Федеральное государственное автономное образовательное учреждение высшего образования «Московский физико-технический институт (национальный исследовательский университет)»

**Практическое задание по АСУ на тему**

**“Автоматизация учета аренды площадей клиентами компании”**

Выполнил:

студент группы B03-113И

Ошметков Степан Сергеевич

г. Жуковский, 2024

Оглавление

[Описание программы 3](#_Toc196312977)

[Описание базы данных rental.db 5](#_Toc196312978)

[Описание интерфейса и работы ПО 9](#_Toc196312979)

[Заключение 19](#_Toc196312980)

# Описание программы

Данная программа предназначена для **автоматизации учета аренды площадей клиентами компании**. Она позволяет эффективно управлять информацией о клиентах, арендуемых площадях и договорах аренды. Программа разработана с использованием языка программирования*Python* и библиотеки *Tkinter* для создания графического пользовательского интерфейса, а также СУБД *SQLite* для хранения и обработки данных.

**Основные возможности, предоставляемые программой:**

1. ***Управление клиентами:***

* Добавление, редактирование и удаление клиентов.
* Хранение информации о клиентах (ФИО, контактные данные).

1. ***Управление помещениями:***

* Добавление, редактирование и удаление информации о помещениях (номер комнаты, местоположение, площадь, статус);
* Учёт площади и стоимости помещения.
* Автоматическое определение статуса помещения (свободно/занято).

1. ***Управление договорами аренды:***

* Создание договоров с указанием клиента, помещения, сроков аренды и ежемесячной платы.
* Наличие разных статусов договоров: *активен, завершён, расторгнут.*
* Автоматическое освобождение помещений при завершении или расторжении договора.
* Возможность изменять статус активных договоров.
* Автоматическое завершение договоров по истечении времени.
* Возможность удалять завершённые и расторгнутые договоры.

1. ***Наличие сводной статистики:***

* Суммарная занятая площадь.
* Общая прибыль от активных договоров.
* Количество свободных помещений.
* Количество уникальных клиентов.
* Количество договоров по статусам.
* Максимальная, минимальная, средняя и суммарная прибыль по завершённым договорам.
* Максимальная, минимальная, средняя и суммарная площадь занятых помещений.
* Максимальная, минимальная, средняя и суммарная площадь всех помещений.

**Используемые технологии и инструменты:**

* ***Python*** - основной язык разработки.
* ***Tkinter*** - *Python*-библиотека для создания оконного интерфейса пользователя.
* ***SQLite*** - СУБД для хранения информации о клиентах, помещениях и договорах.
* ***SQL-запросы*** - запросы для выборки и обработки данных.

**Входные и выходные данные программы:**

1. **Входные данные:**

* База данных rental.db с тремя таблицами Clients, Spaces и Rentals.

1. **Выходные данные:**

* Таблица с актуальной информацией о клиентах.
* Таблица с актуальной информацией о помещениях.
* Таблица с актуальной информацией о договорах.
* Обновляемая статистика по всем ключевым показателям.
* Информационные и предупреждающие сообщения при действиях пользователя программы.

# Описание базы данных rental.db

База данных rental.db построена на *SQLite* и состоит из трёх основных таблиц, каждая из которых отражает ключевую сущность предметной области: *клиенты, помещения* и *договоры аренды*.

**Таблицы базы данных:**

1. **Clients** — информация о клиентах

| **Поле** | **Тип** | **Описание** |
| --- | --- | --- |
| id | INTEGER | Уникальный идентификатор клиента (PK) |
| name | TEXT | Имя клиента (ФИО) |
| email | TEXT | Электронная почта клиента |

**Ограничения:**

* Уникальность пары (name, email) — не допускаются дубли.

