МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет ЦОО ФИСТ

Кафедра Измерительно-вычислительные комплексы

Дисциплина Базы данных

**КУРСОВОЙ ПРОЕКТ (РАБОТА)**

Тема

Разработка базы данных «Инкассаторская служба»

Выполнил студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Курбаниязов Шерали /

Подпись инициалы, фамилия

Курс 3 Группа ИСТбд-32

Направление/ специальность 09.03.02 «Информационные системы и технологии»

Руководитель ассистент

должность, ученая степень, ученое звание

Скалкин Антон Михайлович

фамилия, имя, отчество

Дата сдачи:

« » 2021 г.

Дата защиты:

« » 2021 г.

Оценка:

Ульяновск

2021 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет ЦОО ФИСТ

Кафедра Измерительно-вычислительные комплексы

Дисциплина Базы данных

**ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ (РАБОТУ)**

студенту ИСТбд-32 Курбаниязов Шерали

группа фамилия, инициалы

Тема проекта (работы)

Разработка базы данных «Инкассаторская служба»

Срок сдачи законченного проекта (работы) « » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г.

Исходные данные к проекту (работе)

Содержание пояснительной записки (перечень подлежащих разработке вопросов)

1. Описание поставленной задачи
2. Модель хранения данных представленная в ER-диаграмме
3. Описание структуры разработанной базы данных

Руководитель ассистент / А.М. Скалкин /

должность подпись инициалы, фамилия

« » 2021 г.

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**/** Курбаниязов Шерали **/**

подпись инициалы, фамилия

« » 2021 г.

**Содержание**

[**Введение** ………………………………..……………………………………….4](#_Toc90033861)

[**1. Теоретическая часть** ……………………………………….…………….....6](#_Toc90033862)

[1.1.Описание предметной области …………………………………...………...6](#_Toc90033863)

[1.2 ER-диаграмма с описанием ………………………………………..……….7](#_Toc90033864)

[**2. Проектирование АИС «Инкассаторской службы»** ………………...….10](#_Toc90033865)

[2.1. Структурный подход к проектированию системы автоматизации кассовых операций ………………………………………………………..……10](#_Toc90033866)

[**3. Реализация системы автоматизации кассовых операций в** **программной среде SQL Server** ………………..……………………………15](#_Toc90033867)

[3.1. Обоснование выбора среды реализации …………………………..……..15](#_Toc90033868)

[Исключительные ситуации …………………………..………………………..24](#_Toc90033869)

[**Заключение** ……………………..……………………………………………..25](#_Toc90033870)

[**Список литературы** ……………………………..……………………………28](#_Toc90033871)

# **Введение**

Роль информационных технологий в современных условиях постоянно возрастает. Деятельность, как отдельных людей, так и целых организаций все в большей степени зависит от их информированности и способности эффективно использовать имеющуюся информацию. Современное материальное производство и другие сферы деятельности все больше нуждаются в информационном обслуживании, переработке огромного количества информации. Информатизация на основе внедрения компьютерных и телекоммуникационных технологий является реакцией общества на потребность в существенном увеличении производительности труда в информационном секторе общественного производства, где сосредоточено более половины трудоспособного населения. Проблема информационных технологий не обошла стороной многие учебные заведения. Приемной комиссии вуза или колледжа приходится решать широкий круг задач. Работа приемной комиссии является одной из важнейших составляющих деятельности учебной части каждого учебного заведения.

К сожалению, до сих пор задачи инкассаторской службы в большинстве государственных учреждениях выполняются вручную. Все это очень усложняет обработку данных и многократно увеличивает вероятность ошибок. Автоматическое формирование списков и ведомостей не только сокращает время обработки огромного количества информации, но и уменьшает вероятность ошибок. Именно поэтому так важен переход приемной комиссии к электронным системам учета.

Целью курсовой работы является разработка автоматизированной информационной системы «Инкассаторская служба», для этого необходимо выполнить следующие задачи:

- Собрать необходимую информацию о деятельности инкассаторской службе;

- Разработать алгоритм решения поставленной цели, правильно указав последовательность выполнения соответствующих команд для получения необходимых результатов;

- В качестве среды программирования выбрать среду позволяющую создавать программные продукты с интуитивно понятным интерфейсом;

- Реализовать алгоритм решения в объектно-ориентированной среде;

При написании программного кода необходимо учесть возможность реализации следующих функций:

- Возможность добавлять, удалять, редактировать, сохранять информацию об абитуриенте;

- Просматривать данные об сотруднике;

- Сохранять информацию в базе данных.

# **1. Теоретическая часть**

## 1.1.Описание предметной области

При создании АИС должны быть выполнены действия по изучению деятельности предприятия. Вначале производится сбор информации о предприятии, его целях и задачах, структуре и [финансово-хозяйственной деятельности](https://pandia.ru/text/category/finansovo_hazyajstvennaya_deyatelmznostmz/). Также изучаются внешние процессы, взаимодействующие с предприятием, и среда, в которой предприятие осуществляет свою деятельность. В целом, до момента непосредственного проектирования ИС, должно быть получено комплексное описание предприятия и его бизнеса.

Перевозка ценностей представляет собой наиболее привлекательный вид посягательств со стороны преступников, которые могут применять все виды преступлений, включая разбой и убийство. Преступники могут быть не только внешними или внутренними по отношению к атакуемой организации, но и образовывать их объединения. Нераскрытые преступления становятся укреплением позиций преступников в моральном и материальном аспектах. В связи с этим предпринимаемые отраслью меры общего противодействия преступности направлены на минимизацию вероятности потери ценностей и нанесения ущерба людям. Для этого определяется минимально необходимый уровень обеспечения безопасности, соблюдение которого должно стать обязательным для всех профессиональных перевозчиков ценностей. Стандарт нацелен на создание экономически выгодного делового процесса налично- денежного обращения, обеспечивающего необходимый уровень физической безопасности бригад инкассации, сопряженных с ними сотрудников клиентов и общественной безопасности, а также сохранность перевозимых ценностей. Положения стандарта призваны дать основу для реализации согласованных минимальных требований к условиям, методам и технологиям, применяемым при предоставлении услуг в сфере инкассации и кассового обслуживания, необходимость соблюдения которых добровольно признается всеми участниками отрасли. Положения стандарта учитывают лучшие отечественные и международные рекомендации в области обеспечения безопасности предоставления кассово-инкассаторских услуг.

Безопасность деятельности по инкассации и перевозке ценностей достигается путем использования нормативных правовых, организационных и технических мер защиты, а также путем создания системы управления безопасностью и совершенствованию системы взаимодействия между организациями отрасли, с государственными органами, направленной на создание безопасных условий работы бригад инкассаторов на маршрутах.

## 1.2 ER-диаграмма с описанием

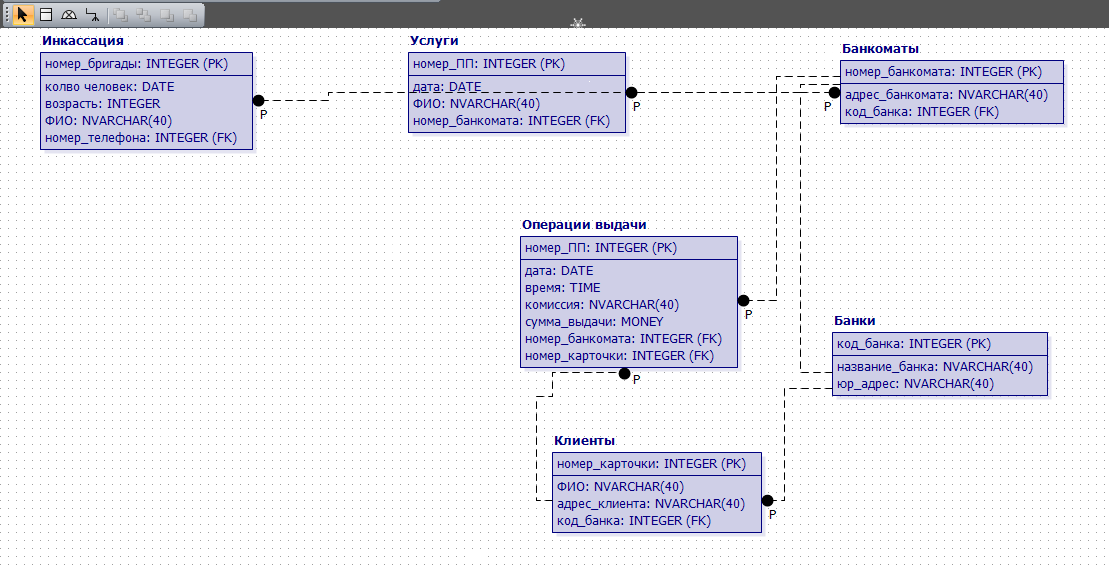


Рис. 1. – ER диаграмма

Связи между классами, показанные в инфологической модели, в дата логической модели отображаются либо за счет совместного расположения связанных элементов, либо путем объявления связей между ними. Данные системы учета кассовых операций хранятся в связанных таблицах. Структура основных таблиц базы представлена ниже:

Таблица 1

Сотрудники.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Табельный номер | числовой | Длинное целое |
| 2 | ФИО | текстовый | 150 |
| 3 | Код должности | числовой | Длинное целое |
| 4 | Должность | текстовый | 150 |
| 5 | Код подразделения | числовой | Длинное целое |
| 6 | Подразделение | текстовый | 150 |
| 7 | Адрес | текстовый | 100 |
| 8 | Дата рождения | дата/время | - |
| 9 | Место рождения | текстовый | 100 |
| 10 | Гражданство | текстовый | 50 |

В таблице «Сотрудники» ключевым полем является поле Табельный номер. Именно ключевое поле однозначно определяет каждую запись в таблице. Ключевые поля используются для быстрого поиска и связи данных из разных таблиц при помощи запросов, форм и отчетов. Если правильно заданы ключевые поля, то исключается возможность дублирования информации в базе данных. Данная таблица связана почти со всеми другими таблицами базы, что показано на схеме данных (рис.2).

Таблица 2

Подразделения.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Код подразделения | числовой | Длинное целое |
| 2 | Наименование подразделения | текстовый | 150 |

Ключевым полем таблицы «Подразделения» является поле Код подразделения. Оно однозначно определяет номер каждого подразделения в системе. Данная таблица связана с такими таблицами, как «Сотрудники», «Расходный кассовый ордер», «Приходный кассовый ордер», «Авансовый отчет».

Таблица 3

Должности.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Код должности | числовой | Длинное целое |
| 2 | Наименование должности | текстовый | 150 |

В данной таблице ключевым полем является Код должности. Таблица имеет связь 1:М с таблицей «Сотрудники».

