

Практика по базам данных
ОТЧЕТ

Алимов Павел
24.M81-мм группа

Предметная область: «Молокозавод»

Реализация: PostgreSQL

Содержание

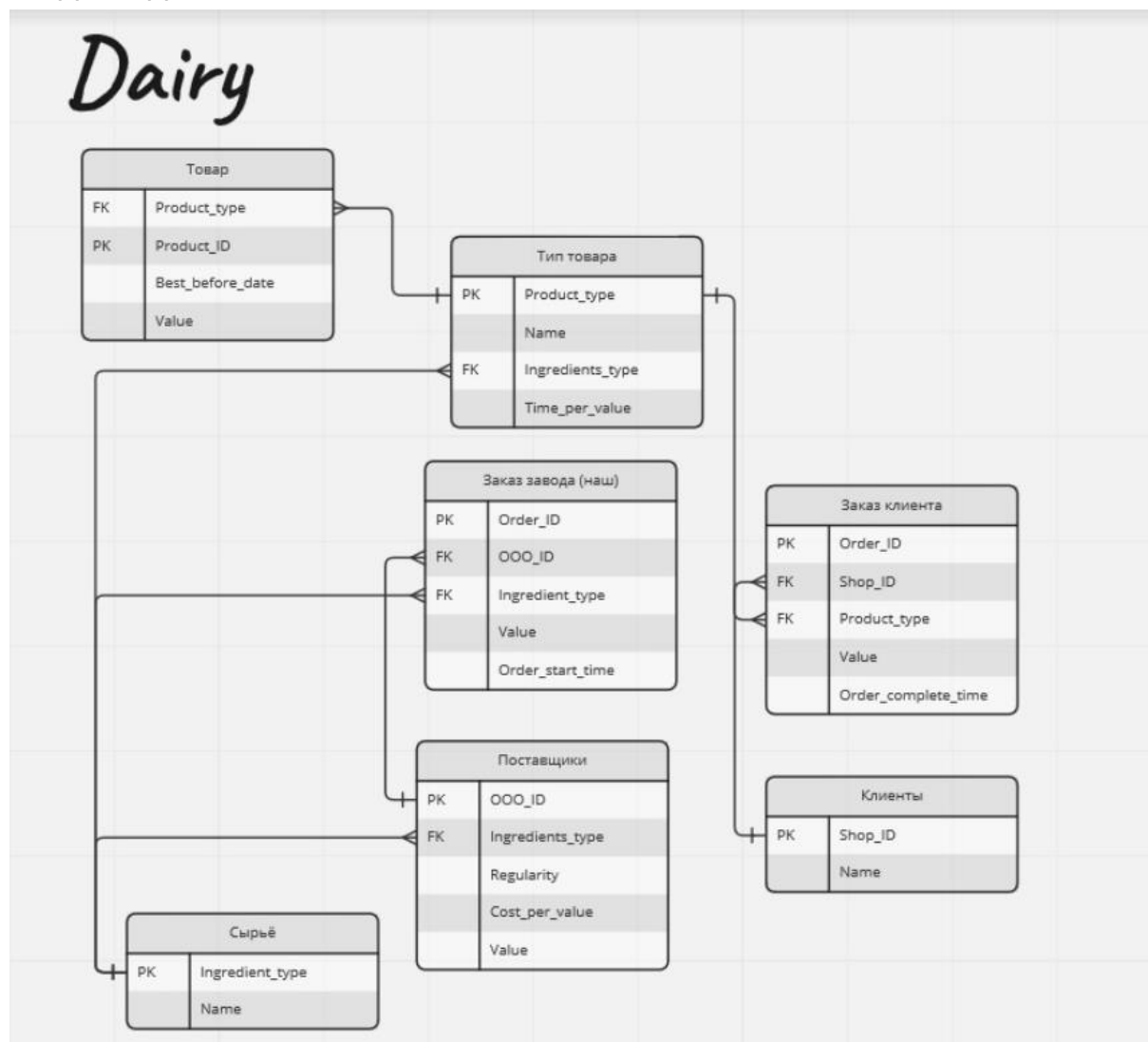
Содержание	1
Описание системы	2
Требования	2
Модель данных	2
Функциональность	3
Серверная часть	3
Клиентская часть	4
Скрипты	4
Серверная часть	4
Хранимые процедуры	4
Представления	5
Клиентская часть	6
Приложение. Скрипт создания и удаления базы данных	7

Описание системы

Требования

Система разрабатывается для внедрения на молокозавод, на котором из поставляемого поставщиками сырья производят потребительские товары. Затем эти товары закупаются клиентами завода – торговыми организациями. Поставщиками являются сельскохозяйственные компании, расположенные в разных районах области, клиентами – городские магазины. Каждый поставщик поставляет некоторое сырье определенных оптовых объемов и стоимости, с определенной регулярностью. В результате переработки получают конечные продукты разного вида, объемов, в разных упаковках. Необходимо иметь возможность хранить информацию о поставщиках и клиентах, о сырье, о конечном продукте (включая информацию о сырье, из которого он был произведен), о заводских заказах на сырье и заказах на конечные продукты – от клиентов.

