

Котова Ксения Алексеевна, 253503
Аптека

1. Employee (Сотрудник):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `first_name`: тип `VARCHAR(30)`, NOT NULL, хранит имя сотрудника
- Поле `last_name`: тип `VARCHAR(39)`, NOT NULL, хранит фамилию сотрудника.
- Поле `photo`: тип `VARCHAR(255)`, хранит путь к фотографии сотрудника.
- Поле `phone`: тип `VARCHAR(20)`, NOT NULL, хранит номер телефона.
- Поле `email`: тип `VARCHAR(255)`, NOT NULL, хранит email сотрудника.
- Поле `password`: тип `VARCHAR(255)`, NOT NULL, хранит хеш пароля сотрудника.

Связь: many-to-many (Position), many-to-many(таблица Action через таблицу Logs)

2. Client (Клиент):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `first_name`: тип `VARCHAR(30)`, NOT NULL, хранит имя клиента.
- Поле `last_name`: тип `VARCHAR(30)`, NOT NULL, хранит фамилию клиента.
- Поле `date_of_birth`: тип `DATE`, NOT NULL, хранит дату рождения клиента.
- Поле `phone`: тип `VARCHAR(20)`, NOT NULL, хранит номер телефона клиента.
- Поле `password`: тип `VARCHAR(255)`, NOT NULL, хранит хеш пароля клиент.
- Поле `email`: тип `VARCHAR(255)`, NOT NULL, хранит email клиента.

Связь: one-to-many (таблица Order), one-to-many(Review)

3. Position (Должность):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `title`: тип VARCHAR(100), уникальное значение (UNIQUE), NOT NULL, хранит название должности.
- Поле `description`: тип TEXT, хранит описание должности.

Связь: many-to-many (таблица Employee)

4. ProductType (Категория продукта):

- Поле `id`: тип INTEGER, первичный ключ (SERIAL PRIMARY KEY).
- Поле `name`: тип VARCHAR(30), обязательное для заполнения (NOT NULL), хранит название категории продукта.

Связь: one-to-many (таблица Product)

5. Product (Продукт):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `name`: тип VARCHAR(30), обязательное для заполнения (NOT NULL), хранит название продукта.
- Поле `description_id`: тип TEXT, (NOT NULL), хранит описание продукта(гель/таблетки/...).
- Поле `price`: тип DECIMAL(10, 2), обязательное для заполнения (NOT NULL), хранит цену продукта.
- Поле `product_type_id`: тип INTEGER, внешний ключ, ссылающийся на таблицу ProductType(id).
- Поле `photo`: тип VARCHAR(255), может быть пустым (NULL), хранит путь к изображению продукта.
- Поле `manufacturer_id`: тип INTEGER, внешний ключ, ссылающийся на таблицу Manufacturer(id).
- `Analog_code`: тип INTEGER, внешний ключ, ссылающийся на таблицу Analog(id).

Связь: one-to-many (таблицы ProductInstance, CartItem, OrderItem)

6. ProductInstance (Экземпляр продукта в аптеке):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `product_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Product(id)`.
- Поле `quantity`: тип INTEGER, хранит количество экземпляров продукта, имеет значение по умолчанию 1 (DEFAULT).
- Поле `pharmacy_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Pharmacy(id)` с ограничением .

7. Pharmacy (Аптека):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `street`: тип TEXT, NOT NULL, хранит название улицы.
- Поле `building`: тип INTEGER, NOT NULL, хранит номер дома.

Связь: one-to-many (таблица `ProductInstance`, `Order`)

8. Manufacturer (Производитель):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `name`: тип VARCHAR(30), хранит название производителя, (NOT NULL) обязательно для заполнения.
- Поле `phone`: тип VARCHAR(20), хранит телефонный номер производителя, NOT NULL.
- Поле `email`: тип VARCHAR(255), хранит email производителя, NOT NULL.
- Поле `country`: тип VARCHAR(100), хранит название страны производителя, (NOT NULL) обязательно для заполнения.

Связь: one-to-many (таблица `Product`)

9. CartItem (Экземпляр продукта в корзине):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `product_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Product(id)`.
- Поле `quantity`: тип INTEGER, хранит количество экземпляров продукта, имеет значение по умолчанию 1 (DEFAULT).
- Поле `cart_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Cart(id)`.