Таблица 4

Справочник счетов.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | № счета | тестовый | 50 |
| 2 | Наименование счета | текстовый | 100 |

В таблице «Справочник счетов» ключевым является поле № счета. Таблица связана с таблицами «Приходный кассовый ордер» и «Расходный кассовый ордер» связью 1:М.

Таблица 5

Приходный кассовый ордер.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Номер | числовой | Длинное целое |
| 2 | Дата | дата/время | - |
| 3 | ФИО | текстовый | 150 |
| 4 | Табельный номер | числовой | Длинное целое |
| 5 | Код структурного подразделения | числовой | Длинное целое |
| 6 | Корреспондирующий счет | текстовый | 50 |
| 7 | Дебет | текстовый | 50 |
| 8 | Тип | текстовый | 100 |

В данной таблице ключевыми являются поле Номер и поле Дата. Вместе они однозначно определяют каждую операцию. Имеются связи с таблицами «Сотрудники», «Подразделения», «Справочник счетов».

Таблица 6

Расходный кассовый ордер

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Номер | числовой | Длинное целое |
| 2 | Дата | дата/время | - |
| 3 | ФИО | текстовый | 150 |
| 4 | Табельный номер | числовой | Длинное целое |
| 5 | Код структурного подразделения | числовой | Длинное целое |
| 6 | Корреспондирующий счет | текстовый | 50 |
| 7 | Кредит | текстовый | 50 |
| 8 | Тип | текстовый | 100 |

В таблице «Расходный кассовый ордер» ключевыми являются поля Номер и Дата. Связи данной таблицы аналогичны связям, имеющимся в таблице «Приходный кассовый ордер».

Таблица7

Авансовый отчет.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Имя поля | Тип поля | Размер поля |
| 1 | Номер | числовой | Длинное целое |
| 2 | Дата | дата/время | - |
| 3 | Табельный номер | числовой | Длинное целое |
| 4 | Подотчетное лицо | текстовый | 150 |
| 5 | Подразделение | текстовый | 50 |
| 6 | Назначение аванса | текстовый | 100 |
| 7 | Получено | денежный | - |
| 8 | Израсходовано | денежный | - |
| 9 | Остаток | денежный | - |
| 10 | Перерасход | денежный | - |
| 11 | Дебет/счет | текстовый | 50 |
| 12 | Дебет/сумма | денежный | - |
| 13 | Кредит/счет | текстовый | 50 |
| 14 | Кредит/сумма | денежный | - |
| 15 | Тип | текстовый | 100 |

В таблице «Авансовый отчет» ключевыми являются поля Номер и Дата. Имеется связь типа М:1 с таблицами «Сотрудники» и «Подразделения.

Схема спроектированной базы данных, т.е. связи и отношения между сущностями показаны на схеме данных.

# **2. Проектирование АИС «Инкассаторской службы»**

## 2.1. Структурный подход к проектированию системы автоматизации кассовых операций

При создании банка основной его задачей является получение прибыли. Но для успешного функционирования банка необходимо соблюдать ряд определенных условий, от них напрямую зависит то, как будет работать банк, его прибыль и убытки, а так же возможность его дальнейшего расширения и предоставления клиентам новых услуг. В связи с тем, что направления работы банка осуществляются с учетом анализа рисков, возможной прибыли и потерь. Возникает необходимость в введении учета кассовых операций, и возможность оперативного анализа полученных доходов и их дальнейшее распределение по усмотрению аналитического отдела. Создание программного продукта, который удовлетворяет ряду требований предъявляемых для автоматизации учета кассовых операций, является одной из главных задач, которыми должен заняться отдел по техническому и программному обеспечению банка.

Программа, разработанная в рамках данного курсового проекта, является наиболее подходящей для автоматизации учета кассовых операций коммерческого банка.

Прежде, чем приступать к созданию системы автоматизированной обработки информации, необходимо было сформировать понятия о предметах, фактах и событиях, которыми будет оперировать данная система. Для того, чтобы привести эти понятия к той или иной модели данных, необходимо заменить их информационными представлениями. Одним из наиболее удобных инструментов унифицированного представления данных, независимого от реализующего его программного обеспечения, является модель "сущность-связь" (entity - relationship model, ER - model).

Модель "сущность-связь" была предложена в 1976 г. Питером Пин-Шэн Ченом.

Модель "сущность-связь" основывается на некой важной семантической информации о реальном мире и предназначена для логического представления данных. Она определяет значения данных в контексте их взаимосвязи с другими данными. Важным для нас является тот факт, что из модели "сущность-связь" могут быть порождены все существующие модели данных. Любой фрагмент предметной области может быть представлен как множество сущностей, между которыми существует некоторое множество связей [19].

Источником поступления данных являются сведения о клиентах банка и суммах, вносимых ими в кассу или получаемых из кассы.

Выходными данными являются отчеты, которые рассчитываются после ввода исходных данных. Все отчеты предоставляются по аналогии с данными, которые представлены в формах программного продукта. Выходными документом являются отчет за период о произведенных операциях по счетам клиентов и кассовая книга. В зависимости от выбранных параметров они могут менять свое содержание и внешний вид.

Для четкого представления структуры нашей системы, необходимо рассмотреть инфологическую модель базы данных.

На этапе инфологического проектирования представляется модель заданной предметной области. Фактическим стандартом инфологического проектирования является ER-модель, которая имеет в основе 2 базовых понятия: сущность и связь. Инфологическая модель дает формализованное описание предметной области независимо от структур данных, исключая неоднозначность за счет использования средств формальной логики. Модель нашей программы приведена на Рис.1



Рис.1. Инфологическая модель предметной области.

После инфологического проектирования базы данных следует построение даталогической модели.

Под даталогической понимается модель, отражающая логические взаимосвязи между элементами данных безотносительно их содержания и физической организации. При этом даталогическая модель разрабатывается с учетом конкретной реализации СУБД, также с учетом специфики конкретной предметной области на основе ее инфологической модели.

Основными задачами даталогического проектирования является создание корректной схемы БД и нормализация исходного отношения.

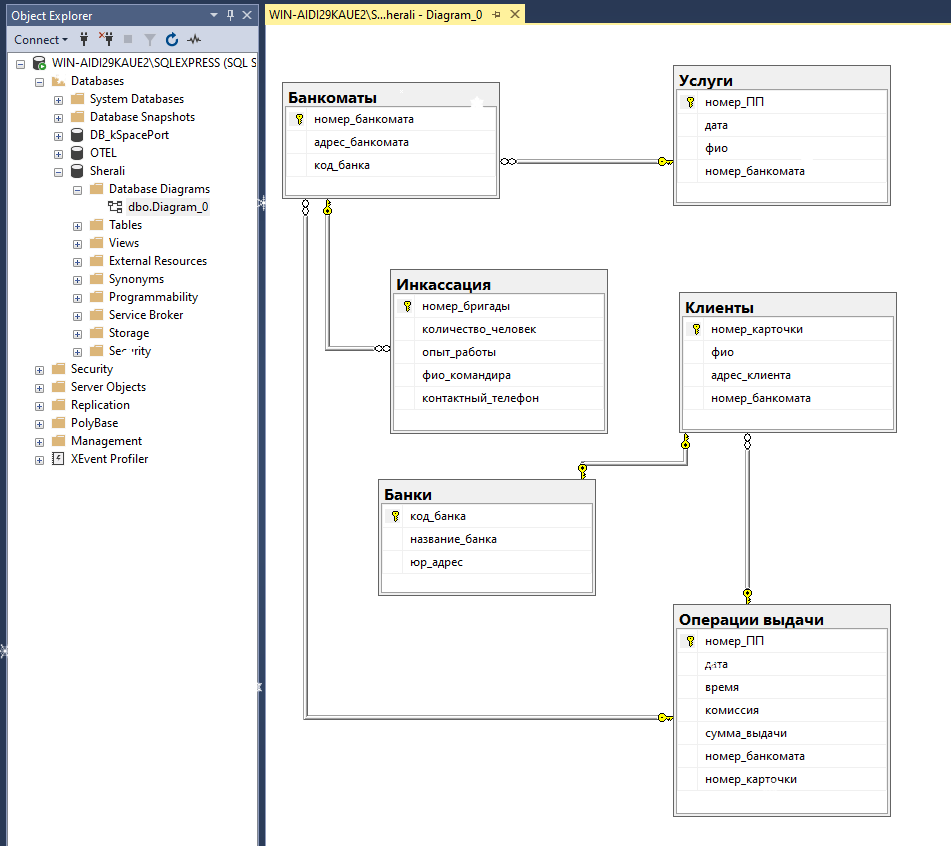


Рис.2. Схема базы данных

На основе данной схемы уже производится физическое проектирование системы.

Все процедуры событий для системы хранятся в модуле форм (Рис.3). При создании первой процедуры события для формы SQL Server автоматически создает модуль формы. Модуль формы представляет способ хранения в одном месте всего кода, который относится только к отдельной форме. Как правило, модули форм содержат только процедуры событий, но в них также могут функции.



Рис.3. Дерево программных модулей

«Главная» форма – форма, через которую осуществляется взаимодействие остальных форм и просмотр отчетов.

Форма «Сотрудники» - форма для ввода, просмотра и редактирования справочников «Сотрудники», «Должности» и «Подразделения».

Формы «Расходный кассовый ордер», «Приходный кассовый ордер» и «Авансовый отчет» - формы для ввода, редактирования и просмотра справочников «Расходный кассовый ордер», «Приходный кассовый ордер», «Авансовый отчет» соответственно и выполнения проводок о выдаче/получении денежных средств в кассу. Пункты меню «Расходный кассовый ордер», «Приходный кассовый ордер» и «Авансовый отчет» также могут работать в нескольких режимах, а именно ввод, изменение и просмотр данных, поэтому блок-схемы технологического процесса для этих пунктов будут выглядеть аналогичным образом.

# **3. Реализация системы автоматизации кассовых операций в** **программной среде SQL Server**

## 3.1. Обоснование выбора среды реализации

Современный рынок СУБД представлен большим количеством программных продуктов, из который наиболее многочисленными и мощными по своим возможностям являются такие пакеты, как Clarion Database Developer, DataEase, DataFlex, dBase IV, Microsoft SQL Server, Microsoft FoxPro, Paradox R:BASE и др.

