**ЛАБОРАТОРНАЯ РАБОТА**

**Тема. Создание представлений**

Общая информация

Представление VIEW – это **именованный хранимый запрос** в базе данных.

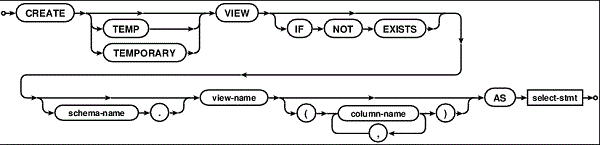
Каждый раз, когда мы обращаемся к VIEW, СУБД выполняет этот запрос SELECT, а следом за ним, она выполняет наш запрос.

**Использование представлений в SQL и реляционных базах данных**

После создания представлений, их можно использовать так же, как таблицы. То есть выполнять запросы к ним, фильтровать и сортировать данные, объединять одни представления с другими. С одной стороны это очень удобный способ хранения частоприменяемых сложных запросов (как в нашем примере).

Но следует помнить, что представления – это не таблицы, то есть они не хранят данные, а лишь извлекают их из других таблиц. Отсюда, во-первых, при изменении данных в таблицах, результаты представления так же будут меняться. А во-вторых, при запросе к представлению происходит поиск необходимых данных, то есть производительность СУБД снижается. Поэтому злоупотреблять ими не стоит.

**SQL синтаксис создания VIEW в базах данных**

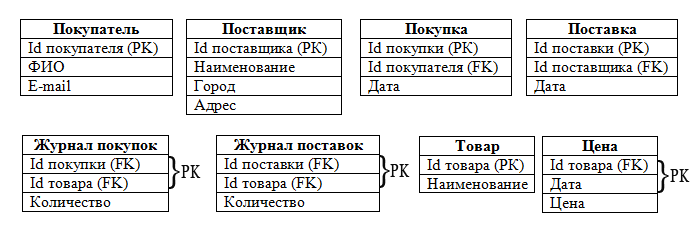


* Используем **CREATE** для создания VIEW – это такой же объект базы данных, как и таблица.
* Указываем, что хотим создать представление при помощи ключевого слова **VIEW**.
* Если вы не уверены, что создаете представление с уникальным именем и не хотите возникновения ошибок при создании VIEW в базе данных, то можете использовать ключевую фразу **IF NOT EXISTS.**
* Далее вам необходимо указать **имя представления, которое должно быть уникальным**, в качестве имени можно использовать квалификатор, в том случае, если вы работаете с несколькими базами данных и хотите быть уверенным в том, что создаете VIEW для нужной базы данных.
* После имени представления идет ключевое слово **AS** и запрос **SELECT**, который будет храниться в файле базы данных и к которому СУБД будет обращаться по тому имени, которое вы указали при создании VIEW.

**SQL синтаксис удаления VIEW из базы данных** 

**ЗАДАНИЕ 1**

1. Установите связь с БД Интернет-магазин (создана в ЛР 23)



1. Создайте представление, выводящее код и наименование Поставщика, код и даты поставок.

**Указания:**

create view vendors\_incoming As select \* from **vendors natural join incoming on vendors .id\_vendor = incoming.id\_vendor;**

CREATE VIEW vendors\_incoming  AS SELECT a.id\_vendor, name, city , address , id\_incoming, date\_incoming

FROM vendors AS a JOIN incoming AS b ON a.id\_vendor=b.id\_vendor;

- Название представления придумайте сами.

- В запросе используйте соединение двух таблиц: **vendors, incoming**.

1. Убедитесь, что такой объект появился в БД
2. Создайте новое представление, выводящее код и наименование Поставщика, код и даты поставок, код и количество поставленных им товаров.

**Указания:**

create view v\_i\_m

As select id\_vendor ,vi.id\_incoming , id\_product , quantity , name , city , address , date\_incoming

from vendors\_incoming as vi

join **magazine\_incoming as mi on vi.**id\_incoming=mi.id\_incoming;

- Название представления придумайте сами.

- В запросе используйте соединение трех таблиц: **vendors, incoming, magazine\_incoming.**

1. Убедитесь, что такой объект появился в БД
2. Составьте запрос, использующий в качестве источника сформированное представление, выводящий имя поставщика, код товара и количество поставленных товаров.

select name , id\_product , quantity

from v\_i\_m;

1. Составьте запрос, использующий в качестве источника сформированное представление, выводящий имя поставщика и общее количество поставленных товаров.

select name , sum(quantity) as q

from v\_i\_m group by id\_vendor,name ;

**Указание:** используйте в запросе группировку и агрегатную функцию sum.

**Результат (будет выведено таблицей):**

('БХВ-Петербург', 60),

('Вильямс', 74),

('Дом печати', 86)

1. Напишите новое представление, добавив цену поставленных товаров.

create view vim\_prices

as select id\_vendor , id\_incoming , v\_i\_m.id\_product , date\_price\_changes , price ,quantity , name , city , address, date\_incoming

from v\_i\_m join **prices on** v\_i\_m.id\_product = **prices.**id\_product;

**Указание:** добавьте соединение с таблицей **prices**.

1. Напишите запрос, использующий в качестве источника сформированное представление, выводящий имя поставщика и сумму денег, на которую поставлены товары.

select name , sum(prices\* quantity) from vim\_prices group by id\_vendor,name;

**Указание:** используйте группировку и агрегатную функцию sum от выражения prices\* quantity.

1. Напечатайте результат.

