**Итоговая работа DWH**

**Описание ETL-процессов**

**1. Трансформация dim\_passengers.ktr**

* 1. Выгрузка из Postgres данных о пассажирах из таблицы bookings.tickets
  2. Проверка: нет пустых значений passenger\_name
  3. Проверка: нет пустых значений контактных данных о пассажирах:

1.4 Проверка: контактные данные содержат номер телефона:

1.5 Результаты с ошибками записываются в таблицу rejected.passengers

**2. Трансформация dim\_aircrafts.ktr**

* 1. Выгрузка из Postgres таблицы bookings.aircrafts
  2. Проверка: максимальная длительность перелета не должна быть равна нулю
  3. Проверка: нет пустых значений названия модели самолета
  4. Проверка: название модели должно содержать Boeing/Airbus/Sukhoi/Cessna/Bombardier
  5. Результаты с ошибками записываются в таблицу rejected.aircrafts

**3. Трансформация dim\_airports.ktr**

* 1. Выгрузка из Postgres таблицы bookings.airports
  2. Проверка: нет пустых значений названий аэропортов
  3. Проверка: долгота меньше или равна 180
  4. Проверка: широта меньше или равна 90
  5. Проверка: часовой пояс содержит Asia/Europe
  6. Результаты с ошибками записываются в таблицу rejected.aiports

**4. Трансформация dim\_tariff.ktr**

* 1. Выгрузка из Postgres таблицы bookings.seats
  2. Проверка: нет пустых значений класс обслуживания
  3. Проверка: класс обслуживания один из вариантов Economy/Business/Comfort
  4. Проверка: код самолета один из вариантов 773/763/SU9/320/321/319/733/CN1/CR2
  5. Результаты с ошибками записываются в таблицу rejected.tariff

**5. Трансформация fact\_flights.ktr**

* 1. Выгрузка из Postgres passenger\_id, ticket\_no из таблицыbookings.tickets
  2. Join таблицы bookings.ticket\_flights по ключу ticket\_no
  3. Join данных о дате и времени вылета, дате и времени прилета, задержки вылета, задержки прилета, код самолета, код аэропорта отправления, код аэропорта прибытия из таблицы bookings.flights по ключу flight\_id
  4. Выбор параметров для загрузки, изменение формата полей дата и время вылета, дата и время прилета
  5. Проверка: дата и время вылета – не пустое значение
  6. Проверка: дата и время прилета – не пустое значение (при условии, что дата и время вылета ненулевое, чтобы не было задвоения строк)
  7. Проверка: дата и время прибытия позже даты и времени вылета
  8. Проверка: аэропорт прибытия не такой же, как аэропорт вылета
  9. Результаты с ошибками записываются в таблицу rejected.fact\_flights