Система управления базами данных Microsoft SQL Server является одним из самых популярных приложений в семействе настольных СУБД. Все версии SQL Server имеют в своем арсенале средства, значительно упрощающие ввод и обработку данных, поиск данных и предоставление информации в виде таблиц, графиков и отчетов. Начиная с версии SQL Server 2000, появились также Web-страницы доступа к данным, которые пользователь может просматривать с помощью программы Internet Explorer. Помимо этого, SQL Server позволяет использовать электронные таблицы и таблицы из других настольных и серверных баз данных для хранения информации, необходимой приложению. Присоединив внешние таблицы, пользователь SQL Server будет работать с базами данных в этих таблицах так, как если бы это были таблицы SQL Server. Популярность СУБД Microsoft SQL Server обусловлена следующими причинами:

* доступность в изучении и понятность позволяют SQL Server являться одной из лучших систем быстрого создания приложений управления базами данных;
* СУБД полностью русифицирована;
* возможность использования OLE технологии;
* интегрированность с пакетами Microsoft Office;
* визуальная технология позволяет постоянно видеть результаты своих действий и корректировать их; кроме того, работа с конструктором форм может существенно облегчить дальнейшее изучение таких систем программирования, как Visual Basic или Delphi;
* широко и наглядно представлена справочная система;
* наличие большого набора «мастеров» по разработке объектов.

Существуют разные варианты использования SQL Server с точки зрения архитектуры приложения. Иногда SQL Server (файл MDB) используется просто как ядро, которое управляет данными, находящимися с таблицами. SQL Server позволяет использовать те данные, которые уже были накоплены раньше другими программными средствами и, следовательно, имеют другой формат путем *импорта* существующей таблицы базы данных, рабочего листа электронной таблицы или текстового файла, созданных приложениями MS-DOS или Windows, во внутренний формат базы данных SQL Server (MDB). Естественно, что SQL Server может также *экспортировать* данные из таблиц базы данных формата MDB в любой формат, из которого можно импортировать данные. Помимо файлов баз данных, SQL Server может работать непосредственно с файлами электронных таблиц, текстовыми файлами, документами HTML, адресными книгами или импортировать данные из этих файлов и документов XML.

Таким образом, MS SQL Server, являясь СУБД реляционного типа, в которой разумно сбалансированы все средства и возможности, типичные для современных СУБД, разумнее всего использовать для разработки базы данных для учета кассовых операций.

1. **Руководство пользователя**

Система является обобщённой моделью инкассаторской службы, которая предназначена для упрощения работы пользователя с объёмом информации в структурированном виде. Она предоставляет информацию пользователю в доступном, интуитивно понятном виде, проверяет корректность вводимых пользователем данных. Также она позволяет выполнять такие пользовательские функции, как: отображение списка сотрудников по названию должности, подсчёт среднего рейтинга сотрудников, улучшение рейтинга сотрудников.

1. После запуска проекта пользователь увидит домашнюю страницу (рис. 5.1), на ней находится общая информация об автоматизированной информационной системе «Космодрома», шапка для навигации по сайту и кнопки для входа и выхода из системы. Регистрация в системе недоступна, так как космодром закрытое предприятие, войти в систему может сотрудник.

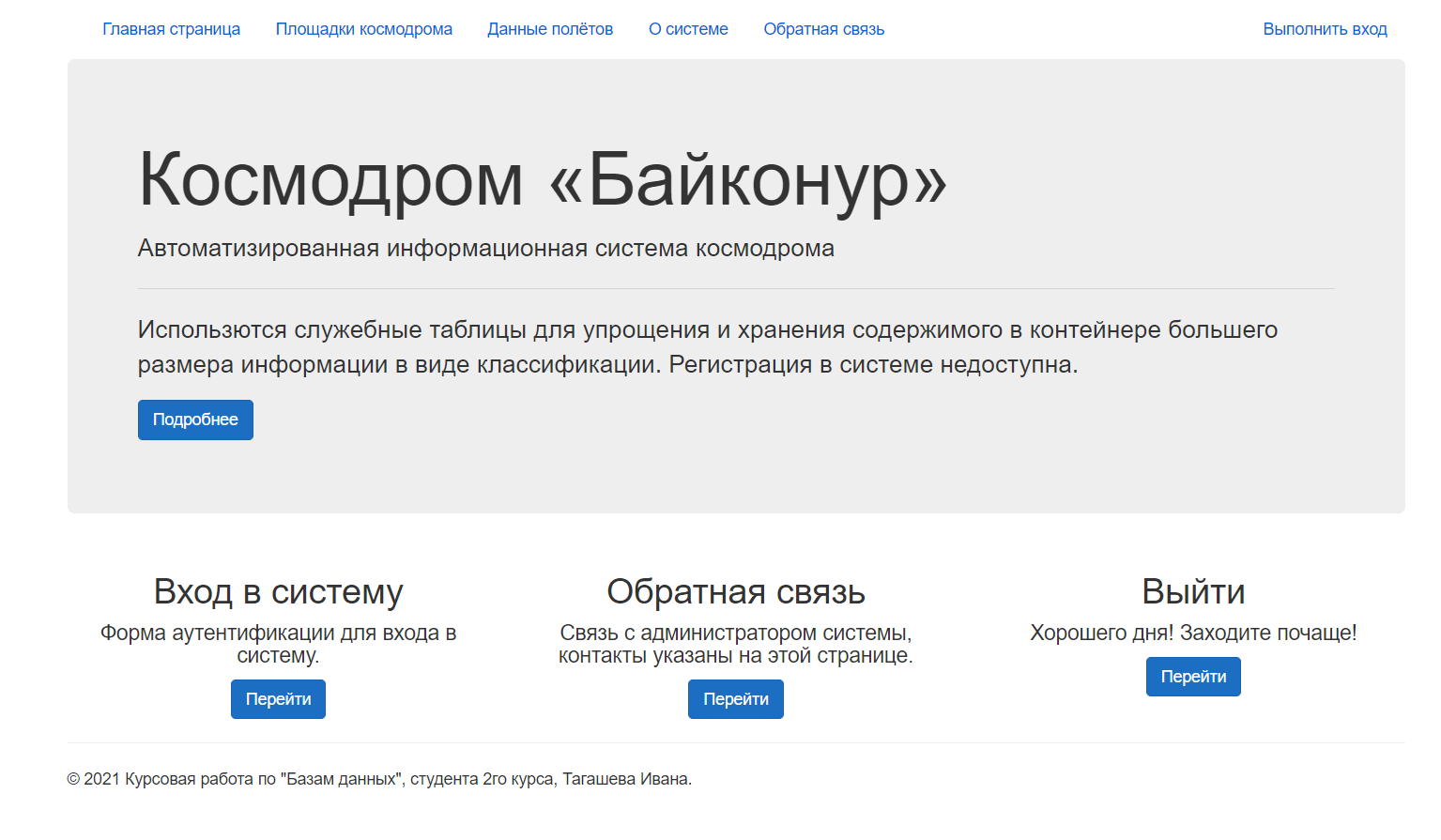


Рисунок 4 – Главная страница

1. Перейдем к полю «Выполнить вход» (рис. 5.2). Данная операция позволяет посетителю сайта использовать все возможности пользователя. Здесь пользователю предлагаются несколько полей для ввода данных, после успешного заполнения которых, необходимо нажать кнопку «Войти в систему» для перехода к следующей странице в зависимости от вашей роли в системе.

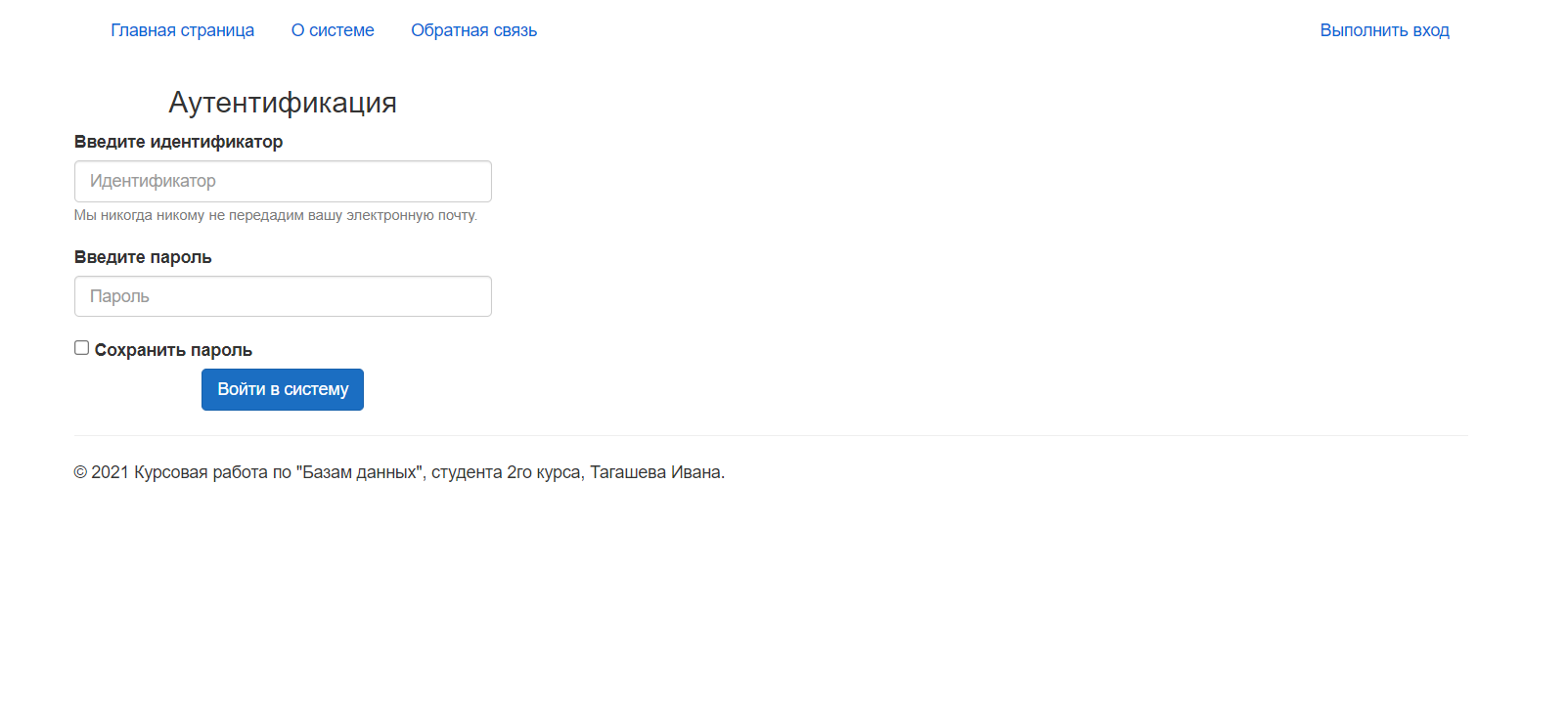


Рисунок 5 – Аутентификация.

