

Выполнение задания «db_normalized_2».

Выполнил: Кусков Арсений

Сценарий: система бронирования в ресторане (управление бронированиями клиентов, столиками и персоналом).

Описание таблиц.

Имя таблицы: customers.

Описание: содержит данные о клиентах, включая имя и контактный телефон.

Атрибуты: customer_id (тип SERIAL, первичный ключ PK), customer_name (тип varchar(51), обязательное поле), contact_phone (тип varchar(14), обязательное поле, уникальное значение).

Ограничения: contact_phone должен быть уникальным для каждого клиента.

Имя таблицы: dining_tables.

Описание: содержит данные о столах, их вместимости и текущем статусе.

Атрибуты: table_id (тип SERIAL, PK), table_number (тип varchar(20), обязательное поле, уникальное значение), capacity (тип integer, обязательное поле), status (тип table_status, кастомный enum, обязательное поле, значение по умолчанию 'available').

Ограничения: capacity не может быть меньше или равно 0, status принимает значения строго из перечисления (available, occupied, reserved, maintenance).

Имя таблицы: staff.

Описание: содержит данные о персонале ресторана.

Атрибуты: employee_id (тип SERIAL, PK), employee_name (тип varchar(51), обязательное поле).

Ограничения: длина имени сотрудника должна быть не менее 3 символов.

Имя таблицы: reservation.

Описание: хранит данные о бронях столов клиентами через сотрудников.

Атрибуты: res_id (тип SERIAL, PK), customer_id (тип integer, внешний ключ FK к таблице customers), employee_id (тип integer, FK к таблице staff), reservation_date (тип date, обязательное поле), reservation_time (тип time, обязательное поле).

Ограничения: customer_id и employee_id ссылаются на соответствующие таблицы, дата и время брони не могут быть пустыми.

Имя таблицы: reservation_tables.

Описание: разрешает отношение многие-ко-многим между бронированиями и столами.

Атрибуты: res_table_id (тип SERIAL, PK), res_id (тип integer, FK к reservation), table_id (тип integer, FK к dining_tables).

Ограничения: комбинация res_id и table_id уникальна (UNIQUE), что исключает дублирование привязки одного стола к одному бронированию.

Описание связей.

Связь между таблицами customers и reservation: один клиент (customers) может иметь множество бронирований (reservation), но каждое бронирование принадлежит только одному клиенту (мощность один-ко-многим 1:N).

Связь между staff и reservation: один сотрудник обслуживает множество бронирований, каждое бронирование обрабатывается одним сотрудником (мощность один-ко-многим 1:N).

Связь между таблицами reservation и dining_tables: реализована через промежуточную таблицу reservation_tables, одно бронирование может включать несколько столов, и один стол может входить в разные бронирования в разное время (мощность многие-ко-многим M:N).