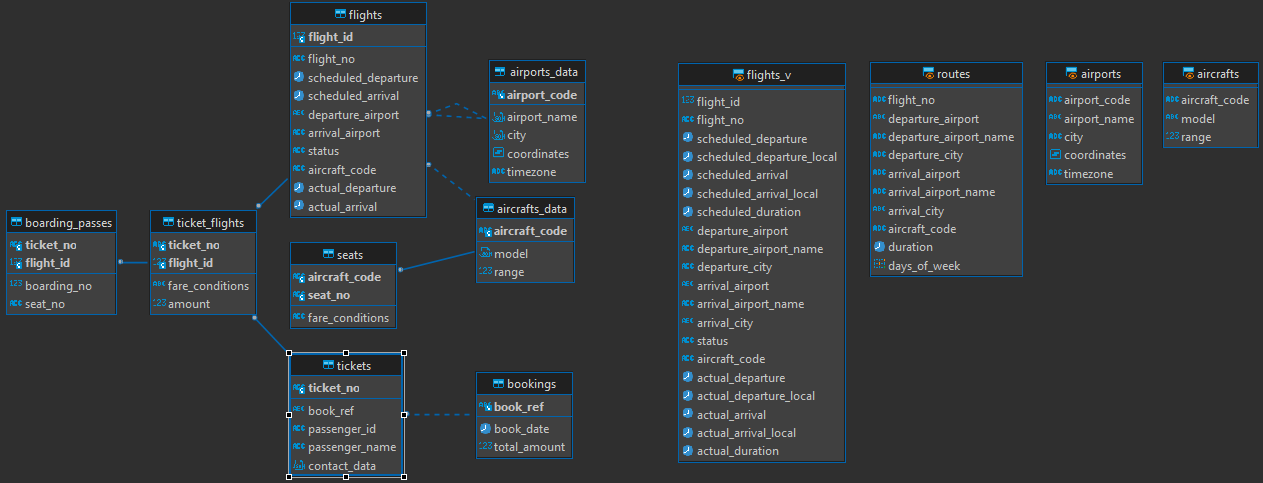
3. Построить ER-диаграмму демонстрационной базы данных (можно использовать сторонний софт для построения диаграмм, например datagrip)

4. Предложить 5 способов оптимизации БД (нарушения НФ, [оптимизация типизации](https://postgrespro.ru/docs/postgrespro/13/datatype), оптимизация [производительности](https://postgrespro.ru/docs/postgrespro/13/performance-tips), [ограничения](https://postgrespro.ru/docs/postgrespro/13/ddl-constraints#DDL-CONSTRAINTS-CHECK-CONSTRAINTS), лишние связи и т.д.):

1. Можно попробовать нормализовать базу, например вынести данные о пассажире из таблицы билета (passenger\_id, passenger\_name, contact\_data).

2. Из частых запросов по таким данным можно выявить запрос на получение билетов по имени пассажира, можно добавить составной индекс на его ФИО (или на passenger\_id если у нас будет отдельная таблица для пассажира)

3. Также можно вынести из информации об аэропортах данные о городах и их тайм зоны.

4. Можно реализовать архивацию таблиц билетов, т. к. она является самой большой в базе. Реализовать архивацию билетов, даты которых > 30 дней назад.

5. Из оптимизации типов можно отметить частое использование JSONB. В части случаев (например данные по пассажиру) можно отказаться от использования JSONB и использовать обычные строковые колонки. JSONB при вставке и обновление совершает дополнительные операции преобразования, а сам тип, является не оптимальным по хранению в памяти.