1. После успешного входа в системе пользователя встречает сообщение об «Успешной аутентификации». После чего ему предлагается «Перейти» в личный кабинет. (рис. 5.3).

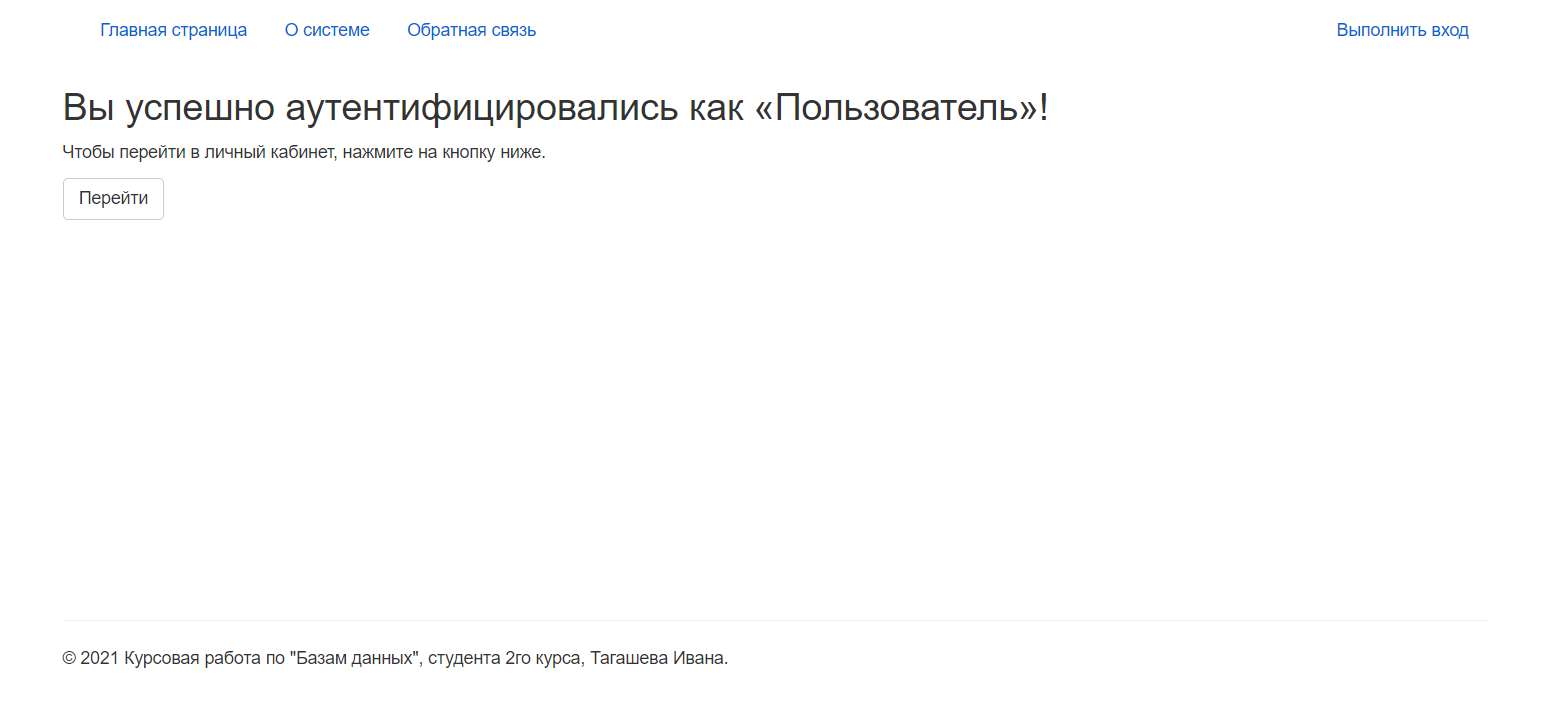


Рисунок 6 – Успешный вход пользователя.

1. Аутентифицированный пользователь попадает на свою домашнюю страницу, которая выглядит в табличном виде, здесь у пользователя появляется возможность ознакомиться со списком таблиц и данных, которые ему доступны для чтения. (рис. 5.4).

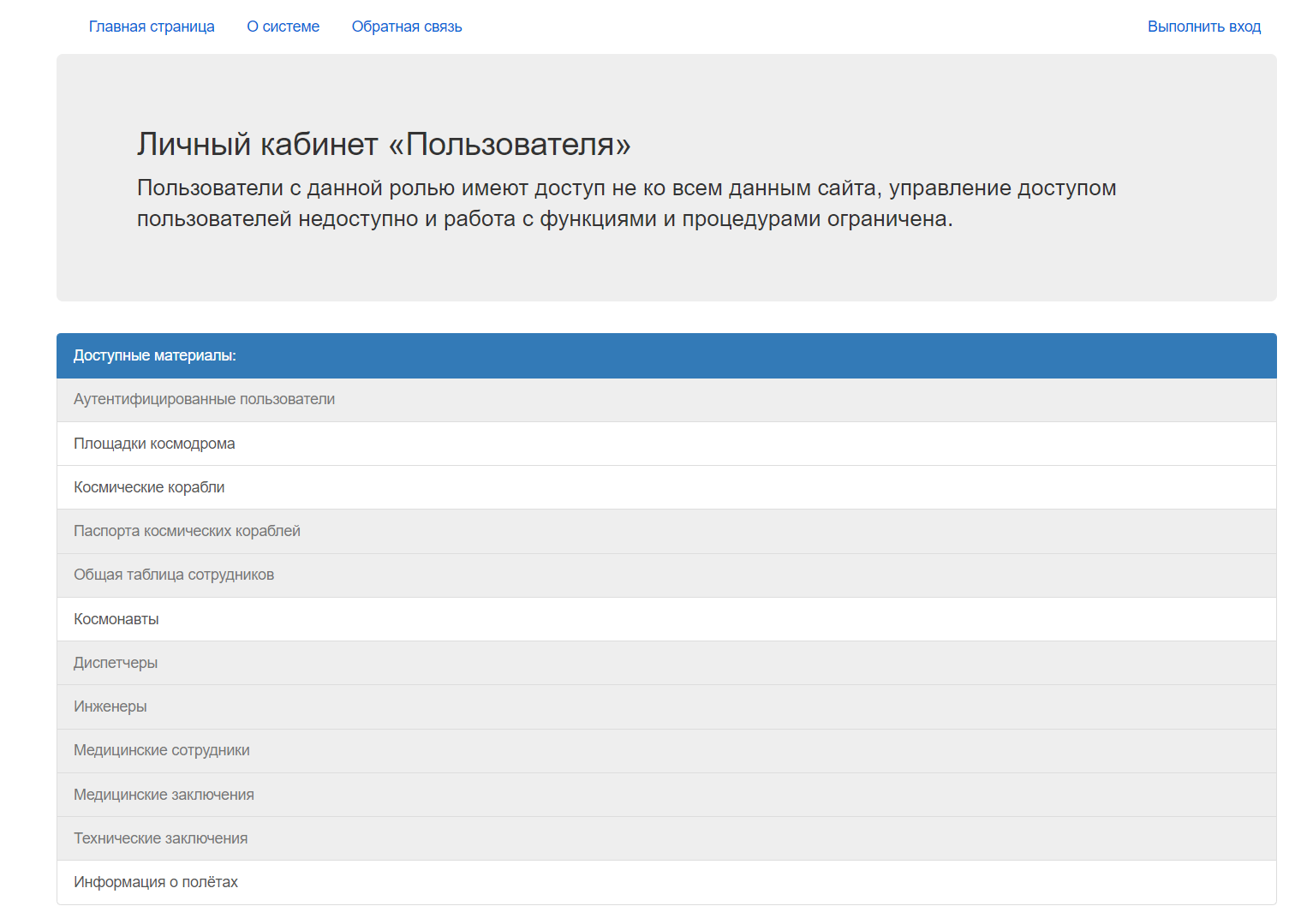


Рисунок 7 – Личный кабинет пользователя.

1. Для ознакомления с открытыми данными пользователю представлен список таблиц в личном кабинете.

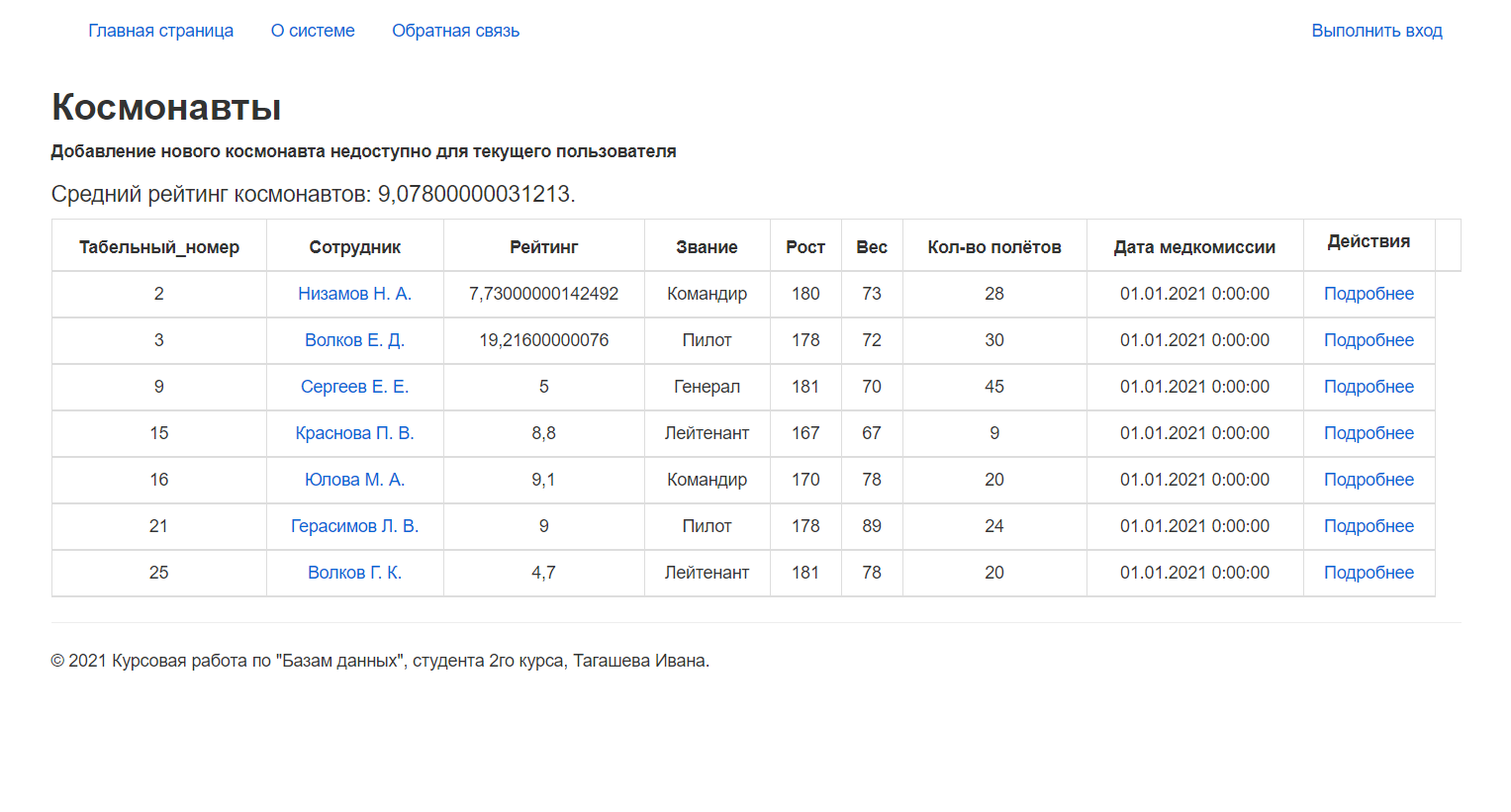


Рисунок 8 – Пример «режима-чтения» для Пользователя.

