Отлично! Теперь перейдем к разработке логической модели данных для базы "Sushi-shop". Мы будем использовать реляционную модель и опишем таблицы, их атрибуты, первичные и внешние ключи, а также типы данных (ориентируясь на PostgreSQL, как было решено ранее). Связи между таблицами будут описаны так, как они представляются в нотации Мартина ("вороньи лапки"), то есть с указанием кардинальности (один-ко-многим, многие-ко-многим и т.д.).

**Условные обозначения для описания:**

* **PK** (Primary Key) - Первичный ключ
* **FK** (Foreign Key) - Внешний ключ
* **NN** (Not Null) - Атрибут не может быть пустым
* **UQ** (Unique) - Атрибут должен быть уникальным

**Логическая модель данных "Sushi-shop"**

**1. Таблица Должность (Positions)**

* Описание: Справочник должностей сотрудников.
* Атрибуты:
  + ID\_должности (SERIAL, PK, NN) - Уникальный идентификатор должности
  + Наименование\_должности (VARCHAR(100), NN, UQ) - Название должности (например, "Повар", "Официант", "Управляющий")
  + Ставка\_в\_час (NUMERIC(10, 2), NN) - Почасовая ставка для данной должности
  + Описание (TEXT) - Описание обязанностей

**2. Таблица Сотрудник (Employees)**

* Описание: Информация о сотрудниках.
* Атрибуты:
  + ID\_сотрудника (SERIAL, PK, NN) - Уникальный идентификатор сотрудника
  + Фамилия (VARCHAR(100), NN)
  + Имя (VARCHAR(100), NN)
  + Отчество (VARCHAR(100))
  + ID\_должности (INTEGER, FK, NN, REFERENCES Должность(ID\_должности)) - Должность сотрудника
  + Телефон (VARCHAR(20), UQ)
  + Email (VARCHAR(100), UQ)
  + Дата\_приема\_на\_работу (DATE, NN)
  + Дата\_увольнения (DATE) - Заполняется при увольнении
  + Логин (VARCHAR(50), NN, UQ) - Для входа в систему (если предполагается)
  + Хэш\_пароля (VARCHAR(255), NN) - Для входа в систему (если предполагается)

*Связь:* Должность (1) ---< (N) Сотрудник (одна должность может быть у многих сотрудников, один сотрудник имеет одну должность)

**3. Таблица Касса (CashRegisters)**  
\* Описание: Справочник кассовых аппаратов.  
\* Атрибуты:  
\* ID\_кассы (SERIAL, PK, NN) - Уникальный идентификатор кассы  
\* Номер\_кассы (VARCHAR(50), NN, UQ) - Уникальный номер или название кассы (например, "Касса 1", "Барная касса")  
\* Расположение (VARCHAR(255)) - Описание местоположения кассы

**4. Таблица Смена (Shifts)**

* Описание: Учет рабочих смен сотрудников.
* Атрибуты:
  + ID\_смены (SERIAL, PK, NN) - Уникальный идентификатор смены
  + ID\_сотрудника (INTEGER, FK, NN, REFERENCES Сотрудник(ID\_сотрудника)) - Сотрудник на смене
  + ID\_кассы (INTEGER, FK, REFERENCES Касса(ID\_кассы)) - Касса, на которой работает сотрудник (если применимо)
  + Дата\_смены (DATE, NN)
  + Время\_начала (TIMESTAMP, NN) - Точное время начала смены
  + Время\_окончания (TIMESTAMP) - Точное время окончания смены
  + Тип\_смены (VARCHAR(50)) - Например, "Полная", "Неполная", "Утренняя"

*Связи:*  
\* Сотрудник (1) ---< (N) Смена (один сотрудник может иметь много смен, одна смена принадлежит одному сотруднику)  
\* Касса (1) ---< (N) Смена (на одной кассе может быть много смен, одна смена может быть привязана к одной кассе)

**5. Таблица Склад (Warehouses)**

* Описание: Справочник складов.
* Атрибуты:
  + ID\_склада (SERIAL, PK, NN) - Уникальный идентификатор склада
  + Наименование\_склада (VARCHAR(100), NN, UQ) - Название склада (например, "Основной склад", "Кухонный склад")
  + Адрес (VARCHAR(255))
  + ID\_ответственного\_сотрудника (INTEGER, FK, REFERENCES Сотрудник(ID\_сотрудника)) - Сотрудник, ответственный за склад

*Связь:* Сотрудник (1) ---< (N) Склад (один сотрудник может быть ответственным за несколько складов (если это возможно), один склад имеет одного ответственного)

