**Описание программы:**

Консольное приложение для поиска и анализа авиационных данных с функционалом поиска аэропортов по координатам, городу и рейсов по направлениям. Предоставляет удобный интерфейс для работы с информацией об аэропортах, авиакомпаниях и маршрутах.

**Основные функции приложения:**

1. Поиск аэропортов по координатам

* Введите диапазон широты и долготы
* Получите список аэропортов в этом районе

1. Поиск аэропорта по городу и стране

* Введите название города и страны
* Найдите все аэропорты в этом городе

1. Поиск рейсов по городу

* Выберите тип рейсов: вылеты, прилеты или все
* Получите список рейсов для выбранного города

1. Поиск прямых рейсов между городами

* Укажите город отправления и назначения
* Найдите все прямые рейсы между ними

**Описание базы данных (далее БД):**

БД PostgreSQL содержит три основные таблицы: airports, airlines и routes. Структура оптимизирована для быстрого поиска и анализа авиационных данных.

**Требования к работе с приложением:**

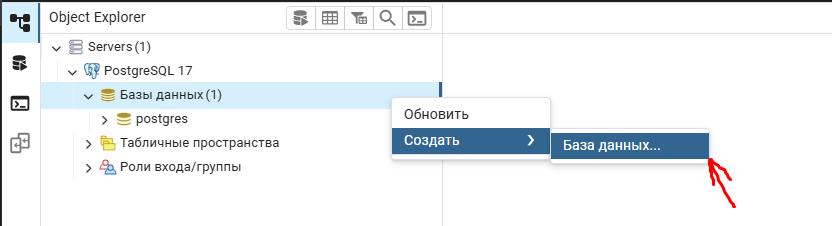
Убедитесь, что на Вашем компьютере установлены следующие программы:

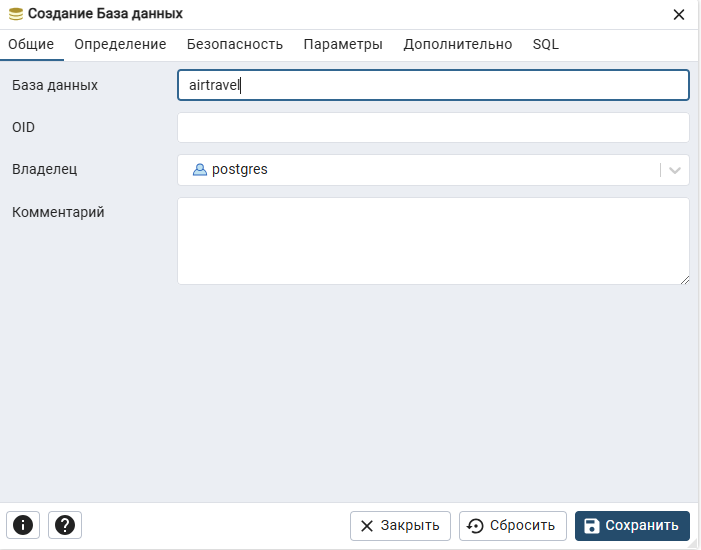
* Python 3.8+
* PostgreSQL 16+

А также путь [ваш путь]\PostgreSQL\17\bin должен быть добавлен в PATH.

***Способ развернуть базу данных (далее-БД) на своем ПК:***

1. Скачать файл backup\_airtravel.sql и поместить в удобную для Вас папку-это копия созданной БД, с которой можно работать другому пользователю.
2. Открыть PG Admin
3. Создать БД с названием "airtravel"





1. Далее открываем платформу, на которой работаем, будь то PyCharm или Spyder и т.д. и вводим следующий запрос:   
   psql -U ваше\_имя\_пользователя -d airtravel -f backup\_airtravel.sql