1. На странице «Космонавтов» для пользователя доступна функция, которая производит расчёт среднего рейтинга космонавтов. Если его заинтересовал, какой-либо космонавт, он может воспользоваться кнопкой «Подробнее».

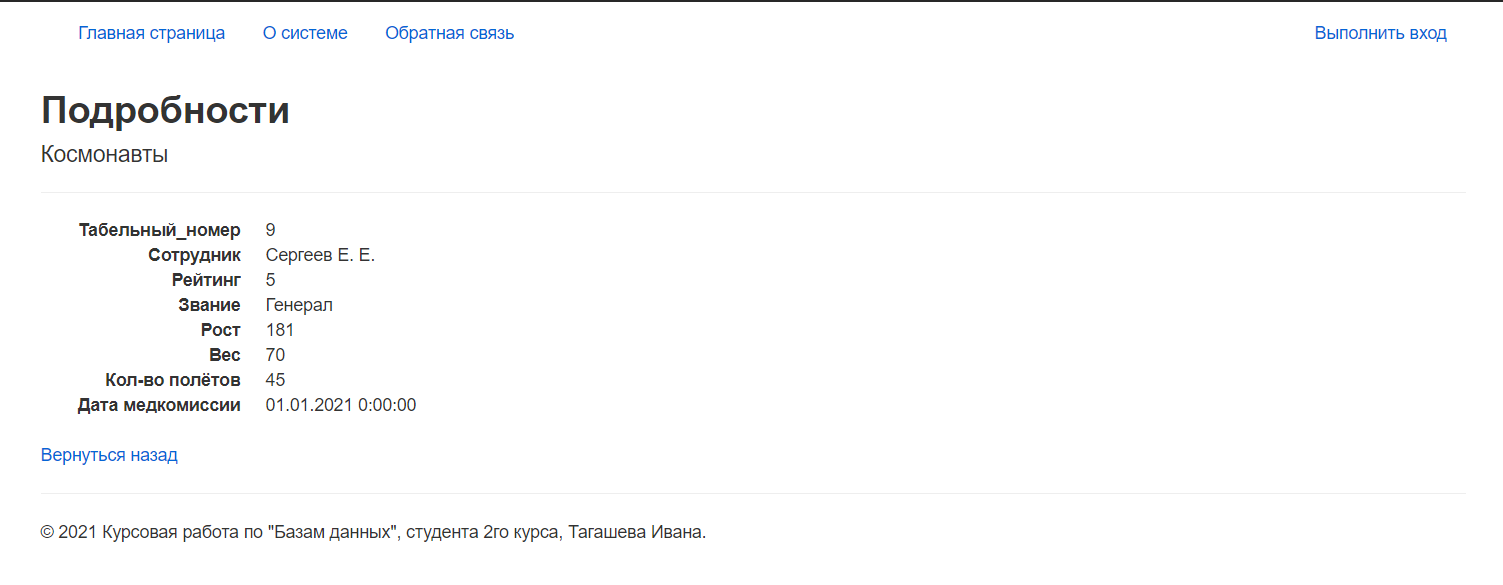


Рисунок 9 – Подробнее.

1. Пользователь с ролью «Администратор» имеет полный доступ к функционалу в системе «Космодрома». У него также имеется свой личный кабинет для упрощения работы с таблицами.

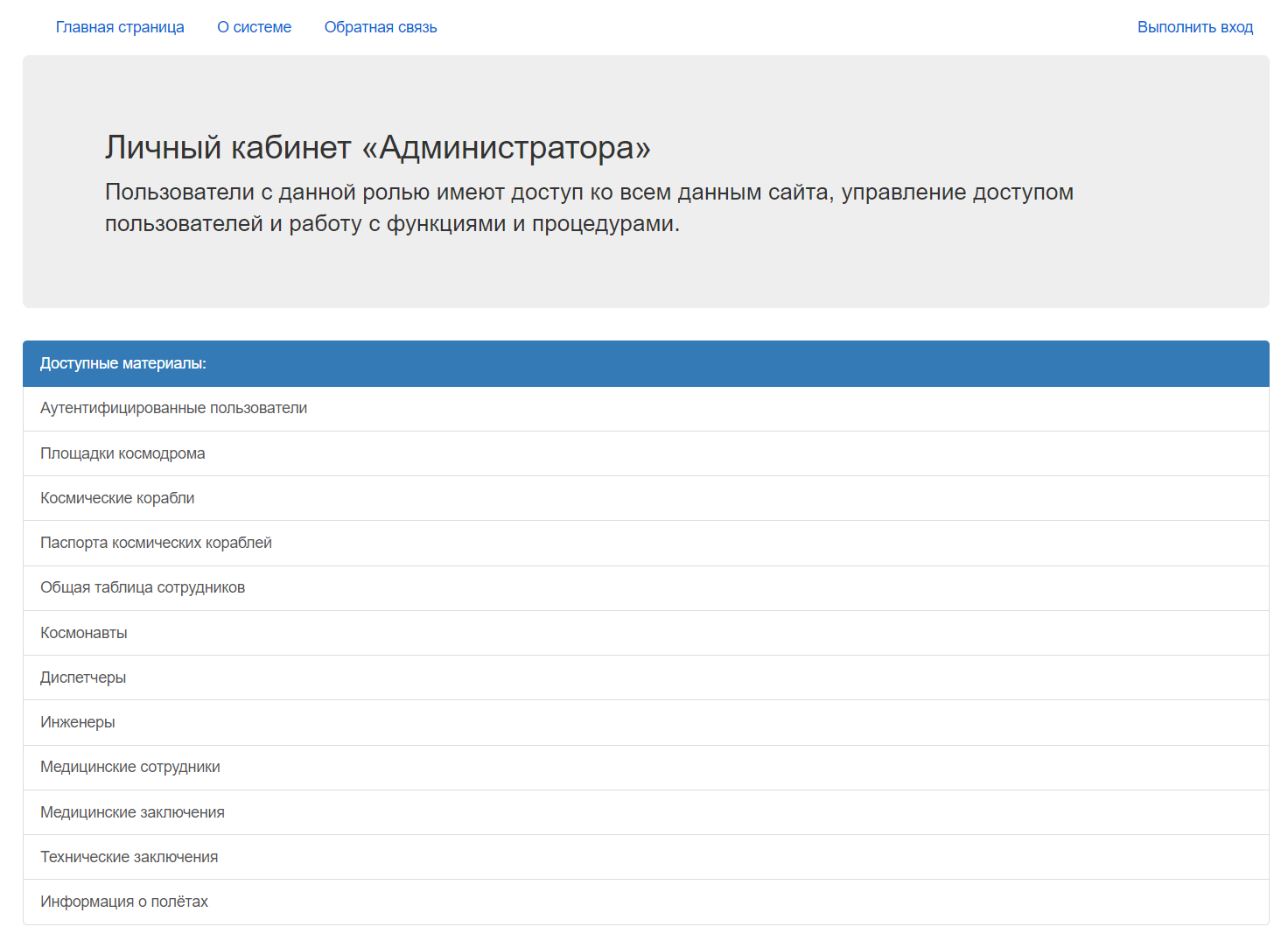


Рисунок 10 – Личный кабинет «Администратора»

1. Основные возможности администратора системы можно также рассмотреть на примере «Сотрудников». Такой пользователь может: добавлять новых сотрудников (рисунок 5.9), изменять информацию о нынешних сотрудниках (рисунок 5.10), удалять информацию (рисунок 5.10).