Код создание таблицы Clients:

CREATE TABLE Clients (

id INTEGER PRIMARY KEY AUTOINCREMENT,

name TEXT NOT NULL,

email TEXT NOT NULL,

UNIQUE (

name,

email

)

);

1. **Spaces** — информация о помещениях

| **Поле** | **Тип** | **Описание** |
| --- | --- | --- |
| id | INTEGER | Уникальный идентификатор помещения (PK) |
| room\_number | INTEGER | Номер комнаты (≥ 0) |
| area | REAL | Площадь помещения в м² (≥ 0) |
| location | TEXT | Местоположение (Адрес) |
| status | TEXT | Статус помещения: свободно или занято |

**Ограничения:**

* Уникальность пары (location, room\_number) — не допускаются дубли.
* Статус строго ограничен значениями ‘свободно’ и ‘занято’.

Код создание таблицы Spaces:

CREATE TABLE Spaces (

id INTEGER PRIMARY KEY AUTOINCREMENT,

room\_number INTEGER NOT NULL

CHECK (room\_number >= 0),

area REAL NOT NULL

CHECK (area >= 0),

location TEXT NOT NULL,

status TEXT CHECK (status IN ('свободно', 'занято') )

NOT NULL,

UNIQUE (

location,

room\_number

)

);

1. **Rentals** — информация о договорах аренды

| **Поле** | **Тип** | **Описание** |
| --- | --- | --- |
| id | INTEGER | Уникальный идентификатор договора (PK) |
| client\_id | INTEGER | ID клиента (внешний ключ к Clients.id) |
| space\_id | INTEGER | ID помещения (внешний ключ к Spaces.id) |
| start\_date | DATE | Дата начала аренды |
| end\_date | DATE | Дата окончания аренды |
| monthly\_price | REAL | Стоимость аренды в месяц в рублях (≥ 0) |
| status | TEXT | Статус договора: активен, завершён, расторгнут |

**Ограничения:**

* status допускает только значения ‘активен’, ‘завершён’, ‘расторгнут’.
* Внешние ключи обеспечивают целостность связей с таблицами клиентов и помещений.

Код создание таблицы Rentals:

CREATE TABLE Rentals (

id INTEGER PRIMARY KEY AUTOINCREMENT,

client\_id INTEGER NOT NULL,

space\_id INTEGER NOT NULL,

start\_date DATE NOT NULL,

end\_date DATE NOT NULL,

monthly\_price REAL NOT NULL

CHECK (monthly\_price >= 0),

status TEXT CHECK (status IN ('активен', 'завершён', 'расторгнут') )

NOT NULL,

FOREIGN KEY (

client\_id

)

REFERENCES Clients (id),

FOREIGN KEY (

space\_id

)

REFERENCES Spaces (id)

);

**Замечание:** крайне не рекомендуется редактировать данные в базе напрямую, в обход программы. База данных не содержит всех логических ограничений, реализованных в приложении. Например, программа не позволяет создать договор аренды для помещения со статусом ‘занято’, однако сама база данных такого ограничения не накладывает. Это может привести к нарушению целостности данных и некорректной работе системы.

# Описание интерфейса и работы ПО

**Запуск** программы осуществляется из консоли командой:

PS C:\Users\stepa\OneDrive\Рабочий стол\4 курс\АСУ\practical 2> python main.py

Для корректной работы приложения необходимо, чтобы файл базы данных rental.db находился в той же директории, что и программный код.

**Интерфейс начального окна**

Главное окно программы (рисунок 1) представляет собой информационную панель, отображающую сводную статистику по аренде помещений. Пользователь сразу видит ключевые показатели:

* число уникальных клиентов;
* количество договоров по статусам (активные, завершённые, расторгнутые);
* количество свободных помещений;
* общую занятую площадь помещений;
* текущую суммарную прибыль от активных договоров;
* статистику по прибыли и площади: максимальные, минимальные, средние и суммарные значения.