**Результат (буде выведен таблицей):**

('БХВ-Петербург', 5750.0),

('Вильямс', 8060.0),

('Дом печати', 7664.0)

1. Создайте еще одно представление, в котором выводятся названия поставленных продуктов и их количество.

create view vim\_products as

select id\_vendor , id\_incoming , v\_i\_m.id\_product , products.name as name\_pr , author, quantity , v\_i\_m.name , city , address, date\_incoming

from v\_i\_m join products on v\_i\_m.id\_product =products.id\_product ;

1. Убедитесь, что новый объект появился в БД.
2. Составьте запрос, использующий в качестве источника сформированное представление, выводящий все формируемые им строки.

**Результат (будет выведен таблицей):**

('Стихи о любви', 10),

('Собрание сочинений, том 2', 5),

('Собрание сочинений, том 3', 7),

('Русская поэзия', 10),

('Машенька', 10),

('Доктор Живаго', 8),

('Мертвые души', 8),

('Три сестры', 8),

('Беглянка', 8),

('Наши', 10),

('Приглашение на казнь', 10),

('Лолита', 6),

('Темные аллеи', 10),

('Дар', 10),

('Идиот', 10),

('Братья Карамазовы', 10),

('Ревизор', 10),

('Гранатовый браслет', 10),

('Сын вождя', 10),

('Эмигранты', 10),

('Горе от ума', 10),

('Анна Каренина', 10),

('Повести и рассказы', 10),

('Антоновские яблоки', 10)

create view products\_prices as

select name\_pr, sum(quantity \*price) from vim\_products join prices on

vim\_products.id\_product = prices.id\_product group by vim\_products.id\_product , name\_pr;

1. Составьте запрос, использующий в качестве источника сформированное представление, выводящий название поставленной книги и общую стоимость.

**Результат:**

('Анна Каренина', 900.0),

('Антоновские яблоки', 900.0),

('Беглянка', 680.0),

('Братья Карамазовы', 700.0),

('Горе от ума', 1000.0),

('Гранатовый браслет', 1300.0),

('Дар', 1050.0),

('Доктор Живаго', 680.0),

('Идиот', 1050.0),

('Лолита', 474.0),

('Машенька', 1100.0),

('Мертвые души', 1200.0),

('Наши', 950.0),

('Повести и рассказы', 750.0),

('Приглашение на казнь', 1000.0),

('Ревизор', 650.0),

('Русская поэзия', 1000.0),

('Собрание сочинений, том 2', 650.0),

('Собрание сочинений, том 3', 630.0),

('Стихи о любви', 1000.0),

('Сын вождя', 850.0),

('Темные аллеи', 490.0),

('Три сестры', 1120.0),

('Эмигранты', 1350.0)

1. Оформите предыдущий запрос как новое представление.

select name\_pr , sum from products\_prices where sum = (select max(sum) from products\_prices);

1. Используйте его, чтобы вывести название товара с наибольшей выручкой.

**Указание:** используйте подзапрос для нахождения максимума общей стоимости товара.

**Результат:**

('Эмигранты', 1350.0)

**ЗАДАНИЕ 2**

**Составьте и выполните запросы без использования представлений**

**Указание:** Во всех задачах считайте, что цена на каждый товар была установлена только один раз.

2.1 Вывести минимальную, максимальную и среднюю цену на книги в магазине.

select

from products join prices on products.id\_product= prices.id\_product

select min(price) , max(price) , avg(price) from prices;

2.2 Выбрать идентификаторы (id\_incoming) тех поставок, которые осуществлялись поставщиком "Дом печати" (id=2).

select id\_incoming from incoming where id\_vendor = 2;

2.3 Из таблицы Журнал поставок (magazine\_incoming) надо выбрать товары (id\_product) и их количества (quantity), которые осуществлялись в тех поставках, которые осуществлялись поставщиком "Дом печати" (id=2). (Используйте запрос из задачи 2.2 как вложенный).

select id\_product , quantity

from magazine\_incoming where id\_incoming in (select id\_incoming from incoming where id\_vendor = 2);

2.4 Добавьте в результирующую таблицу задачи 2.3 цены на найденные товары (понадобится соединение таблиц Журнал поставок (magazine\_incoming) и Цены (prices) по столбцу id\_product).

2.5 Добавьте в результирующую таблицу задачи 2.4 вычисляемый столбец summa, в котором будет вычислено произведение столбцов quantity (количество) и price (цена).

2.6 Внесите изменения в предыдущий запрос и посчитайте для каждого продукта, на какую сумму был он поставлен поставщиком "Дом печати" (id=2) в результате всех произведенных им поставок.

.