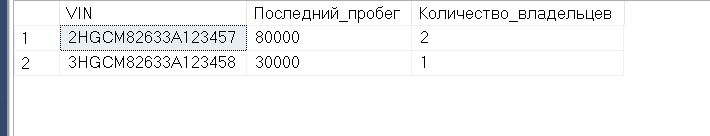




*Рис3.7 История пробега для двух автомобилей*

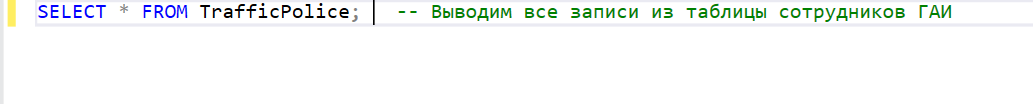
1. Вывести последний пробег и количество владельцев для каждого автомобиля

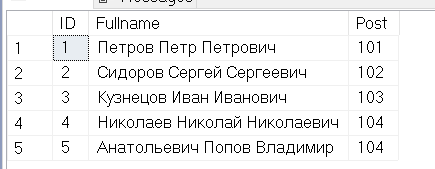




*Рис3.8 Последний пробег и владельцев для каждого автомобиля*

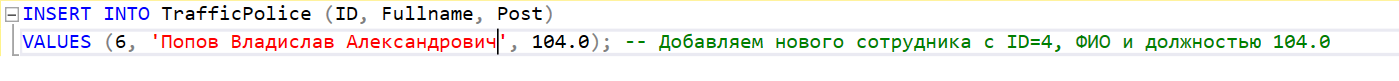
1. Вывести всех сотрудников ГАИ





*Рис3.9 Сотрудники ГАИ*

1. Добавить нового сотрудника ГАИ





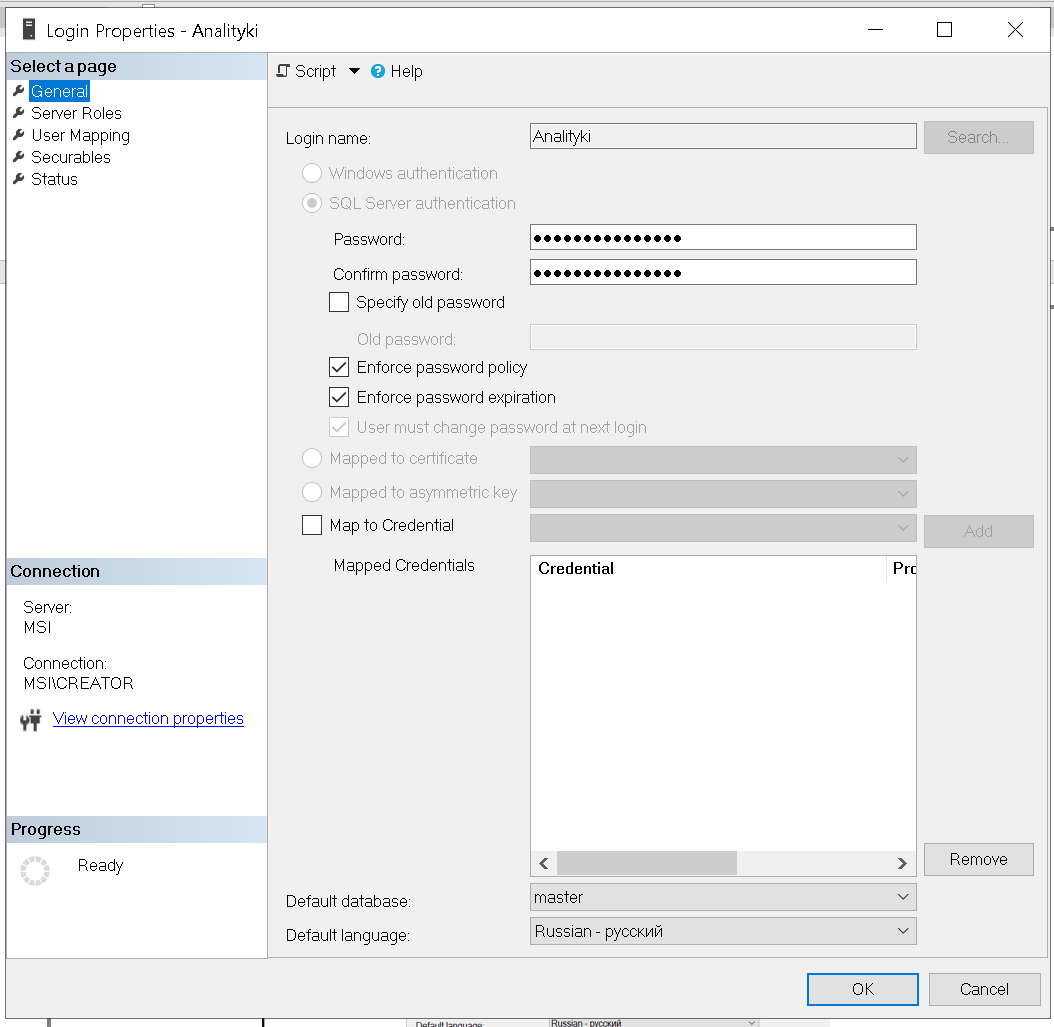
*Рис3.10 Добавление*

# 3.2 Введение в безопасность SQL Server

Среда MS SQL Server предоставляет множество различных функций для создания безопасных приложений баз данных. В каждой версии MS SQL Server есть свои средства безопасности, как и в каждой версии Windows, при этом возможности более поздних версий шире, чем возможности более ранних. Важно понимать, что сами по себе средства безопасности не могут гарантировать защиту приложения базы данных. Каждое приложение базы данных имеет уникальные требования, среду выполнения, модель