Ниже расположены кнопки навигации, с помощью которых пользователь может открыть отдельные окна для работы с клиентами, помещениями и договорами аренды. Также доступна кнопка "Обновить" для отображения актуальных данных в статистике.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1 – Начальное окно ПО

**Интерфейс окна ‘Клиенты’**

Окно ‘Клиенты’(рисунок 2) предназначено для просмотра и редактирования информации о клиентах, заключающих договоры аренды. В верхней части окна отображается таблица со списком клиентов, содержащая следующие данные:

* ID клиента;
* ФИО клиента;
* адрес электронной почты.

Ниже таблицы расположены кнопки управления:

* **Добавить** — открывает всплывающее окно для ввода информации о новом клиенте (рисунок 3)**;**
* **Редактировать** — позволяет изменить данные выбранного клиента (рисунок 4);
* **Удалить** — удаляет выбранного клиента (рисунок 5).
* **Обновить** — обновляет таблицу, подгружая актуальные данные из базы данных.

Форма добавления и редактирования содержит поля для ввода имени и email, с валидацией на обязательность и уникальность данных (рисунок 6).

**Внимание**: при удалении клиента все связанные с ним активные договоры становятся расторгнутыми и, как следствие, занятые помещения по договорам свободными.

**Замечание:** ID новых клиентов увеличивается независимо от удаления предыдущих записей. Например, если в базе данных были добавлены клиенты с ID 1, 2 и 3, а затем некоторые из них были удалены, то следующий добавленный клиент получит ID 4, даже если ID 2 или 3 освободились. Это стандартное поведение автоинкрементных полей в SQLite.

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 – Окно ‘клиенты’

|  |  |
| --- | --- |
| Изображение выглядит как текст, снимок экрана, Шрифт, число  Контент, сгенерированный ИИ, может содержать ошибки.  Рисунок 3 – Окно добавления клиента | Изображение выглядит как текст, снимок экрана, Шрифт, число  Контент, сгенерированный ИИ, может содержать ошибки.  Рисунок 4 – Окно редактирования клиента |
| Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение  Контент, сгенерированный ИИ, может содержать ошибки.  Рисунок 5 – Окно удаления клиента | Изображение выглядит как текст, снимок экрана, Шрифт  Контент, сгенерированный ИИ, может содержать ошибки.  Рисунок 6 – Предупреждение |

**Интерфейс окна ‘Помещения’**

Окно ‘Помещения’ (рисунок 7) предназначено для просмотра, добавления, редактирования и удаления информации о доступных площадях, сдаваемых в аренду. Интерфейс представляет собой таблицу со списком всех помещений, хранящихся в базе данных.

Каждое помещение отображается со следующими характеристиками:

* **ID** — уникальный идентификатор помещения;
* **Номер комнаты** — условный номер, присвоенный помещению;
* **Площадь (м²)** — размер помещения в квадратных метрах;
* **Расположение** — адрес или описание местоположения помещения;
* **Статус** — текущий статус помещения: *‘свободно’* или *‘занято’*.

В нижней части окна расположены кнопки управления:

* **Добавить** — открывает всплывающее окно для ввода информации о новом помещении (рисунок 8);
* **Редактировать** — позволяет изменить данные выбранного помещения (рисунок 9);
* **Удалить** — удаляет выбранное помещение из базы данных (если по нему нет активных договоров аренды) (рисунок 10);
* **Обновить** — обновляет таблицу, подгружая актуальные данные из базы данных.

Форма добавления и редактирования содержит поля для ввода номера комнаты, местоположения, площади и статуса, с валидацией на обязательность и уникальность данных, отрицательные значения площади или номера комнаты (рисунок 11).

**Внимание:** нельзя поменять статус помещения ‘занято’ на ‘свободно’, если есть активные договоры, связанные с данным помещением.

**Замечание**: при завершении или расторжении договора связанное с ним помещение **автоматически** становится свободным. Также не связанные договорами свободные помещения можно перевести в статус ‘занято’ (например, в случае ремонта помещения).