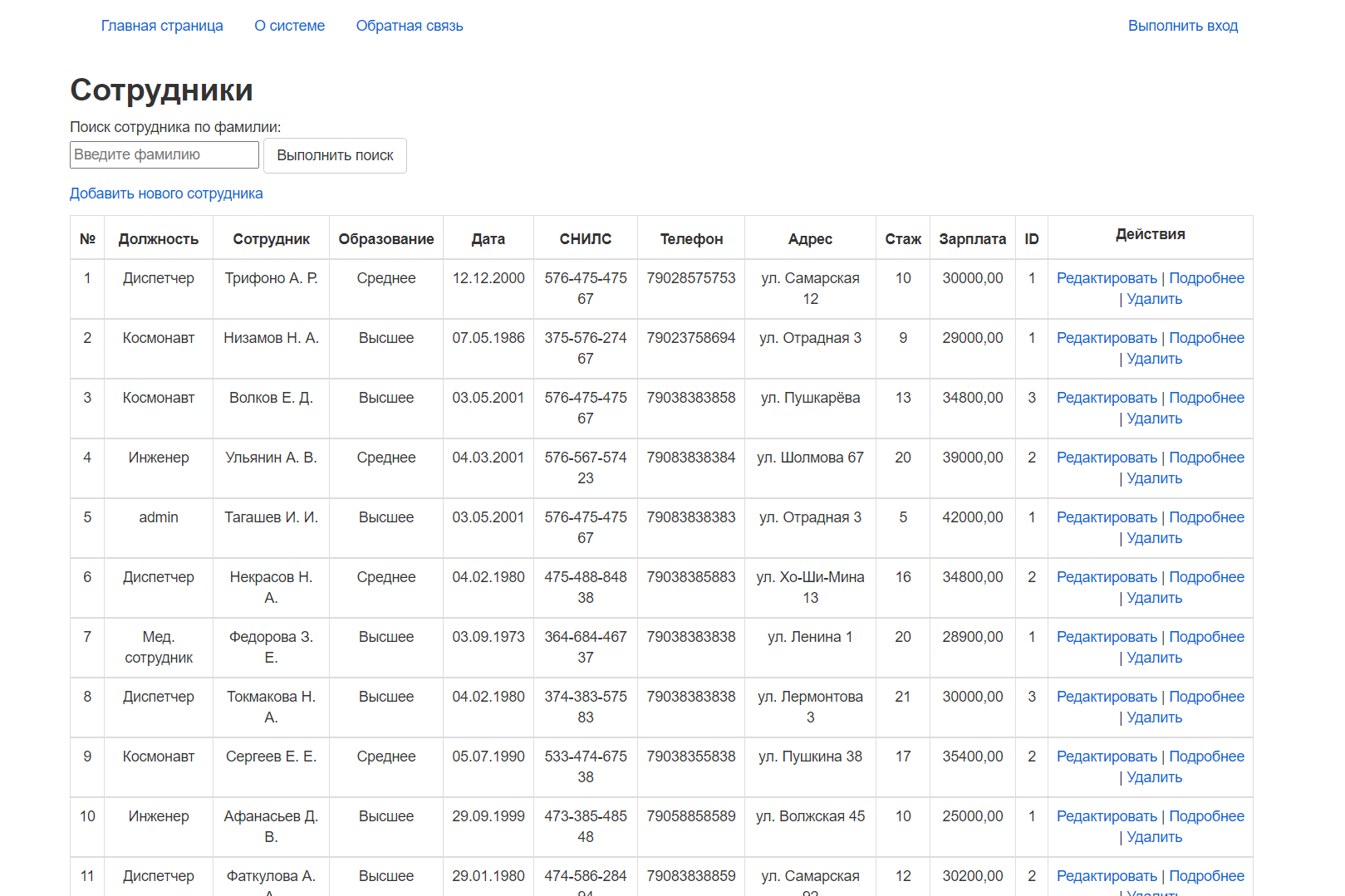


Рисунок 11 – Внешний вид таблицы у «Администратора»

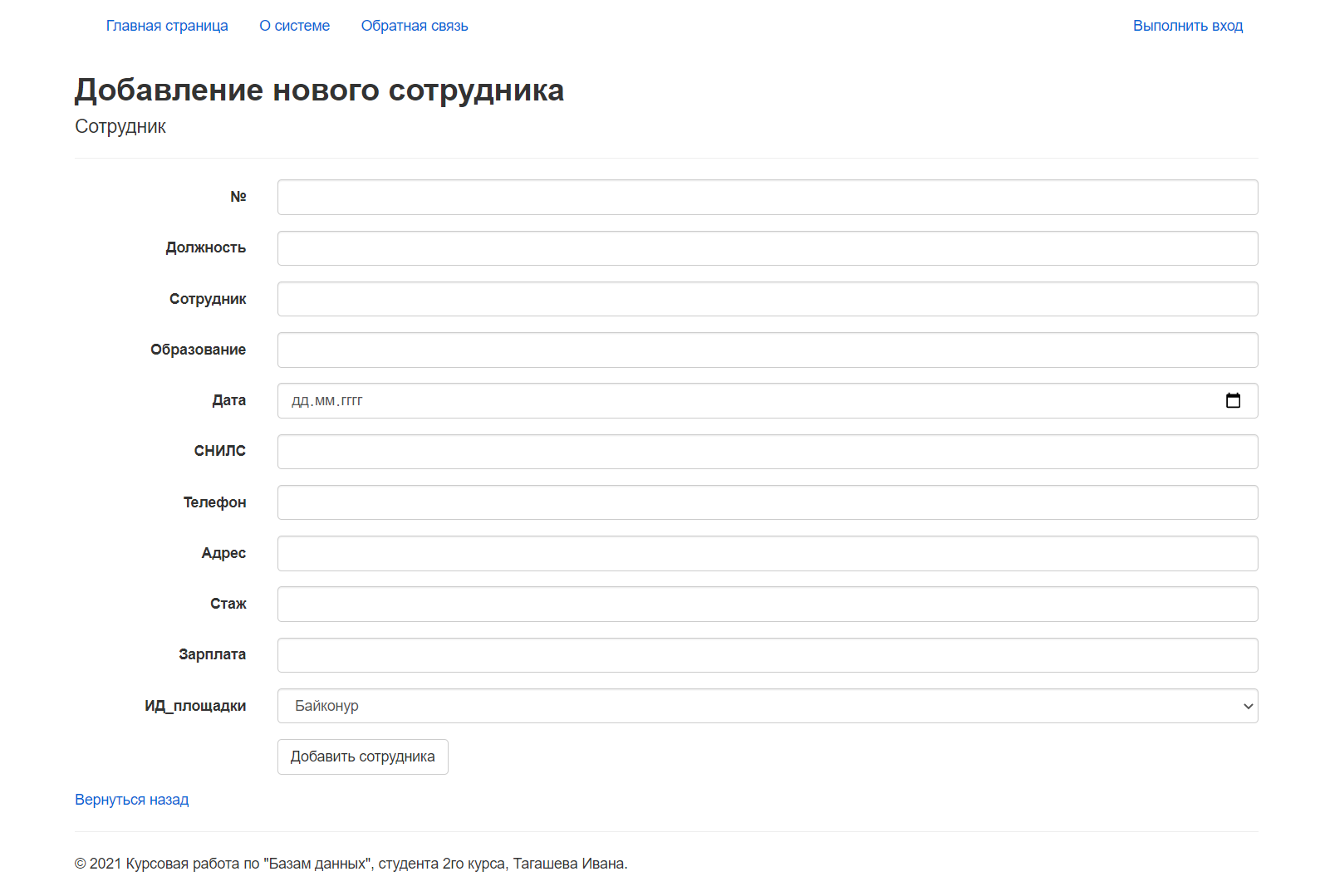


Рисунок 12 – Добавление нового сотрудника.

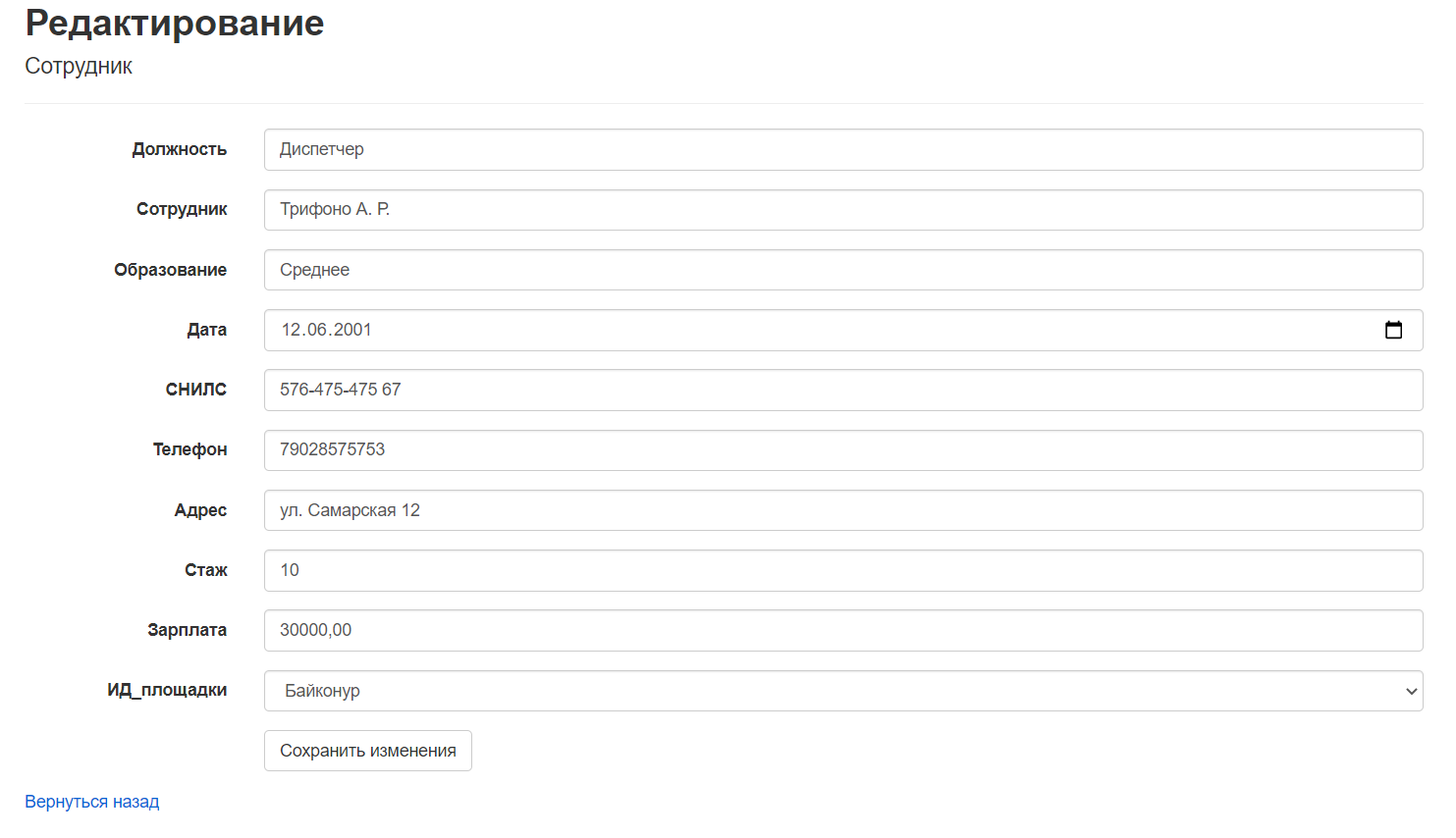


Рисунок 13 – Редактирование данных сотрудника.