развертывания, физическое расположение и количество пользователей. Некоторым приложениям, работающим локально, необходима минимальная защита, тогда как другим локальным приложениям или приложениям, развернутым через Интернет, могут требоваться строгие меры безопасности вместе с постоянным мониторингом и контролем.

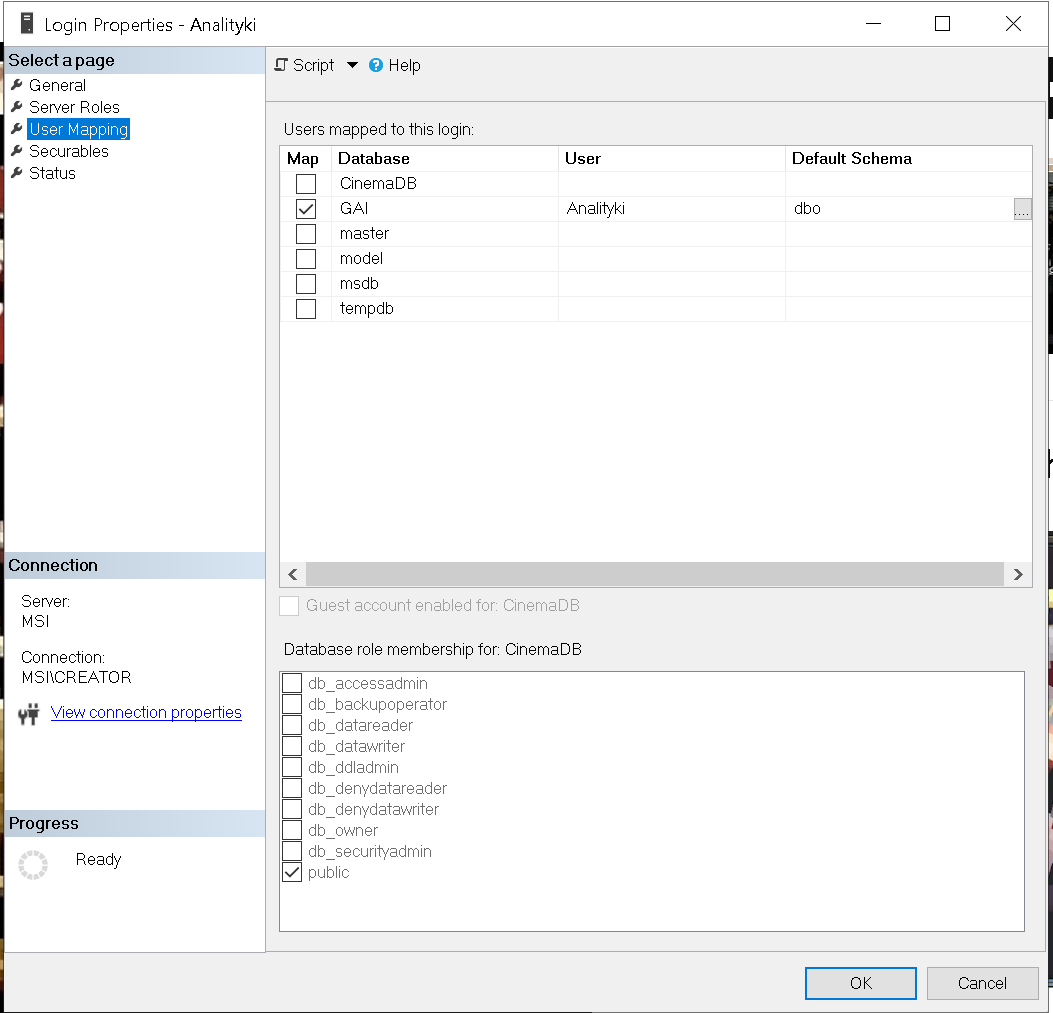
Помимо стандартных для системы управления базами данных функций, MS SQL Server содержит большой набор интегрированных служб по анализу данных. Доступ к данным, расположенным на MS SQL Server могут получить любые приложения, разработанные с помощью технологии.Net и среды разработки Visual Studio. Для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server используется инструментарий утилиты SQL Server Management Studio. В ней существует поддержка ряда компонент и средств по созданию и управлению базами данных, средств аналитической обработки данных (Analysis Services), средств отчётности (Reporting Services), а также множество средств, упрощающих разработку приложений.