Модель данных



Функциональность

Серверная часть

<i>Хранимые процедуры\функции</i>	<i>Реализация</i>	<i>Комментарии</i>
Регистрация нового магазина	add_new_shop	
Регистрация нового заказа от магазина		
Добавление нового типа продуктов		
Добавление товара на склад		
Изменение информации о поставщике		
Изменение заказа поставщику		Изменились условия (ст., объём)
Изменение заказа магазина		Изменились условия (ст., объём)
Удаление поставщика	delete_OOO	

<i>Представления</i>	<i>Реализация</i>	<i>Комментарии</i>
Количество больших групп товаров на складе по типам с истекающим сроком годности	V_old_Product	Со сроком годности кончающимся не больше, чем через 2 месяца
Количество заказов для магазинов	V_orders_per_shop	
Продукты из сыра	V_cheese_products	

Клиентская часть

№	Запрос	Использование серверной части
1	Количество сыра, который скоро испортится	V_old_Product
2	Самый срочный заказ магазина	
3	Рассчитать сколько времени займёт производство 100 единиц сметаны	
4	Названия предприятий и магазинов, с которыми налажен контакт	
5	Наш лучший клиент (больше всего заказов)	V_orders_per_shop
6	Самый быстроприготавливаемый сыр	V_cheese_products
7	Количество продукции из ингредиентов	
8	Заказы клиентов с добавленными именами товаров	
9	Товары, которых нет на складе	

Скрипты

Серверная часть

Хранимые процедуры

```
CREATE PROCEDURE add_new_shop
(
    Shop_name  VARCHAR (20)
)
LANGUAGE SQL
AS $$
    INSERT INTO Shop(Shop_ID, Shop_Name)
        VALUES ((SELECT MAX(Shop_ID) + 1 FROM Shop), Shop_name);
$$;

-- удаление
-- DROP PROCEDURE add_new_shop;

-- вызов
/*
SELECT * FROM Shop;
CALL add_new_shop('name');

SELECT * FROM Shop;
```

*/

```
CREATE PROCEDURE delete_OOO
(
    input_OOO_name    VARCHAR (20)
)
LANGUAGE plpgsql
AS $$
    DECLARE temp_OOO_ID INT;
BEGIN
    SELECT OOO.OOO_ID INTO temp_OOO_ID FROM OOO WHERE
    OOO.OOO_name = input_OOO_name;

    DELETE FROM OOO WHERE OOO.OOO_ID = temp_OOO_ID;
END;
$$;
```

```
-- удаление
-- DROP PROCEDURE delete_OOO;
```

```
-- ВЫЗОВ
```

/*

```
SELECT * FROM OOO;
SELECT * FROM Order_to_factory;
CALL delete_OOO('MolokoFerma');
```

```
SELECT * FROM OOO;
SELECT * FROM Order_to_factory;
*/
```

Представления

```
-- Количество товаров на складе по типам с истекающим сроком годности
```

```
CREATE VIEW V_old_Product(Product_type_ID, Volume)
AS
    SELECT Product.Product_type_ID, SUM(Product_volume) FROM Product
        WHERE (DATE_PART('month', CURRENT_DATE) - DATE_PART('month',
Product.Best_before_date)) <= 2
        GROUP BY Product.Product_type_ID
        HAVING SUM(Product_volume) > 20;
```

```
-- DROP VIEW V_old_Product;
```

-- Продукты из сыра

```
-----  
CREATE VIEW V_cheese_products(Product_type_ID, Product_name)  
AS  
    SELECT Product_type_ID, Product_Name FROM Product_Type, Ingredientns  
        WHERE Product_Type.Ingredient_type_ID = Ingredientns.Ingredient_type_ID  
            AND Ingredient_Name = 'Cheese';
```

-- DROP VIEW V_cheese_products;

-- Количество заказов для магазинов

```
-----  
CREATE VIEW V_orders_per_shop(Shop_Name, Order_count)  
AS  
    SELECT Shop_name, COUNT(Order_ID) AS "Count" FROM Shop  
        LEFT JOIN Client_Order ON Client_Order.Shop_ID = Shop.Shop_ID  
        GROUP BY Shop_Name;
```