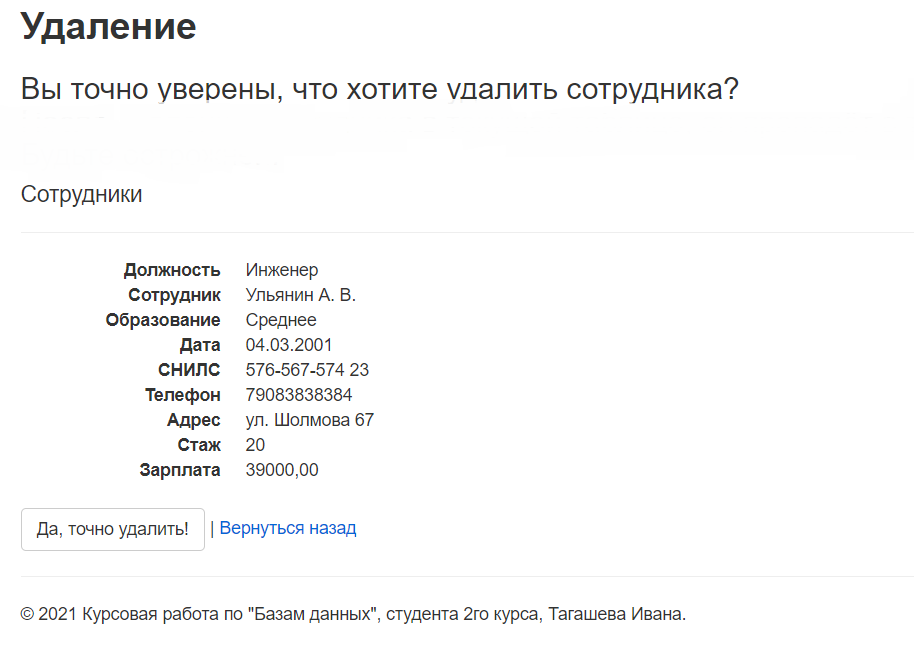


Рисунок 14 – Удаление сотрудника.

Также присутствует процедура поощрения «лучших космонавтов», повышая им рейтинг по определенным критериям, функция доступна только для администраторов (рисунок 5.12) и расчёт КМТ.

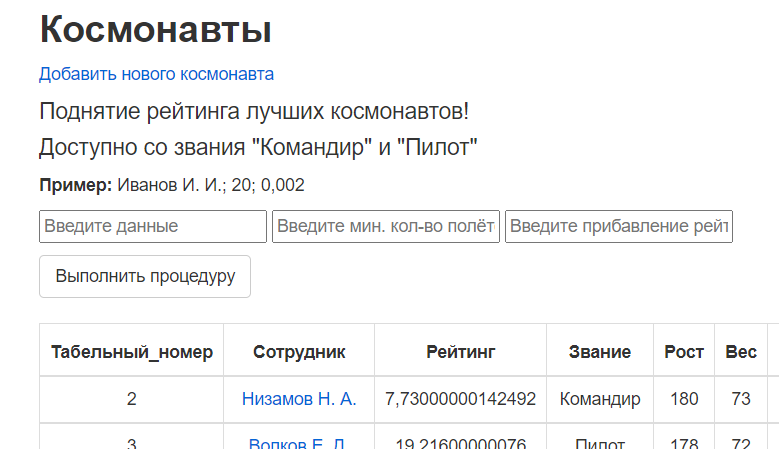
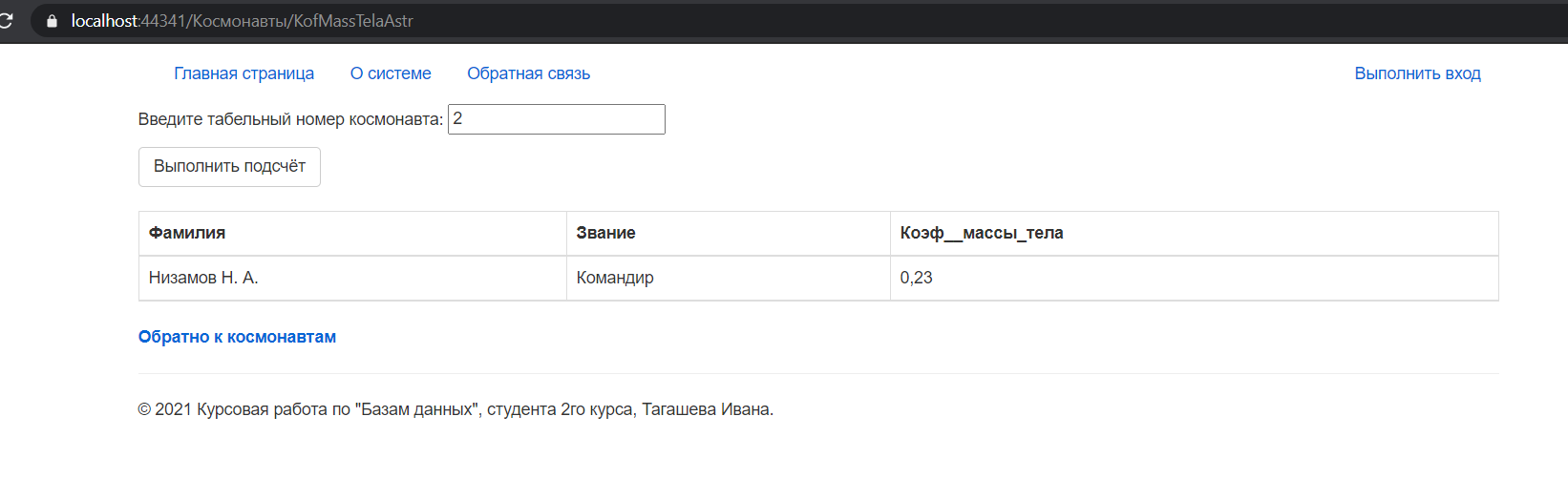


Рисунок 15 – Повышение рейтинга космонавтов и расчёт КМТ.



1. Аутентифицированному пользователю доступна информация о площадках космодрома (рисунок 5.13) и сотрудников с должностью «Космонавт» (рисунок 5.14).



Рисунок 16 – Таблица «Площадки космодрома»

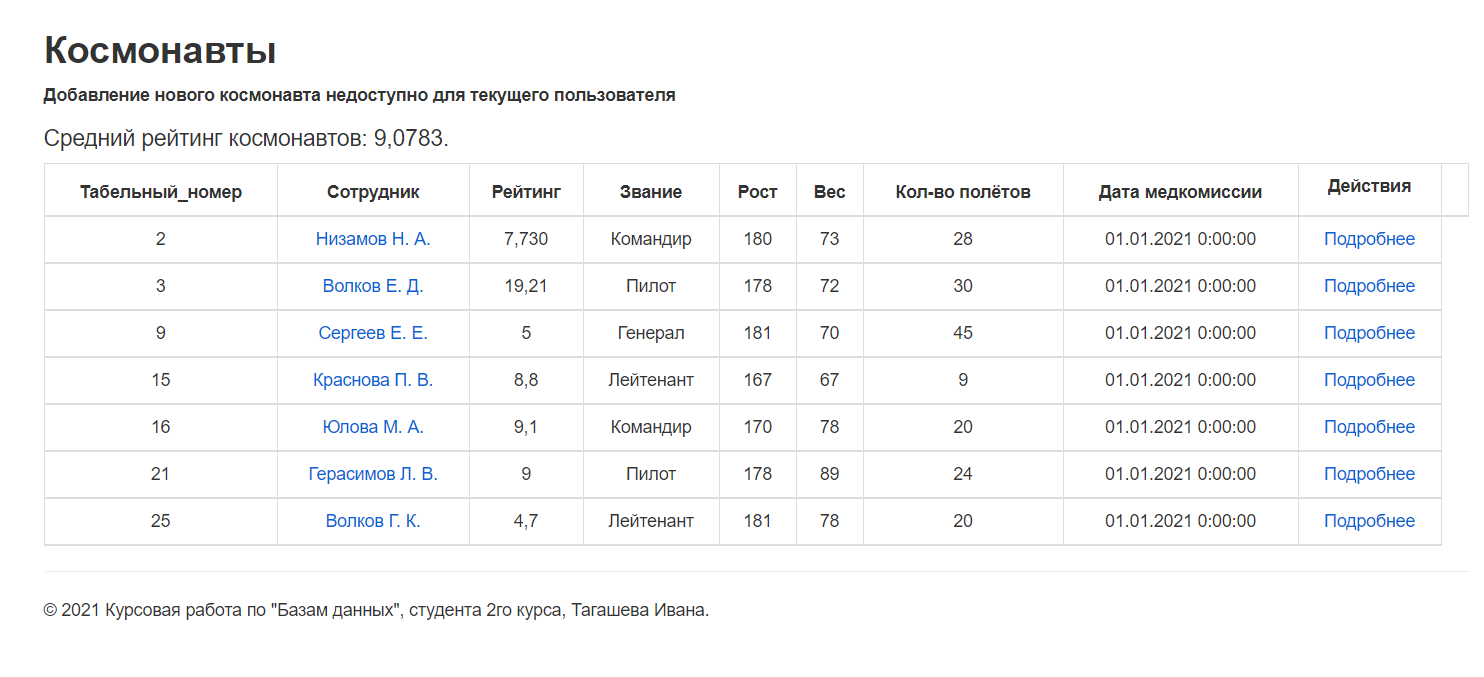


Рисунок 17 – Таблица «Космонавты»

1. В таблице «Площадки космодрома» реализована операция для фильтрации площадок по диапазону «Количества полётов» (рисунок 5.15). А в такблице «Космонавты» добавлена функция, которая показывает средний рейтинг среди космонавтов. (рисунок 5.16).

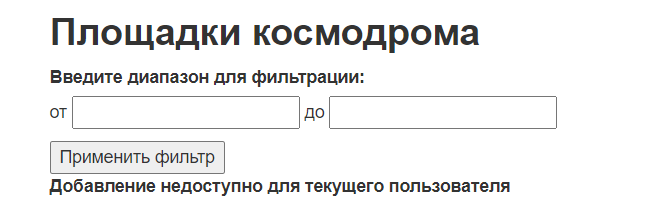


Рисунок 18 – Фильтрация.



Рисунок 19 – Средний рейтинг.

## Исключительные ситуации

1. Если при попытке аутентификации пользователь введёт неверные данные - система даст понять ему о неудачной попытке аутентификации и предложит войти в систему ещё раз.
2. Также исключительная ситуация может возникуть в том случае, когда пользователь оставит какое-либо обязательное поле в любой из таблиц незаполненным при редактировании информации или добавлении новых данных. Система не даст совершить изменения в базе данных, пока пользователь не предоставит данные корректно, и выделит поля, в которых допущена ошибка.
3. Страница ошибки будет отображена пользователю, если он попытается добавить нового сотрудника или изменить информацию о текущем так, что идентификатор или СНИЛС сотрудника будет уже занят кем-либо из сотрудников в базе данных.
4. Страница ошибки будет отображена пользователю, если он попытается удалить сотрудника, который присутствует в каком-либо заключении. Для начала нужно убрать сотрудника из всех заключений в которых фигурирует, тогда он станет свободным, и только тогда он может быть удалён из системы.
5. Страница ошибки может быть получена пользователем, когда он добавит двух одинаковых сотрудников в одно заключение, тем самым ошибка ему скажет, что сотрудники должны быть различны, либо отсутствовать.
6. Исключительная ситуация при вводе данных в процедуру, страница ошибки может появиться если данный сотрудник не является космонавтом, либо его звание не соответствует условиям.
7. Страница ошибки 400, и 404 могут появиться, если любой пользователь введёт неверную ссылку в адресную строку.
8. **Описание SQL кода работы.**

Физическая модель БД определяет способ размещения данных на носителях (устройствах внешней памяти), а также способ и средства организации эффективного доступа к ним. Поскольку СУБД функционирует в составе и под управлением операционной системы, то организация хранения данных и доступа к ним зависит от принципов и методов управления данными операционной системы.