# 3.3 Управление безопасностью уровня сервера

 Создание пользователя и присвоение ему роли для базы данных.

*Рис3.10 Добавление пользователя*

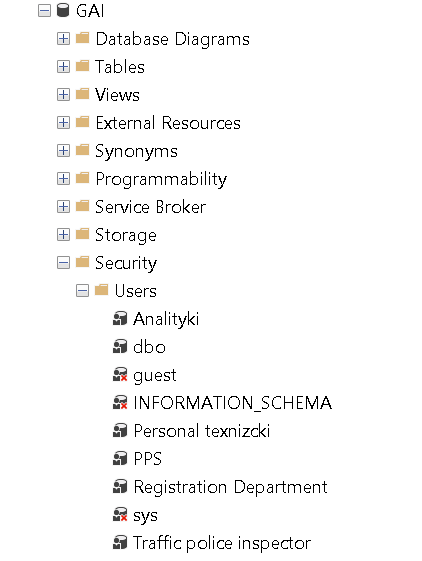
Подключение пользователя к базе данных



*Рис3.11 Разграничение прав доступа к базе данных*

# 3.4 Управление участниками уровня базы данных

Создание пяти пользователей Инспекторов ДПС, Сотрудников регистрационных отделов, Аналитиков и статистиков, Технического персонала, Сотрудников ППС на уровне сервера, для дальнейшего подключения их к базе данных.

