

1. ***Вывести количество товаров для каждой категории.***  
   SELECT p.category\_id COUNT(\*)

FROM products AS p

GROUP BY p.category\_id

1. ***Вывести список пользователей, которые не оплатили хотя бы 1 заказ полностью.***  
     
   SELECT users.id, users.login FROM users  
   JOIN orders ON users.id=orders.users\_id AND orders.status=’Partly\_paid’
2. ***Удалить товары, которые ни разу не были куплены (отсутствуют информация в таблице basket).***

DELETE FROM products

WHERE products.id NOT IN (

SELECT basket.product\_id

FROM basket

)

1. ***Вывести заказы, которые оплачены только частично*.**  
   SELECT orders.id, SUM(payments\_orders.value) FROM orders   
   JOIN payments\_orders ON orders.status!= 'Partly\_paid'

AND payments\_orders.value>'0.00'

AND payments\_orders.order\_id=orders.id

GROUP BY orders.id

1. ***Вывести для каждого покупателя количество его заказов по статусам.***

SELECT u.id

COUNT(o.\*) filter (

WHERE o.status = 'N' ) AS n\_count,

count(o.\*) filter (

WHERE o.status = 'D' ) AS d\_count,

count(o.\*) filter (

WHERE o.status = 'P' ) AS p\_count,

count(o.\*) filter (

WHERE o.status = 'F' ) AS f\_count,

count(o.\*) filter (

WHERE o.status = 'C') AS c\_count,

FROM users u

RIGHT JOIN orders o ON o.user\_id = u.id

GROUP BY u.id;

1. **Вывести средний чек для выполненных заказов (status='F').**

SELECT AVG(orders.price::numeric::float8) AS srCheck FROM orders

WHERE orders.status='F '

1. ***Вывести топ-10 самых продаваемых товаров по суммарной прибыли (заказы для этих товаров должны быть полностью оплачены).***

*SELECT p.\*,*

*SUM(b.price) AS basket\_value*

*FROM products p*

*RIGHT JOIN basket b ON b.product\_id = p.id,*

*(*

*SELECT o.id, o.price*

*SUM(p\_o.* *value) AS paid\_value*

*FROM*

*orders o*

*RIGHT JOIN payment\_orders p\_o ON p\_o.order\_id = o.id*

*GROUP BY*

*o.id, p\_o.value*

*) AS paid\_orders*

*JOIN basket ON b.order\_id = paid\_orders.id*

*WHERE paid\_orders.price <= paid\_orders.paid\_value*

*GROUP BY*

*p.id*

*ORDER BY*

*basket\_value DESC*

*LIMIT 10;*

1. ***Вывести список товаров, которые лежат пока только в корзине и не привязаны к заказу, и их количества не хватает на складе для продажи (считается, что товар списывается со склада только тогда, когда заказ будет доставлен).***

*SELECT p.\**

*FROM products p*

*RIGHT JOIN basket b ON b.product\_id = p.id and b.order\_id IS NULL*

*GROUP BY p.id*

*HAVING SUM(b.quantity)<SUM(p.quantity)*

1. ***Вывести список пользователей, которые "бросили" свои корзины (не оформили заказ) за последние 30 дней.***SELECT \* FROM users

JOIN orders ON orders.user\_id=users.id AND orders.status=’N’

JOIN payments\_orders ON orders.id=payments\_orders.order\_id

AND payments\_orders.create\_at>=(SELECT NOW()-INTERVAL '30' DAY)

ORDER BY users.id

1. ***Добавить скидку 10% на все товары, которые покупались (статус заказов "P" или "F") не более 10 раз.***

UPDATE products

SET discount=discount\* 1.1

WHERE products.id IN

(SELECT basket.product\_id

FROM basket

RIGHT JOIN orders ON basket.order\_id=orders.id

WHERE orders.status SIMILAR TO 'P|F'

GROUP BY basket.product\_id

HAVING COUNT(orders.\*) <= 10)

1. ***Вывести количество заказов оплаченных полностью с внутреннего счета.***SELECT COUNT(\*) AS pau\_inside FROM orders

JOIN payments\_orders

ON orders.id=payments\_orders.order\_id

WHERE

orders.status=’P’ AND

payments\_orders.payment\_id IS NULL AND

payments\_orders.from\_account IS TRUE

GROUP BY orders.id

1. ***Сделать скидку 50% (без доставки, только на товары) на новые заказы (статус "N") для VIP-пользователей.***

*UPDATE orders*

*SET discount = discount\*1.5*

*WHERE orders.user\_id IN*

*(SELECT user.id FROM users*

*RIGHT JOIN accounts ON accounts.user\_id = users.id*

*WHERE*

*accounts.is\_vip = true*

*GROUP BY users.id*

*) AND orders.status = 'N';*

1. ***Вывести самую популярную доставку и самый популярный способ оплаты (результат из 2 записей)***

*SELECT*

*delivery\_top.\*,*

*payment\_top.\**

*FROM*

*(*

*SELECT d.\*,*

*COUNT(d\_o.\*) AS orders\_count*

*FROM delivery d*

*LEFT JOIN delivery\_orders d\_o ON d\_o.delivery\_id = d.id*

*ORDER BY*

*orders\_count DESC*

*GROUP BY d.id*

*LIMIT 1*

*) AS delivery\_top,*

*(*

*SELECT p.\*,*

*COUNT(p\_o .\*) AS orders\_count*

*FROM payments p*

*LEFT JOIN payments\_orders p\_o ON p\_o.payment\_id = p.id*

*ORDER BY*

*orders\_count DESC*

*GROUP BY p.id*

*LIMIT 1*

*) AS payment\_top;*

1. ***Удалить пустые категории.***

DELETE FROM categories c

WHERE c.id NOT IN (SELECT products.category\_id FROM products)

1. ***Вывести список пользователей, которые были оплачены полностью не более чем через час, с момента добавления первого товара в корзину.***

*SELECT o.\*,*

*MIN(b.created\_at) AS basket\_first,*

*MAX(p\_o.created\_at) AS payment\_last,*

*SUM(p\_o.value) AS paid\_value*

*FROM orders o*

*LEFT JOIN payments\_orders p\_o ON p\_o.order\_id = o.id*

*LEFT JOIN basket b ON b.order\_id = o.id*

*WHERE*

*orders.price = (SELECT SUM(payments\_orders.value) FROM payments\_otders)*

*AND (*

*SELECT*

*date\_part*

*( 'hour',*

*(SELECT MAX(payments\_orders.create\_at)-MIN(basket.create\_at) FROM payments\_orders, basket)*

*)*

*) <= 1*

*GROUP BY o.id;*

1. ***Вернуть все деньги пользователей на внутренний счет для заказов, которые были оплачены (как внутренним счетом, так и некоторым способом оплаты) и были отменены (status = "С").***

*WITH acc AS (*

*SELECT*

*SUM(o.price) AS price,*

*SUM(p\_o.value) AS paid\_value*

*FROM orders o*

*LEFT JOIN payments\_orders p\_o ON p\_o.order\_id = o.id*

*WHERE*

*o.status = 'C'*

*GROUP BY o.id*

*HAING SUM(price) <= SUM(p\_o.value)*

*)*

*UPDATE accounts*

*SET*

*value = value + price*

*FROM acc ;*