Министерство образования Республики Беларусь

УО «Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №10**

По дисциплине: “Языки программирования”

Тема: “Создание простых запросов на выборку”

**Выполнил**: студент 2 курса группы ПО-7 Якимчик Д.В.

**Проверил:** Бойко Д.О.

Брест 2021

### Цель работы:

### Изучение простых запросов на выборку.

**Задание для выполнения:**

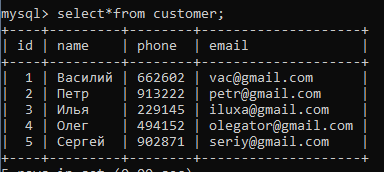
При выполнении лабораторной работы необходимо для заданной предметной области средствами MySQL:

1. Изучить теоретический материал
2. Создать не менее пяти простых запросов на выборку(придумать запросы и согласовать с преподавателем)
3. Для заданной предметной области построить многотабличный запрос на выборку с использованием объединения
4. Для заданной предметной области построить запрос на выборку, содержащий вложенный запрос
5. Оформить отчет

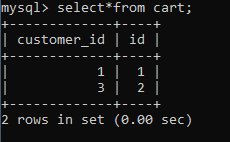
**Ходработы:**

**Из предыдущей лабораторной работы имеем следующие таблицы:**

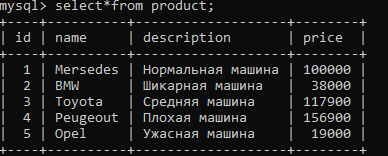
**Customer**

****

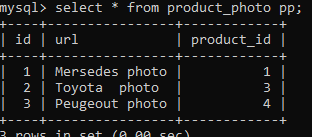
**Cart**



**Product**



**Product\_photo**

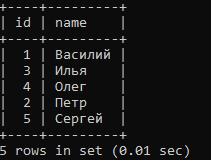
****

**Cart\_ptoduct**

**Начнем с простых запросов:**

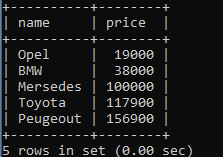
**Выведем конкретное поле (имена и id) из таблицы customerс сортировкой по имени:**

select id, name from customer order by name;

****

**Отобразим товар и его цену по возрастанию:**

select name, price from product order by price;

****

**Сделаем удалениt cnjk,wf таблицы продуктов в корзине:**

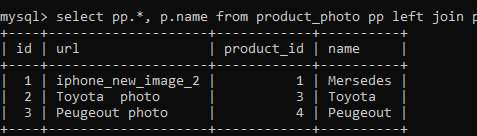
ALTER TABLE cart\_product DROP COLUMN cart\_id;

DELETE FROM name WHERE id>1

**Перейдем к запросам посложнее (объединения, вложенные запросы):**

**Получим новую таблицу с названием продукта и его фото используя ALIASES:**

select pp.\*, p.name from product\_photo pp left join product p on p.id=pp.product\_id;



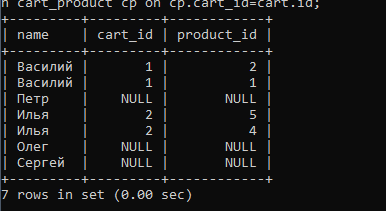
Добавим вывод цены от определённого и до определённого значения

SELECT \* FROM customer

WHERE price BETWEEN 50000 AND 100000;

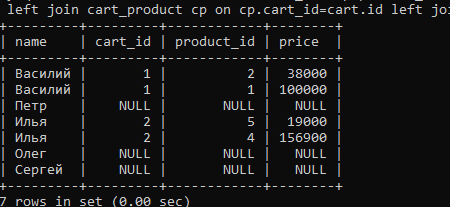
**Еще номер продукта в корзине:**

select c.name, cart.id as cart\_id, cp.product\_id from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id;



**Как насчет цены каждого из товаров, добавим и эту колонку:**

select c.name, cart.id as cart\_id, cp.product\_id, p.price from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id left join product p on p.id = cp.product\_id;



SELECT name

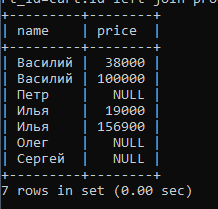
FROM cart\_product

WHERE price < (SELECT AVG(price)

FROM cart\_product)

**Попробуем отобразить таблицу с именами покупателей и ценой товаров, которые они добавили в корзину:**

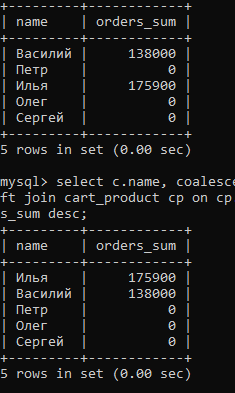
select c.name, p.price from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id left join product p on p.id = cp.product\_id;



**При помощи coalesce вместо null выведем 0 и отсортируем по убыванию цены:**

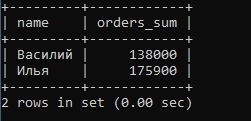
select c.name, coalesce(sum(p.price), 0) as orders\_sum from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id left join product p on p.id = cp.product\_id group by c.name;

select c.name, coalesce(sum(p.price), 0) as orders\_sum from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id left join product p on p.id = cp.product\_id group by c.name order by orders\_sum desc;



**Отобразим только тех покупателей, сумма корзины которых отлична от 0:**

select c.name, coalesce(sum(p.price), 0) as orders\_sum from customer c left join cart on cart.customer\_id=c.id left join cart\_product cp on cp.cart\_id=cart.id left join product p on p.id = cp.product\_id group by c.name having sum(p.price)>0;



**Вывод:** В ходе лабораторной работы освоил основные команды SQL по выборке данных из таблиц, в том числе объединения (join) и вложенные запросы и применил их на практике.