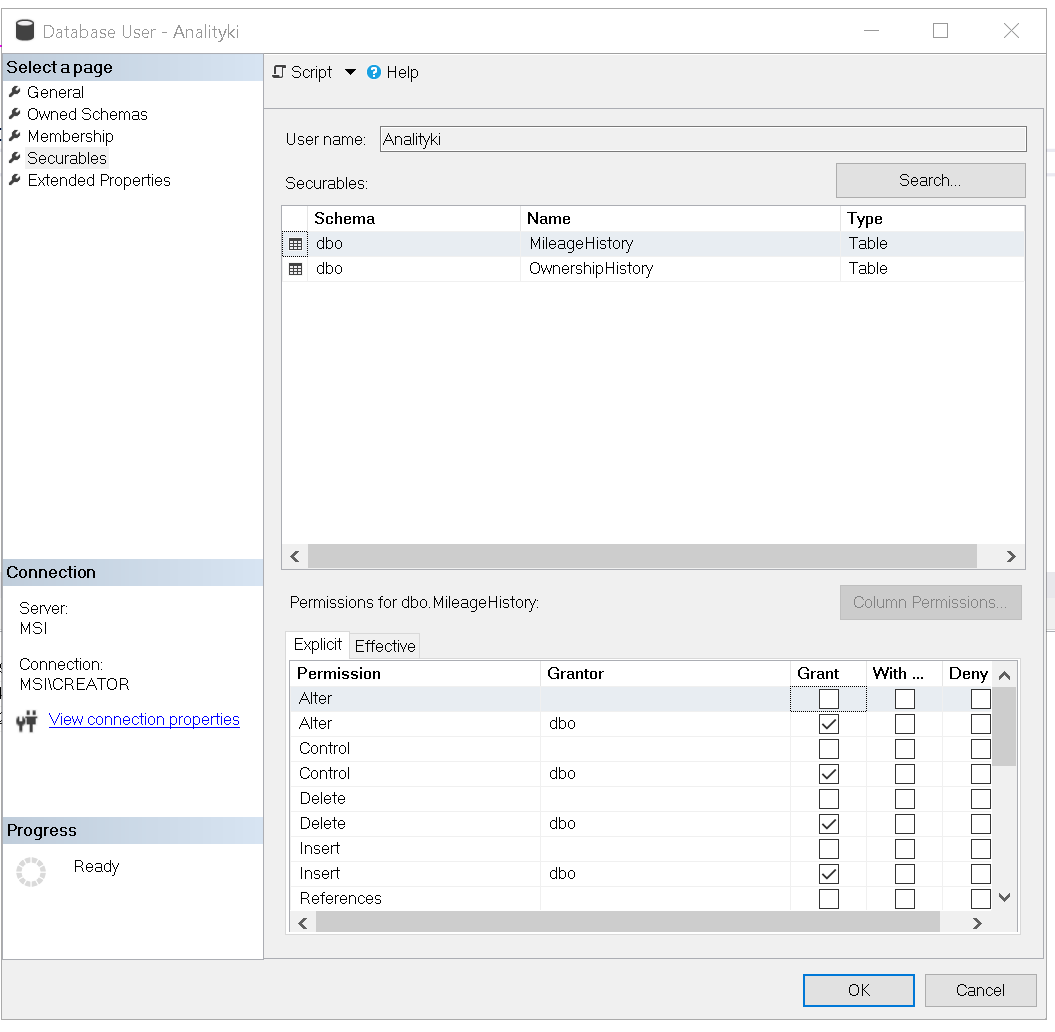


*Рис3.12 Созданные пользователи в базе данных GAI*

# 3.5 Управление разрешениями уровня базы данных

В SQL Server системная база данных хранит внутренние метаданные, которыми администратор должен уметь управлять, а пользователи - корректно пользоваться в рамках своих прав.

* **Инспекторы ДПС** имеют права на просмотр и обработку данных проверок и штрафов, а также использование функций для работы с транспортными средствами и водителями.
* **Сотрудники регистрационных отделов** обладают правами на изменение регистрационных данных и управление сведениями о владельцах транспортных средств.
* **Аналитики и статистики** имеют права только на чтение и использование функций для анализа и формирования отчетов без возможности изменять данные.
* **Технический персонал** обладает расширенными правами, включая управление структурой базы, настройку и обслуживание функций, а также доступ к системным метаданным для поддержки и администрирования.
* **Сотрудники ППС** имеют ограниченный доступ, позволяющий использовать необходимые функции и просматривать данные для выполнения служебных обязанностей, без прав на изменение данных или метаданных.

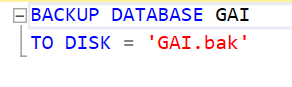


*Рис3.13 Выдача прав пользователю*

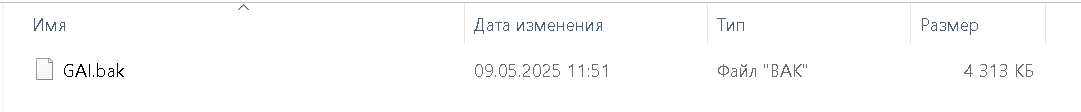
# 3.6 Резервное копирование баз данных

BackUp (резервное копирование) – это процесс создания копии данных с целью их сохранения и возможного восстановления в случае их потери, повреждения или удаления. Резервное копирование может быть выполнено как на физических носителях, таких как внешние жесткие диски, USB – накопители, так и в различных облачных хранилищах. Резервное копирование играет важную роль в обеспечении безопасности данных и защите от потери информации.