-- DROP VIEW V_orders_per_shop;

Клиентская часть

-- Самый срочный заказ магазина

```
SELECT * FROM Client_Order  
    ORDER BY Client_Order.Order_complete_time ASC  
LIMIT 1  
;
```

-- Рассчитать сколько времени займёт производство 100 единиц сметаны

```
SELECT 100 * MAX(Time_per_volume) AS "Days" FROM Product_Type  
    JOIN Ingredientns ON Product_Type.Ingredient_type_ID =  
Ingredientns.Ingredient_type_ID  
    WHERE Ingredient_Name = 'Milk'  
LIMIT 10  
;
```

-- Названия предприятий и магазинов, с которыми налажен контакт

```
SELECT OOO_Name AS "Contacts" FROM OOO  
UNION  
SELECT Shop_Name FROM Shop  
LIMIT 10  
;
```

-- Количество продукции из ингредиентов

```
SELECT Ingredient_Name, Ingredientns.Ingredient_type_ID, COUNT(*) AS "Count" FROM  
Ingredientns
```

```

        JOIN Product_Type ON Product_Type.Ingredient_type_ID =
Ingredienttns.Ingredient_type_ID
        GROUP BY Ingredienttns.Ingredient_type_ID, Ingredienttns.Ingredient_Name
LIMIT 10
;

-- Заказы клиентов с добавленными именами товаров
SELECT *, (SELECT Product_Name FROM Product_Type WHERE
Client_Order.Product_type_ID = Product_Type.Product_type_ID) AS "Product name" FROM
Client_Order
LIMIT 10
;

-- Товары которых нет на складе
SELECT * FROM Product_Type
        WHERE Product_Type.Product_type_ID <> ALL(SELECT Product_type_ID FROM
Product)
LIMIT 10
;

-- Количество сыра, который скоро испортится
SELECT Volume AS "Cheese count" FROM V_old_Product
        JOIN Product_Type ON V_old_Product.Product_type_ID =
Product_Type.Product_type_ID
        JOIN Ingredienttns ON Product_Type.Ingredient_type_ID =
Ingredienttns.Ingredient_type_ID
        WHERE Ingredienttns.Ingredient_Name = 'Cheese'
LIMIT 10
;

-- Наш лучший клиент
SELECT Shop_Name FROM V_orders_per_shop
        ORDER BY Order_count DESC
LIMIT 1
;

-- Самый быстроприготавливаемый сыр
SELECT Product_Type.Product_Name FROM V_cheese_products AS V
        JOIN Product_Type ON V.Product_type_ID = Product_Type.Product_type_ID
        ORDER BY Time_per_volume ASC
LIMIT 1
;