**6. Таблица КатегорияПродукта (ProductCategories)**

* Описание: Справочник категорий продуктов.
* Атрибуты:
  + ID\_категории\_продукта (SERIAL, PK, NN) - Уникальный идентификатор категории
  + Наименование\_категории (VARCHAR(100), NN, UQ) - Например, "Рыба", "Морепродукты", "Рис", "Овощи", "Соусы"

**7. Таблица ЕдиницаИзмерения (UnitsOfMeasure)**

* Описание: Справочник единиц измерения.
* Атрибуты:
  + ID\_единицы\_измерения (SERIAL, PK, NN) - Уникальный идентификатор
  + Наименование\_единицы (VARCHAR(50), NN, UQ) - Например, "кг", "г", "л", "мл", "шт"
  + Сокращение (VARCHAR(10), NN, UQ)

**8. Таблица Продукт (Products)**

* Описание: Информация о продуктах (ингредиентах).
* Атрибуты:
  + ID\_продукта (SERIAL, PK, NN) - Уникальный идентификатор продукта
  + Наименование\_продукта (VARCHAR(150), NN, UQ)
  + ID\_категории\_продукта (INTEGER, FK, NN, REFERENCES КатегорияПродукта(ID\_категории\_продукта))
  + ID\_единицы\_измерения (INTEGER, FK, NN, REFERENCES ЕдиницаИзмерения(ID\_единицы\_измерения)) - Основная единица измерения для учета
  + Описание (TEXT)
  + Минимальный\_остаток (NUMERIC(10,3)) - Для оповещений (логика оповещений на уровне приложения)

*Связи:*  
\* КатегорияПродукта (1) ---< (N) Продукт  
\* ЕдиницаИзмерения (1) ---< (N) Продукт

**9. Таблица ОстатокПродуктаНаСкладе (ProductStock)**  
\* Описание: Учет текущих остатков продуктов на конкретных складах. Обновляется триггерами или через процедуры при поступлениях и списаниях.  
\* Атрибуты:  
\* ID\_склада (INTEGER, FK, NN, REFERENCES Склад(ID\_склада))  
\* ID\_продукта (INTEGER, FK, NN, REFERENCES Продукт(ID\_продукта))  
\* Количество (NUMERIC(12, 3), NN, CHECK (Количество >= 0)) - Текущий остаток продукта на складе  
\* Средняя\_себестоимость\_единицы (NUMERIC(10,2)) - Опционально, может рассчитываться при поступлении  
\* PRIMARY KEY (ID\_склада, ID\_продукта) (Составной первичный ключ)

*Связи:*  
\* Склад (1) ---< (N) ОстатокПродуктаНаСкладе  
\* Продукт (1) ---< (N) ОстатокПродуктаНаСкладе

**10. Таблица Поставщик (Suppliers)**  
\* Описание: Справочник поставщиков.  
\* Атрибуты:  
\* ID\_поставщика (SERIAL, PK, NN) - Уникальный идентификатор поставщика  
\* Наименование\_поставщика (VARCHAR(150), NN, UQ)  
\* Контактное\_лицо (VARCHAR(100))  
\* Телефон (VARCHAR(20))  
\* Email (VARCHAR(100))  
\* Адрес (TEXT)

**11. Таблица Поступление (Receivings)**  
\* Описание: Документ о поступлении продуктов на склад.  
\* Атрибуты:  
\* ID\_поступления (SERIAL, PK, NN) - Уникальный идентификатор документа поступления  
\* Дата\_поступления (TIMESTAMP, NN, DEFAULT CURRENT\_TIMESTAMP)  
\* ID\_склада (INTEGER, FK, NN, REFERENCES Склад(ID\_склада)) - Склад, на который поступает товар  
\* ID\_поставщика (INTEGER, FK, REFERENCES Поставщик(ID\_поставщика))  
\* ID\_сотрудника\_принявшего (INTEGER, FK, NN, REFERENCES Сотрудник(ID\_сотрудника)) - Сотрудник, оформивший поступление  
\* Номер\_накладной\_поставщика (VARCHAR(100))  
\* Комментарий (TEXT)

*Связи:*  
\* Склад (1) ---< (N) Поступление  
\* Поставщик (1) ---< (N) Поступление (если поставщик обязателен, то NN)  
\* Сотрудник (1) ---< (N) Поступление

**12. Таблица ПозицияПоступления (ReceivingItems)**  
\* Описание: Детализация продуктов в документе поступления.  
\* Атрибуты:  
\* ID\_позиции\_поступления (SERIAL, PK, NN) - Уникальный идентификатор строки документа  
\* ID\_поступления (INTEGER, FK, NN, REFERENCES Поступление(ID\_поступления))  
\* ID\_продукта (INTEGER, FK, NN, REFERENCES Продукт(ID\_продукта))  
\* Количество (NUMERIC(12, 3), NN, CHECK (Количество > 0))  
\* Цена\_закупки\_за\_единицу (NUMERIC(10, 2), NN, CHECK (Цена\_закупки\_за\_единицу >= 0))  
\* Срок\_годности (DATE) - Если применимо для продукта

