

Задание

Срок выполнения задания - неделя.

Готовое задание присыпать на почту m.grishina@i-t-p.pro, в теме письма укажите вашу Фамилию и Имя , а в теле письма ссылку на Git

1. Спроектировать схему БД.

Модель данных реляционная.

Сущности

- 1.1. Номенклатура (наименование, кол-во, цена)

- 1.2. Каталог номенклатуры/Дерево категорий.

Необходимо хранить данные о категориях товара, при этом сами категории могут иметь неограниченный уровень вложенности

Пример дерева категорий:

Пример дерева категорий:

```
Бытовая техника
    Стиральные машины
    Холодильники
        однокамерные
        двухкамерные
    Телевизоры
    ...
Компьютеры
    Ноутбуки
        17"
        19"
    ...
    ...
Моноблоки
    ...
```

Схема данных категорий номенклатуры должна безболезненно позволять добавлять категории любого уровня вложенности. На этапе проектирования максимальный уровень вложенности неизвестен.

- 1.3. Клиенты (наименование, адрес)

- 1.4. Заказы покупателей. Необходимо предусмотреть возможность делать заказ из разного набора товаров.

Продумать схему БД, бизнес логику описывать не требуется.

2. Написать следующие SQL запросы:

2.1. Получение информации о сумме товаров заказанных под каждого клиента (Наименование клиента, сумма)

2.2. Найти количество дочерних элементов первого уровня вложенности для категорий номенклатуры.

2.3.

2.3.1. Написать текст запроса для отчета (view) «Топ-5 самых покупаемых товаров за последний месяц» (по количеству штук в заказах). В отчете должны быть: Наименование товара, Категория 1-го уровня, Общее количество проданных штук.

2.3.2. Проанализировать написанный в п. 2.3.1 запрос и структуру БД. Предложить варианты оптимизации этого запроса и общей схемы данных для повышения производительности системы в условиях роста данных (тысячи заказов в день).

3. Написать сервис «Добавление товара в заказ» который работает по REST-API. Метод должен принимать ID заказа, ID номенклатуры и количество. Если товар уже есть в заказе, его количество должно увеличиваться, а не создаваться новая позиция. Если товара нет в наличии то должна возвращаться соответствующая ошибка. Стек - любой фреймворк в пределах Python. Git репозиторий, контейнеризация, документация, и прочее — приветствуется.

Пример дерева категорий:

Бытовая техника	3
Стиральные машины	0
Холодильники	2
однокамерные	0
двухкамерные	0
Телевизоры	0
Компьютеры	2
Ноутбуки	2
17"	0
19"	0
Моноблоки	0

Результатом выполнения задания должна быть даталогическая схема данных, SQL запросы по пункту 2 и сервис по пункту 3.