База данных booking.

Booking собирает данные от отелей о доступности номеров и дает возможность своим пользователям осуществлять поиск по различным запросам. БД должна быть спроектирована чтобы обеспечить поиск доступного жилья по городу, ценовому диапазону, уровню комфорта, доступных скидок, сети гостиниц и отзывам. На главную страницу обеспечивается выгрузка информации по наиболее популярным направлениям поездок, с указанием количества вариантов размещения, наиболее популярные типы размещения (отели, виллы, итп), наиболее выгодные предложения с учетом скидок и соотношения цена - качество. Кроме того БД booking содержит отзывы клиентов о отелях, а также оценку полезности этих отзывов.

Сущностями БД выступят

1. Customer и Hotel как субьекты
2. Hostel\_room, post, like как обьект. Таблицы post, like позволят фиксировать обратную связь между клиентов.
3. На сайте booking реализована электронной покупки, поэтому будет созданы booking\_session, catr\_item, order\_item, order\_detail в качестве событий. Booking\_session позволят отслеживать комнаты находящиеся в процессе бронирования, чтобы другие пользователи не видели их в качестве доступных. Данные из первых двух таблиц после оплаты будут перенесены в последние две таблицы, чтобы перегружать их данными, что ускорит работу пользователя при бронировании.
4. Для расширения возможностей базы данных в будущем и оптимизации, будет созданы справочные таблицы: room\_type, hotel\_chain, discount\_booking, gest\_payment.

Для ускорения поиска будет так называемая материализация: таблица hostel\_room будет содержать все доступные комбинации бронирования на 3 дня вперед. Например отель предоставил данные по номеру о его доступности на 1, 2, 3 января по ценам 1000, 1200, 3000 с учетом всех скидок. На booking это информация в таблице hostel\_room будет представлена как 1-2 января – 1000, 2-3 1200, 3-4 3000, 1-4 5200, 1-3 2200, 2-4 4200.

Помимо вышеуказанного, для оптимизации:

1. Тип данных для столбца «рейтинг» был выбран TINYINT
2. Данные о пользователе были разбиты на таблиц: gest, gest\_payment. Необновляемые данные были собраны в gest, обновляемые в gest\_payment. То же самое было сделано для данных о номере в отеле.
3. Индексирование по геоположению, что позволит хранить данные о близко находящиеся отели рядом. При поиске будет кластеризация по близости отелей, что может быть полезным когда клиент вынужден искать размещение большой семьи или группы в 2 и более отелей. Также будет проиндексировано имя клиента, с уетом предположения что будет много обращений по этому атрибуту.

В таблице discount\_booking собираем данные для подсчета бонусных очков клиентов от booking. Бонусные очки расчитываются как произведение disc\_persent на сумму оплаченных счетов.