*Связи:*  
\* Поступление (1) ---< (N) ПозицияПоступления  
\* Продукт (1) ---< (N) ПозицияПоступления

**13. Таблица ПричинаСписания (WriteOffReasons)**  
\* Описание: Справочник причин списания продуктов/блюд.  
\* Атрибуты:  
\* ID\_причины\_списания (SERIAL, PK, NN) - Уникальный идентификатор причины  
\* Наименование\_причины (VARCHAR(150), NN, UQ) - Например, "Порча", "Истек срок годности", "Внутреннее потребление", "Приготовление блюда" (для явного списания ингредиентов, если не автоматическое)

**14. Таблица Списание (WriteOffs\_Base) - Базовая таблица для наследования**  
\* Описание: Общая информация о документе списания.  
\* Атрибуты:  
\* ID\_списания (SERIAL, PK, NN) - Уникальный идентификатор документа списания  
\* Дата\_списания (TIMESTAMP, NN, DEFAULT CURRENT\_TIMESTAMP)  
\* ID\_склада (INTEGER, FK, NN, REFERENCES Склад(ID\_склада)) - Склад, с которого происходит списание  
\* ID\_сотрудника\_ответственного (INTEGER, FK, NN, REFERENCES Сотрудник(ID\_сотрудника))  
\* ID\_причины\_списания (INTEGER, FK, NN, REFERENCES ПричинаСписания(ID\_причины\_списания))  
\* Комментарий (TEXT)

*Связи:*  
\* Склад (1) ---< (N) Списание  
\* Сотрудник (1) ---< (N) Списание  
\* ПричинаСписания (1) ---< (N) Списание

**15. Таблица СписаниеПродуктов (ProductWriteOffs) - Наследует от Списание**  
\* Описание: Детализация списания конкретных продуктов.  
\* Атрибуты:  
\* ID\_списания (INTEGER, PK, FK, NN, REFERENCES Списание(ID\_списания)) - Первичный и внешний ключ  
\* ID\_продукта (INTEGER, FK, NN, REFERENCES Продукт(ID\_продукта))  
\* Количество (NUMERIC(12, 3), NN, CHECK (Количество > 0))

*Связи:*  
\* Списание (1) --- (1) СписаниеПродуктов (каждое списание продукта связано с одним общим документом списания)  
\* Продукт (1) ---< (N) СписаниеПродуктов

**16. Таблица СписаниеБлюд (DishWriteOffs) - Наследует от Списание**  
\* Описание: Детализация списания готовых блюд (например, порча готового блюда).  
\* Атрибуты:  
\* ID\_списания (INTEGER, PK, FK, NN, REFERENCES Списание(ID\_списания)) - Первичный и внешний ключ  
\* ID\_блюда (INTEGER, FK, NN, REFERENCES Блюдо(ID\_блюда))  
\* Количество\_порций (INTEGER, NN, CHECK (Количество\_порций > 0))

*Связи:*  
\* Списание (1) --- (1) СписаниеБлюд  
\* Блюдо (1) ---< (N) СписаниеБлюд

**17. Таблица КатегорияБлюда (DishCategories)**  
\* Описание: Справочник категорий блюд.  
\* Атрибуты:  
\* ID\_категории\_блюда (SERIAL, PK, NN) - Уникальный идентификатор категории  
\* Наименование\_категории (VARCHAR(100), NN, UQ) - Например, "Роллы", "Суши", "Супы", "Горячие блюда", "Напитки"

**18. Таблица Блюдо (Dishes)**  
\* Описание: Информация о блюдах в меню.  
\* Атрибуты:  
\* ID\_блюда (SERIAL, PK, NN) - Уникальный идентификатор блюда  
\* Наименование\_блюда (VARCHAR(150), NN, UQ)  
\* ID\_категории\_блюда (INTEGER, FK, NN, REFERENCES КатегорияБлюда(ID\_категории\_блюда))  
\* Описание (TEXT)  
\* Цена\_продажи (NUMERIC(10, 2), NN, CHECK (Цена\_продажи >= 0))  
\* Время\_приготовления\_мин (INTEGER) - Ориентировочное время приготовления  
\* Фото\_URL (VARCHAR(255)) - Ссылка на фотографию блюда  
\* Активно\_в\_меню (BOOLEAN, NN, DEFAULT TRUE) - Признак, доступно ли блюдо для заказа