Cоздание BackUp для базы данных GAI

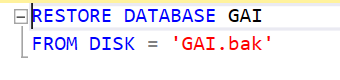


*Рис3.14 Код для создания файла восстановления*



*Рис3.15 Файл восстановления*

Запрос восстановления файла с BackUp нашей базы данных

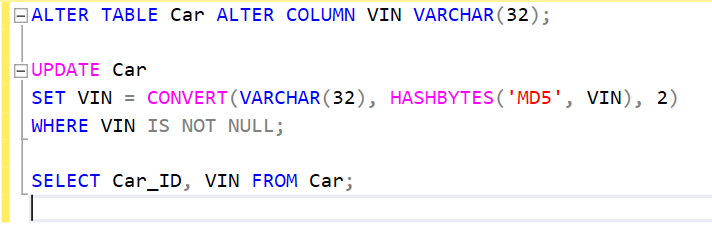


*Рис3.16 Код для восстановления базы данных*

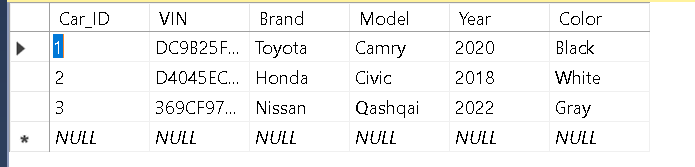
# 3.7 Шифрование данных баз данных

Шифрование базы данных – использование технологии шифрования для преобразования информации, хранящейся в базе данных (БД), в шифротекст, что делает её прочтение невозможным для лиц, не обладающих ключами шифрования.

Данные шифруются перед записью на диск, а так же шифруются в самой программе во время ввода пароля и логина и сравнивается потом друг с другом в коде, что решает проблему защиты «неактивных» данных, но не обеспечивает сохранность информации при передаче по каналам связи или во время использования.



*Рис3.17 Запрос на шифрование данных*



*Рис3.18 Зашифрованные данные*

# ЗАКЛЮЧЕНИЕ

База данных ГАИ приобретает особую значимость в современных условиях внедрения цифровых технологий и автоматизации государственных служб. С развитием технологий управление информацией о транспортных средствах, водителях, проверках, штрафах и регистрациях становится ключевым для эффективного функционирования системы ГАИ. Автоматизация процессов учета и контроля позволяет повысить оперативность работы, улучшить качество обслуживания граждан и обеспечить прозрачность и безопасность данных.

В рамках выполнения данной работы была разработана концептуальная модель базы данных ГАИ с использованием диаграммы «сущность-связь» (ERD), спроектирована и реализована база данных в среде Microsoft SQL Server Management Studio. Для удобства работы сотрудников ГАИ создан автоматизированный интерфейс и функционал информационной системы на платформе C# WPF. Данная система обеспечивает быстрый доступ к данным, возможность создания отчетов, редактирования и добавления информации, а также защищенный вход с использованием методов шифрования.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕАРТУРЫ

1. ER – Diagramm Entities and connections 2023 [Электронный ресурс]. https://practicum.yandex.ru/blog/chto-takoe-er-diagramma/
2. Введение в MySQL 2023 [Электронный ресурс]. https://metanit.com/sql/mysql/1.1.php
3. Краткое руководство. Подключение и запрос экземпляра SQL Server с помощью SQL Server Management Studio (SSMS) 2024
4. [Электронный ресурс]. https://learn.microsoft.com/ru-ru/sql/ssms/quickstarts/ssms-connect-query-sql-server?view=sql-server-ver16
5. Руководство по SQL 2022 [Электронный ресурс]. https://proselyte.net/tutorials/sql/
6. Резервное копирование и восстановление баз данных SQL Server 2025

[Электронный ресурс]. https://learn.microsoft.com/ru-ru/sql/relational-databases/backup-restore/back-up-and-restore-of-sql-server-databases?view=sql-server-ver16

1. Соединение таблиц 2021[Электронный ресурс]. https://metanit.com/sql/mysql/5.1.php
2. Создание полной резервной копии базы данных 2025 [Электронный ресурс]. https://learn.microsoft.com/ru-ru/sql/relational-databases/backup-restore/create-a-full-database-backup-sql-server?view=sql-server-ver16
3. Создание пользователя базы данных 2025 [Электронный ресурс]. https://learn.microsoft.com/ru-ru/sql/relational-databases/security/authentication-access/create-a-database-user?view=sql-server-ver16
4. Шифрование в базах данных SQL Server 2023 [Электронный ресурс]. https://www.osp.ru/winitpro/2013/05/13035359