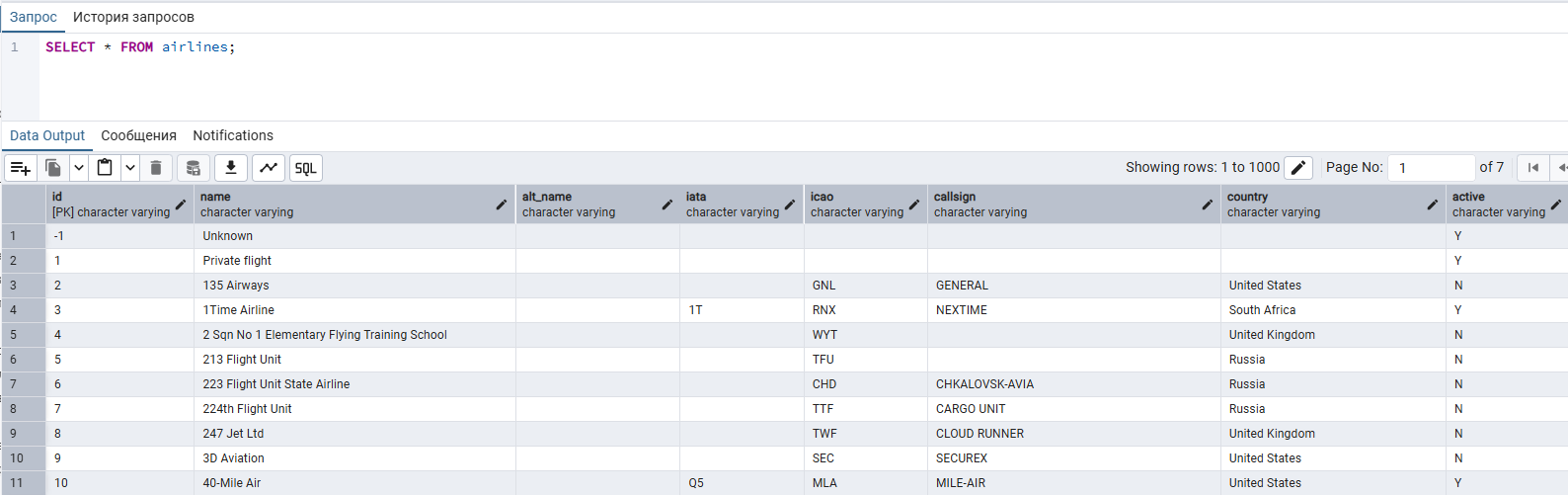
Вводим пароль пользователя и нажимаем Enter. В результате получаем БД с восстановленными таблицами.

Если все сделано верно, то в обозревателе объектов (слева) в СУБД, в БД airtravel, в схеме “public” должны отобразиться 3 таблицы: airlines, airports и routes. Для проверки импортирования данных введите следующие sql-коды:

**Для таблицы airlines:**

SELECT \* FROM airlines;

После вызова команды должна отобразиться полная таблица, содержащая всю информацию об авиаперелетах.



**Аналогично для таблицы airports:**

SELECT \* FROM airports;

**И таблицы routes:**

SELECT \* FROM routes;

***Далее переходим к работе с приложением:***

1. Продолжаем работу в платформе (для примера используется PyCharm).

Далее необходимо установить необходимые библиотеки для работы с приложением. Для этого существует файл requirements.txt, где написаны необходимые для установки пакеты. Чтобы установить все пакеты из requirements.txt в виртуальное окружение, выполните:

**pip install -r requirements.txt**

1. Открываем проект, в котором содержатся файлы airtravel.py, .env, requirements.txt.
2. Открываем файл .env, где вводим свои данные: порт, имя пользователя, пароль и т. д.
3. Далее запускаем файл airtravel.py. Если все параметры введены верно, то Вы подключитесь к БД и получите следующее сообщение:



Если допустили ошибку в данных, то у вас не получится подключиться к БД и следует перепроверить введенные параметры.

1. После успешного подключения к базе данных, перед пользователем открывается рабочее поле, где он может выбрать любой необходимый параметр для получения той или иной сортированной информации.

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

1. Для выхода из приложения нажмите 5.

Для открытия приложения (с графическим интерфейсом) требуется выполнить подключение к БД (как описано выше) и затем скачать файлы config.py, app.py, help.txt, plane2.ico и plane1.png и сохранить все в одну папку. В файле config.py изменяем параметры для подключения к БД (пароль, имя пользователя и т.д.), а затем запускаем app.py и можно работать!