Изображение выглядит как текст, число, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 7 – Окно ‘Помещения’

|  |  |
| --- | --- |
| Рисунок 8 – Окно добавления помещения | Рисунок 9 – Окно редактирования помещения |
| Рисунок 10 – Окно удаления помещения | Рисунок 11 – Предупреждение |

**Интерфейс окна ‘Договоры аренды’**

Окно **‘Договоры аренды’** (рисунок 12) предназначено для просмотра, добавления, редактирования и удаления информации о договорах аренды, заключённых между клиентами и арендодателем. Интерфейс представляет собой таблицу со списком всех договоров, хранящихся в базе данных.

Каждый договор отображается со следующими характеристиками:

* **ID** — уникальный идентификатор договора аренды;
* **ID клиента** — уникальный идентификатор клиента, заключившего договор;
* **ID помещения** — уникальный идентификатор арендованного помещения;
* **Дата начала** — дата начала действия договора аренды;
* **Дата окончания** — дата завершения аренды;
* **Цена в месяц** — стоимость аренды в рублях за месяц;
* **Статус** — текущий статус договора: ‘активен’, ‘завершён’ или ‘расторгнут’.

В нижней части окна расположены кнопки управления:

* **Добавить** — открывает всплывающее окно для заключения нового договора аренды (рисунок 13);
* **Редактировать** — позволяет изменить данные выбранного договора (рисунок 14);
* **Удалить** — удаляет выбранный договор из базы данных (рисунок 15);
* **Обновить** — обновляет таблицу, подгружая актуальные данные из базы данных.

Форма добавления и редактирования содержит следующие поля:

* выбор ID клиента из выпадающего списка;
* выбор ID помещения из выпадающего списка (только свободные);
* дата начала аренды;
* дата окончания аренды;
* цена в месяц (неотрицательное значение);
* статус договора.

**Валидация** формы предотвращает создание договора (рисунок 16-19):

* при незаполненных полях;
* если помещение уже занято другим активным договором;
* если дата окончания раньше даты начала.

**Внимание: редактировать можно только статус договора и только у активных договоров. Удалять можно только договоры со статусами ‘**завершён’ **и ‘расторгнут’.**

**Замечание**: статус активного договора **автоматически** меняется на ‘завершён’, если дата окончания уже прошла (сравнивается с настоящим временем). При завершении или расторжении договора связанное с ним помещение **автоматически** становится свободным.

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 12 – Окно **‘Договоры аренды’**

|  |  |
| --- | --- |
| Рисунок 13 – Окно добавления договора | Рисунок 14 – Окно редактирования договора |
| Рисунок 15 – Окно удаления договора |  |
| Рисунок 16 – Предупреждение | Рисунок 17 – Предупреждение |
| Рисунок 18 – Предупреждение | Рисунок 19 – Предупреждение |

# Заключение

В ходе выполнения практической работы была разработана программная система для **автоматизации учёта аренды помещений клиентами компании**. Созданное программное обеспечение позволяет эффективно управлять следующими сущностями: *клиентами*, *помещениями* и *договорами* *аренды.*

В системе реализован удобный графический интерфейс, включающий главное окно со **сводной статистикой** по всем арендам, а также отдельные окна для **работы с клиентами**, **помещениями** и **договорами аренды**, поддерживающие операции добавления, редактирования, удаления и обновления данных.

Особое внимание было уделено **валидации** данных и **логике** взаимодействия между таблицами.

Программное обеспечение предназначено для **администраторов** или **сотрудников арендных компаний**, которым необходимо вести учёт арендаторов и площадей в удобной и понятной форме. Оно значительно **упрощает** рутинные процессы, выявляет ошибки при вводе данных и позволяет быстро получать сводную информацию о текущем состоянии арендного фонда.

Таким образом, в результате работы была реализована **универсальная и расширяемая система**, которая может быть легко адаптирована под нужды конкретной компании или организации.

Код ПО можно найти на github.com по ссылке <https://github.com/Razakys/rental-manager-practice> .