

## Домашнее задание

1.

- Функция FreeSeats
  - Я считаю что минимально удовлетворяющий уровень изоляции это read committed так как в этой функции мы не изменяем никакие данные и скорее всего она будет часто вызываться. К сожалению при read committed возможна ситуация когда кто-то параллельно бронирует часть мест которые мы хотим обработать и мы можем вернуть данные верные наполовину. Для решения этой проблемы можно использовать repeatable read, который гарантирует что никакая другая транзакция не может изменять данные, читаемые текущей транзакцией, пока та не окончена. Максимальный уровень Serializable.
- Reserve
  - Операция reserve состоит из этапов: проверка занятости места и обновления уже существующей, истекшей брони под нового пользователя или вставка в таблицу новых данных о брони. Вставить новую запись о бронировании с разными пользователями на одно и тоже место мы не можем так как ключ у таблицы Reservation(FlightId, SeatNo). Для решения случая когда две транзакции обновляют пользователя у уже существующей, истекшей записи о брони в базе достаточно уровня repeatable read. Максимальный уровень Serializable.
- ExtendReservation
  - Достаточно уровня read committed, так как вставки не происходит, только обновление существующей. Максимальный уровень Serializable.
- BuyFree
  - Аналогично с reserve. Минимальный repeatable read. Максимальный уровень Serializable
- BuyReserved
  - Аналогично с ExtendReservation. Минимальный read committed. Максимальный Serializable.
- FlightStatistics
  - Минимальный уровень read committed. Не страшно если данные будет немного неактуальные. Выбирать уровень read uncommitted не желательно, так как возможно получить совсем не актуальную статистику. Уровни Snapshot и Serializable не желательны. Запрос достаточно долгий, а при таких уровнях пользователи не смогут бронировать билеты.
- FlightStat — аналогично

- CompressSeats
  - Подходит уровень Snapshot. Если данные встали в начало, то мы не сможем смириться и повторим операцию. Если в конце и конфликтов нет, то смиримся и продолжим. Уровень Serializable так же подходит, но влечет за собой большую нагрузку на таблицу. Repeatable-read не подходит из-за блокировки таблицы, операция долгая, пользователи не смогут бронировать и покупать билеты. Repeatable-commited не дает нужных гарантий, сжатие может выдавать спецэффекты.