10. Order (Заказ):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `client_id`: тип INTEGER, внешний ключ, (FOREIGN KEY) связывает заказ с таблицей `Client(id)`.
- Поле `order_date`: тип TIMESTAMP, хранит дату и время создания заказа, автоматически заполняется при создании.
- Поле `total_price`: тип DECIMAL(10, 2), хранит общую стоимость заказа, вычисляемое поле.
- Поле `promo_code_id`: тип INTEGER, внешний ключ, связывает заказ с таблицей `PromoCode(id)`.
- Поле `status`: тип CHAR(1), хранит статус заказа, имеет значения ('p' - Processing, 's' - Shipped, 'd' - Delivered, 'i' - Issued), по умолчанию 'p'.
- Поле `pharmacy_id`: тип INTEGER, внешний ключ, связывает заказ с таблицей `Pharmacy(id)`.

Связь: one-to-many (таблица OrderItem)

11. Cart (Корзина):

- Поле `client_id`: тип INTEGER, внешний ключ, связывает корзину с таблицей `Client(id)`.
- Поле `total_price`: тип DECIMAL(10, 2), хранит общую стоимость продуктов в корзине, вычисляемое поле.

Связь: one-to-many (таблица CartItem), one-to-one (таблица Client)

12. OrderItem (Экземпляр продукта в заказе):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `product_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Product(id)`.
- Поле `quantity`: тип INTEGER, хранит количество экземпляров продукта, имеет значение по умолчанию 1 (DEFAULT).
- Поле `cart_id`: тип INTEGER, внешний ключ, связывает экземпляр с таблицей `Order(id)` с ограничением .

13. PromoCode (Промокод):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `code`: тип VARCHAR(10), хранит код промокода, обязательно для заполнения.
- Поле `discount`: тип DECIMAL(10, 2), хранит процент скидки, обязательно для заполнения.

Связь: one-to-many (таблица Order)

14. Review (Отзыв):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `user_id`: тип INTEGER, внешний ключ, связывает отзыв с таблицей `User(id)` с ограничением.
- Поле `rating`: тип INTEGER, хранит оценку от 1 до 5, обязательно для заполнения.
- Поле `text`: тип TEXT, хранит текст отзыва, обязательно для заполнения.
- Поле `date`: тип TIMESTAMP, хранит дату и время создания отзыва, автоматически заполняется при создании записи.

15. Action (Действие):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `action_name`: тип VARCHAR(100), уникальное значение (UNIQUE), хранит название действия.
- Поле `description`: тип TEXT, хранит описание действия.
Связь: many-to-many(таблица Employee через таблицу Logs)
- Поле `table_name`: тип VARCHAR(100), хранит имя таблицы, где было действие.

16. Logs (Действия пользователя):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `employee_id`: тип INTEGER, внешний ключ, связывает действия с таблицей **Employee(id)**, обязательно для заполнения.
- Поле `action_id`: тип INTEGER, внешний ключ, связывает действия с таблицей **Action(id)**, обязательно для заполнения.
- Поле `timestamp`: тип TIMESTAMP, хранит дату и время совершения действия пользователем, обязательно для заполнения, автоматически заполняется текущим временем.

17. Description (описание):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `name`: тип VARCHAR(100), уникальное значение (UNIQUE), хранит название (гель/таблетки/...).

Связь: one-to-many (Таблица Product)

18. Analog (Аналоги):

- Поле `id`: первичный ключ (SERIAL PRIMARY KEY), автоматически увеличивающийся.
- Поле `name`: тип VARCHAR(100), уникальное значение (UNIQUE), хранит название.
- Поле `NHM`: тип VARCHAR(100), уникальное значение (UNIQUE), хранит код для поиска аналогов.

Связь: one-to-many (Таблица Product)

Функциональные требования:

- a. Авторизация/аутентификация пользователя.
- b. Управление пользователями (CRUD).
- c. Система ролей.
- d. Журналирование действий пользователя.

Admin:

добавлять и редактировать информацию о медикаментах(код, название, описание, стоимость, фото), отделах аптеки, сотрудниках и производителях, информацию о всех продажах выбранных медикаментов, информация о конкретном лечащем средстве, о конкретном сотруднике

Employee:

Просматривать наличие товаров в различных аптеках,
Заказывать товар в подходящую клиенту аптеку

Client:

Имеет возможность купить товар, видит инфо о точках самовывоза, покупках и промокодах, инфо о категориях медикаментов и самих медикаментах с фильтрацией по цене