```

Приложение. Скрипт создания и удаления базы данных

-- Создание таблиц и PK

```
CREATE TABLE Shop(
    Shop_ID INTEGER PRIMARY KEY NOT NULL,
    Shop_Name VARCHAR(20) NOT NULL
)
CREATE TABLE Ingredietns(
    Ingredient_type_ID INTEGER PRIMARY KEY NOT NULL,
    Ingredient_Name VARCHAR(20) NOT NULL
)
CREATE TABLE Product_Type(
    Product_type_ID INTEGER PRIMARY KEY NOT NULL,
    Product_Name VARCHAR(20) NOT NULL,
    Ingredient_type_ID INTEGER NOT NULL,
    Time_per_volume INTEGER NOT NULL,
    FOREIGN KEY (Ingredient_type_ID)
        REFERENCES Ingredietns (Ingredient_type_ID)
        ON DELETE CASCADE
)
CREATE TABLE Product(
    Product_ID INTEGER PRIMARY KEY NOT NULL,
    Product_type_ID INTEGER NOT NULL,
    Best_before_date DATE DEFAULT CURRENT_DATE NOT NULL,
    Product_volume INTEGER NOT NULL,

    FOREIGN KEY (Product_type_ID)
        REFERENCES Product_Type (Product_type_ID)
        ON DELETE CASCADE
)

-- providers
CREATE TABLE OOO(
    OOO_ID INTEGER PRIMARY KEY NOT NULL,
    OOO_Name VARCHAR(20) NOT NULL,
    Ingredient_type_ID INTEGER NOT NULL,
    Regularity INTEGER CHECK (Regularity > 0) NOT NULL,
    Cost_per_volume INTEGER NOT NULL,
    Delivered_volume INTEGER CHECK (Delivered_volume > 0) NOT NULL,

    FOREIGN KEY (Ingredient_type_ID)
        REFERENCES Ingredietns (Ingredient_type_ID)
        ON DELETE CASCADE
)
CREATE TABLE Client_Order(
    Order_ID INTEGER PRIMARY KEY NOT NUL,
```



```

Shop_ID INTEGER NOT NULL,
Product_type_ID INTEGER NOT NULL,
Product_volume INTEGER CHECK (Product_volume > 0) NOT NULL,
Cost INTEGER NOT NULL,
Order_complete_time DATE DEFAULT CURRENT_DATE NOT NULL,

FOREIGN KEY (Shop_ID)
    REFERENCES Shop (Shop_ID)
    ON DELETE CASCADE,
FOREIGN KEY (Product_type_ID)
    REFERENCES Product_Type (Product_type_ID)
    ON DELETE CASCADE
)

CREATE TABLE Order_to_factory(
    Order_ID INTEGER PRIMARY KEY NOT NULL,
    OOO_ID INTEGER NOT NULL,
    Product_volume INTEGER CHECK (Product_volume > 0) NOT NULL,
    Order_start_time DATE DEFAULT CURRENT_DATE NOT NULL,

    FOREIGN KEY (OOO_ID)
        REFERENCES OOO (OOO_ID)
        ON DELETE CASCADE )
-----
-- Заполнение таблиц тестовыми данными
-----

INSERT INTO Shop (Shop_ID, Shop_Name)
VALUES      (1, 'LENTA'),
            (2, 'OKEI'),
            (3, 'OBI'),
            (4, 'ASHAN'),
            (5, 'HOFF'),
            (6, 'DIKSI');

INSERT INTO Ingredietns (Ingredient_type_ID, Ingredient_Name)
VALUES      (1, 'Cheese'),
            (2, 'Milk'),
            (3, 'Butter'),
            (4, 'Curd'),
            (5, 'Sour cream');

INSERT INTO Product_Type (Product_type_ID, Product_Name, Ingredient_type_ID,
Time_per_volume)
VALUES      (1, 'Russian cheese', 1, 3),
            (2, 'Holland cheese', 1, 5),
            (3, 'Lamber cheese', 1, 12),
            (4, 'Granular curd', 4, 3),
            (5, 'Curd', 4, 2),
            (6, 'Packaged sour cream', 5, 1),

```

```

        (7, 'Ice cream', 2, 4),
        (8, 'Yogurt', 2, 6);
INSERT INTO Product (Product_ID, Product_type_ID, Product_volume, Best_before_date)
VALUES      (1, 1, 10, '2023-09-04 00:00'),
            (2, 1, 5, '2022-12-12 00:00'),
            (3, 3, 100, '2023-01-01 00:00'),
            (4, 7, 2, '2023-09-04 00:00'),
            (5, 8, 10, '2022-03-12 00:00');
INSERT INTO OOO (OOO_ID, OOO_name, Ingredient_type_ID, Regularity,
Delivered_volume, Cost_per_volume)
VALUES      (1, 'SirFerma', 1, 7, 1, 1000),
            (2, 'MolokoFerma', 2, 2, 100, 100),
            (3, 'MasloFerma', 3, 12, 50, 200),
            (4, 'TvorogFerma', 4, 4, 100, 300),
            (5, 'SmetanaFerma', 5, 4, 100, 400);
INSERT INTO Client_Order(Order_ID, Product_type_ID, Shop_ID, Cost, Product_volume,
Order_complete_time)
VALUES      (1, 1, 1, 1500, 1, '2022-12-24T00:00:00'),
            (2, 2, 2, 500, 10, '2022-12-24T00:00:00'),
            (3, 3, 3, 500, 10, '2022-12-24T00:00:00');
INSERT INTO Order_to_factory(Order_ID, OOO_ID, Order_start_time, Product_volume)
VALUES      (1, 1, '2022-12-04 00:00', 10),
            (2, 2, '2022-12-05 00:00', 20),
            (3, 3, '2022-12-06 00:00', 30);

```

```

-----
-- Создание индексов
-----

```

```

CREATE INDEX OOO_ingredients_idx
ON OOO (Ingredient_type_ID);
CREATE UNIQUE INDEX shops_id_idx
on Shop (Shop_ID);
-----

```

```

-- Удаление таблиц
-----

```

```

/*
DROP TABLE Order_to_factory;
DROP TABLE Client_Order;
DROP TABLE Shop;
DROP TABLE Product;
DROP TABLE OOO;
DROP TABLE Product_Type;
DROP TABLE Ingredietns;
*/

```