В отличие от ранних СУБД, многие современные системы не предоставляют разработчику какого-либо выбора на этой стадии. Реально к вопросам проектирования физической модели можно отнести:

* выбор схемы размещения данных (разделение по файлам или тип RAID-массива);
* определение числа и типа индексов (например, кластеризованный или некластеризованный в случае MS SQL Server).

**Создание БД в СУБД**

Основным структурным элементом БД является таблица. В ней хранятся вводимые данные, а также структура базы (поля, их типы и свойства). Данная база состоит из нескольких таблиц, созданных в режиме конструктора.

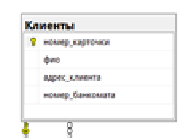
****

Рисунок 20 – Таблица Клиенты

Для создания данной таблицы, используется SQL код:

Служебный код, создаёт сам SQL Server:

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Создание таблицы «Клиенты»:

CREATE TABLE [dbo].[Клиенты](

[номер\_карточки] [int] NOT NULL,

[фио] [nvarchar](50) NULL,

[адрес\_клиента] [nvarchar](50) NULL,

[номер\_банкомата] [int] NULL,

Создаём первичный ключ, для дальнейшей связи:

CONSTRAINT [PK\_Клиенты] PRIMARY KEY CLUSTERED

(

[номер\_карточки] ASC

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

) ON [PRIMARY]

GO

Создание внешних ключей:

ALTER TABLE [dbo].[Клиенты] WITH CHECK ADD CONSTRAINT [FK\_Клиенты\_Операции выдачи] FOREIGN KEY([номер\_банкомата])

REFERENCES [dbo].[Операции выдачи] ([номер\_ПП])

GO

ALTER TABLE [dbo].[Клиенты] CHECK CONSTRAINT [FK\_Клиенты\_Операции выдачи]

GO

Создание остальных таблиц аналогично, приведенного выше примера.

**Описание функции на языке SQL:**

Для создания функции требуется перейти во вкладку «Программирование, выбрать табличные функции и открыть окно редактора SQL.

Код функции:

USE [Sherali]

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[func1] Script Date: 09.12.2021 20:05:17 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Обновление существующей функции:

ALTER FUNCTION [dbo].[func1]()

RETURNS @return TABLE ("Средняя выдача за месяц: " float)

AS

Тело функции:

BEGIN

INSERT INTO @return

Действие функции:

SELECT AVG(сумма\_выдачи) FROM [Операции выдачи]

RETURN;

END

Функция позволяет облегчить процесс для пользователя, считая среднюю выдачу за месяц для оформления документации.

# **Заключение**

Данный курсовой проект оформлен в удобном виде для пользователя, интерфейс сделан простым и понятным для использования. Все необходимые требования, написанные в задании на курсовую работу с этой работой, были соблюдены.

В ходе работы «время от времени» использовались интернет-источники, например, интернет-форумы из-за нехватки знаний и неполном понимании, касаемых нескольких пунктов заданий на определённую отметку. Несмотря на все недочеты, связанные с реализацией, в итоге от них удалось избавиться.

# **Список литературы**

1. О порядке ведения кассовых операций и правилах хранения, перевозки и инкассации банкнот и монеты Банка России в кредитных организациях на территории Российской Федерации: Положение Банка России от 24.04.2008 № 318-П [электронный ресурс] // СПС «Консультант Плюс». Версия Проф.
2. Белоцветова А. Новое положение о порядке ведения кассовых операций. Комментарии / А. Белоцветова // М.: Бухгалтерия и банки. - 2007. - № 3
3. О признаках платежеспособности и правилах обмена банкнот и монеты Банка России: Указание Банка России от 26 декабря 2006 года № 1778-У // Вестник Банка России. – 2007. - № 5
4. Галицкий В.Ю. «Все о кассовых операциях». //М.: ГроссМедиа. 2006 г.
5. Гуккаев В.Б. «Контрольно-кассовые машины и бланки строгой  
   отчётности». // М.: ЗАО «Издательский Дом «главбух». - 2002. - 208 с.
6. «Документальное оформление кассовых операций». // М.: Бератор-Пресс,  
   2002.
7. Журавлев В.Н. «Кассовые операции». // М.: Налог-инфо, 2007г.
8. «Кассовые операции и ККМ». // М.: Книга сервис, 2007 г.
9. «Кассовые операции».// М.: Статус-Кво, 2007 г.
10. Мякота В., Рудяк Ю., Кузнецов В. «Кассовые операции».// М.: ГроссМедиа, 2007 г.
11. Нестеров В.И. «Учет и ведение кассовых операций». // М.: ДИС. 2005 г..
12. В.В. Васильева «Новый реестр контрольно-кассовой техники».// М. : ГроссМедиа : РОСБУХ, 2007 г.
13. Федотов А.В. «Учет кассовых операций». // М.: Главбух**, 2008 г.**
14. Юцковская И.Д., Наличное денежное обращение в организации // М.: Финансовые и бухгалтерские консультации. 2006.-№ 2.
15. Правовая система «Консультант+».

ПРИЛОЖЕНИЕ №1

Код курсовой работы на языке SQL:

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Банки] Script Date: 09.12.2021 20:03:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Банки](

[код\_банка] [int] NOT NULL,

[название\_банка] [nvarchar](50) NULL,

[юр\_адрес] [nvarchar](50) NULL,

CONSTRAINT [PK\_Банки] PRIMARY KEY CLUSTERED

(

[код\_банка] ASC

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

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Банки] WITH CHECK ADD CONSTRAINT [FK\_Банки\_Клиенты] FOREIGN KEY([код\_банка])

REFERENCES [dbo].[Клиенты] ([номер\_карточки])

GO

ALTER TABLE [dbo].[Банки] CHECK CONSTRAINT [FK\_Банки\_Клиенты]

GO

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Банкоматы] Script Date: 09.12.2021 20:03:39 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Банкоматы](

[номер\_банкомата] [int] NOT NULL,

[адрес\_банкомата] [nvarchar](50) NULL,

[код\_банка] [int] NULL,

CONSTRAINT [PK\_Банкоматы] PRIMARY KEY CLUSTERED

(

[номер\_банкомата] ASC

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

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Банкоматы] WITH CHECK ADD CONSTRAINT [FK\_Банкоматы\_Операции выдачи] FOREIGN KEY([код\_банка])

REFERENCES [dbo].[Операции выдачи] ([номер\_ПП])

GO

ALTER TABLE [dbo].[Банкоматы] CHECK CONSTRAINT [FK\_Банкоматы\_Операции выдачи]

GO

ALTER TABLE [dbo].[Банкоматы] WITH CHECK ADD CONSTRAINT [FK\_Банкоматы\_Услуги] FOREIGN KEY([код\_банка])

REFERENCES [dbo].[Услуги] ([номер\_ПП])

GO

ALTER TABLE [dbo].[Банкоматы] CHECK CONSTRAINT [FK\_Банкоматы\_Услуги]

GO

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Инкассация] Script Date: 09.12.2021 20:03:55 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Инкассация](

[номер\_бригады] [int] NOT NULL,

[количество\_человек] [int] NOT NULL,

[опыт\_работы] [int] NULL,

[фио\_командира] [nvarchar](50) NULL,

[контактный\_телефон] [int] NULL,

CONSTRAINT [PK\_Инкассация] PRIMARY KEY CLUSTERED

(

[номер\_бригады] ASC

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

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Инкассация] WITH CHECK ADD CONSTRAINT [FK\_Инкассация\_Банкоматы] FOREIGN KEY([контактный\_телефон])

REFERENCES [dbo].[Банкоматы] ([номер\_банкомата])

GO

ALTER TABLE [dbo].[Инкассация] CHECK CONSTRAINT [FK\_Инкассация\_Банкоматы]

GO

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Клиенты] Script Date: 09.12.2021 20:04:15 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Клиенты](

[номер\_карточки] [int] NOT NULL,

[фио] [nvarchar](50) NULL,

[адрес\_клиента] [nvarchar](50) NULL,

[номер\_банкомата] [int] NULL,

CONSTRAINT [PK\_Клиенты] PRIMARY KEY CLUSTERED

(

[номер\_карточки] ASC

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

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Клиенты] WITH CHECK ADD CONSTRAINT [FK\_Клиенты\_Операции выдачи] FOREIGN KEY([номер\_банкомата])

REFERENCES [dbo].[Операции выдачи] ([номер\_ПП])

GO

ALTER TABLE [dbo].[Клиенты] CHECK CONSTRAINT [FK\_Клиенты\_Операции выдачи]

GO

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Операции выдачи] Script Date: 09.12.2021 20:04:29 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Операции выдачи](

[номер\_ПП] [int] NOT NULL,

[дата] [date] NULL,

[время] [time](7) NULL,

[комиссия] [nvarchar](50) NULL,

[сумма\_выдачи] [money] NULL,

[номер\_банкомата] [int] NULL,

[номер\_карточки] [int] NULL,

CONSTRAINT [PK\_Операции выдачи] PRIMARY KEY CLUSTERED

(

[номер\_ПП] ASC

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

) ON [PRIMARY]

GO

USE [Sherali]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Услуги] Script Date: 09.12.2021 20:04:41 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Услуги](

[номер\_ПП] [int] NOT NULL,

[дата] [date] NULL,

[фио] [nvarchar](50) NULL,

[номер\_банкомата] [int] NULL,

CONSTRAINT [PK\_Услуги] PRIMARY KEY CLUSTERED

(

[номер\_ПП] ASC

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

) ON [PRIMARY]

GO

Функция:

USE [Sherali]

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[func1] Script Date: 09.12.2021 20:05:17 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER FUNCTION [dbo].[func1]()

RETURNS @return TABLE ("Средняя выдача за месяц: " float)

AS

BEGIN

INSERT INTO @return

SELECT AVG(сумма\_выдачи) FROM [Операции выдачи]

RETURN;

END