*Связь:* КатегорияБлюда (1) ---< (N) Блюдо

**19. Таблица СоставБлюда (DishComposition / RecipeItems)**  
\* Описание: Рецептура блюда - какие продукты и в каком количестве входят в состав.  
\* Атрибуты:  
\* ID\_состава (SERIAL, PK, NN)  
\* ID\_блюда (INTEGER, FK, NN, REFERENCES Блюдо(ID\_блюда))  
\* ID\_продукта (INTEGER, FK, NN, REFERENCES Продукт(ID\_продукта))  
\* Количество\_продукта\_на\_порцию (NUMERIC(10, 3), NN, CHECK (Количество\_продукта\_на\_порцию > 0)) - Количество в основной единице измерения продукта  
\* Единица\_измерения\_в\_рецепте (INTEGER, FK, NN, REFERENCES ЕдиницаИзмерения(ID\_единицы\_измерения)) - Единица измерения, как указано в рецепте (может отличаться от базовой для продукта, но лучше стремиться к базовой)  
\* Брутто (NUMERIC(10,3)) - опционально  
\* Нетто (NUMERIC(10,3)) - опционально  
\* Примечание (TEXT)

*Связи:* Это связующая таблица для отношения "многие-ко-многим" между Блюдо и Продукт.  
\* Блюдо (1) ---< (N) СоставБлюда  
\* Продукт (1) ---< (N) СоставБлюда

**20. Таблица СтатусЗаказа (OrderStatuses)**  
\* Описание: Справочник статусов заказов.  
\* Атрибуты:  
\* ID\_статуса\_заказа (SERIAL, PK, NN) - Уникальный идентификатор статуса  
\* Наименование\_статуса (VARCHAR(50), NN, UQ) - Например, "Новый", "Принят", "Готовится", "Готов к выдаче", "Оплачен", "Отменен"

**21. Таблица Заказ (Orders)**  
\* Описание: Информация о заказах клиентов.  
\* Атрибуты:  
\* ID\_заказа (SERIAL, PK, NN) - Уникальный идентификатор заказа  
\* Номер\_заказа\_дневной (INTEGER) - Порядковый номер заказа за день (может генерироваться приложением или триггером)  
\* Дата\_создания\_заказа (TIMESTAMP, NN, DEFAULT CURRENT\_TIMESTAMP)  
\* ID\_сотрудника\_оформившего (INTEGER, FK, NN, REFERENCES Сотрудник(ID\_сотрудника))  
\* ID\_кассы (INTEGER, FK, REFERENCES Касса(ID\_кассы)) - Касса, через которую прошел заказ  
\* ID\_статуса\_заказа (INTEGER, FK, NN, REFERENCES СтатусЗаказа(ID\_статуса\_заказа))  
\* Номер\_столика (VARCHAR(20))  
\* Итоговая\_сумма\_заказа (NUMERIC(12, 2), CHECK (Итоговая\_сумма\_заказа >= 0)) - Может рассчитываться триггером или на уровне приложения  
\* Тип\_оплаты (VARCHAR(50)) - Например, "Наличные", "Карта", "Онлайн"  
\* Комментарий\_клиента (TEXT)

*Связи:*  
\* Сотрудник (1) ---< (N) Заказ  
\* Касса (1) ---< (N) Заказ  
\* СтатусЗаказа (1) ---< (N) Заказ

**22. Таблица ПозицияЗаказа (OrderItems)**  
\* Описание: Детализация блюд в заказе.  
\* Атрибуты:  
\* ID\_позиции\_заказа (SERIAL, PK, NN) - Уникальный идентификатор строки заказа  
\* ID\_заказа (INTEGER, FK, NN, REFERENCES Заказ(ID\_заказа))  
\* ID\_блюда (INTEGER, FK, NN, REFERENCES Блюдо(ID\_блюда))  
\* Количество\_порций (INTEGER, NN, CHECK (Количество\_порций > 0))  
\* Цена\_за\_порцию\_на\_момент\_заказа (NUMERIC(10, 2), NN) - Фиксируется цена из меню на момент заказа  
\* Сумма\_по\_позиции (NUMERIC(12, 2), NN) - (Количество\_порций \* Цена\_за\_порцию\_на\_момент\_заказа)

*Связи:* Это связующая таблица для отношения "многие-ко-многим" между Заказ и Блюдо.  
\* Заказ (1) ---< (N) ПозицияЗаказа  
\* Блюдо (1) ---< (N) ПозицияЗаказа

Эта логическая модель представляет собой достаточно детализированную структуру для базы данных "Sushi-shop". На ее основе можно будет создавать физическую модель (SQL-скрипты для создания таблиц) в PostgreSQL. Важно помнить про создание индексов на часто используемые в запросах поля (особенно внешние ключи и поля, участвующие в условиях WHERE, JOIN, ORDER BY) для оптимизации производительности.