**Министерство науки и высшего образования** **Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет МГТУ им. Н.Э. Баумана)**

**Факультет «Информатика и системы управления» Кафедра «Системы обработки информации и управления»**

Лабораторная работа №2

Курс «Базы данных»

«Основы SQL»

Выполнил:

студент группы ИУ5-45Б

Шакиров Т.М.

Проверил:

Маслеников К.Ю.

2024 г.

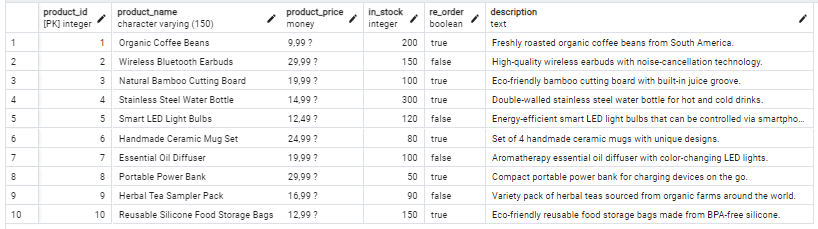
**Цель лабораторной работы:**

Сформировать знания и умения по программированию на языке SQL, приобрести практические навыки работы со средствами языка SQL для выборки и редактирования данных в БД.

**Команды по созданию таблиц:**

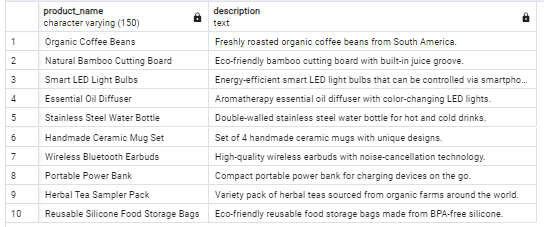
Запрос, выбирающий все данные из таблицы:

SELECT \* FROM products



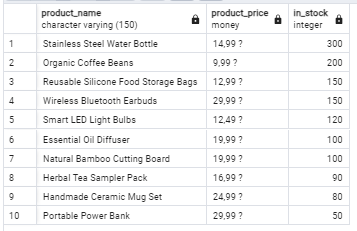
Запрос, выбирающий данные из некоторых столбцов таблицы:

SELECT product\_name, description FROM products as prod, items as itm WHERE prod.product\_id = itm.product\_id



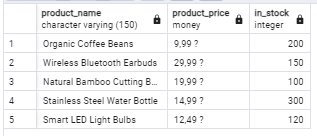
Запрос с использованием сортировки данных:

SELECT product\_name, product\_price, in\_stock FROM products ORDER BY in\_stock DESC



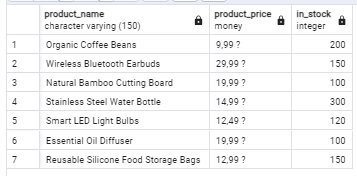
Запрос с использованием ограничения на выборку данных:

SELECT product\_name, product\_price, in\_stock FROM products LIMIT 5



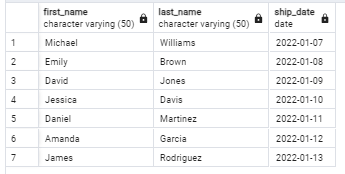
Запрос с использованием операторов сравнения:

SELECT product\_name, product\_price, in\_stock FROM products AS prod WHERE prod.in\_stock >= 100

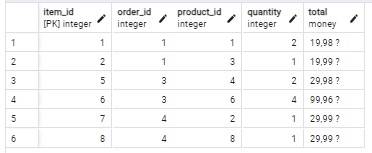


Запрос с использованием оператора BETWEEN:

SELECT first\_name, last\_name, ship\_date FROM orders AS ord, customers AS cust WHERE ord.customer\_id = cust.customer\_id AND ship\_date BETWEEN '2022-01-07' AND '2022-01-13'



Запрос с использованием оператора IN, содержащий подзапрос:

SELECT \* FROM items WHERE order\_id IN (1, 3, 4)

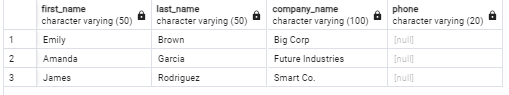
Запрос с использованием оператора LIKE и строковых функций:

SELECT UPPER(company\_name) AS company\_name, first\_name, last\_name FROM customers WHERE first\_name LIKE 'V%'



Запрос с использованием предиката IS NULL:

SELECT first\_name, last\_name, company\_name, phone FROM customers WHERE phone IS null



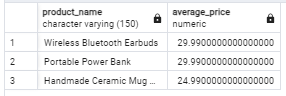
Запрос с использованием агрегатных функций:

SELECT SUM(product\_price) AS sum\_price FROM products



Запрос с использованием агрегатных функций и предложения HAVING:

SELECT product\_name, AVG(product\_price::numeric) AS average\_price FROM products GROUP BY product\_name HAVING AVG(product\_price::numeric) > 20



Запрос, выбирающий данные из нескольких таблиц с использованием

соединения по предикату:

SELECT orders.order\_id, customers.first\_name, customers.last\_name, products.product\_name

FROM orders

JOIN customers ON orders.customer\_id = customers.customer\_id

JOIN items ON orders.order\_id = items.order\_id

JOIN products ON items.product\_id = products.product\_id



Запрос с использованием ключевого слова DISTINCT:

SELECT DISTINCT last\_name FROM customers



Запрос с использованием оператора EXISTS:

SELECT EXISTS (

SELECT last\_name

FROM customers

WHERE last\_name = 'Diesel'

) as last\_name\_Diesel



Запрос с использованием функции CASE:

SELECT

o.order\_id,

o.order\_date,

i.product\_id,

i.quantity,

CASE

WHEN i.quantity > 10 THEN 'Large'

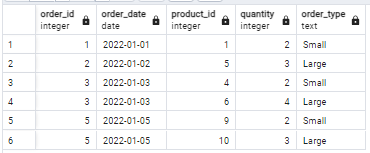
ELSE 'Small'

END AS order\_type

FROM orders o

JOIN items i ON o.order\_id = i.order\_id

WHERE i.quantity > 10;



**Задание по варианту:**

1. Получить информацию о покупателях (компания, фамилия, имя, адрес, телефон, город), которые совершили заказ со статусом «P». Список отсортировать по городу и фамилиям.

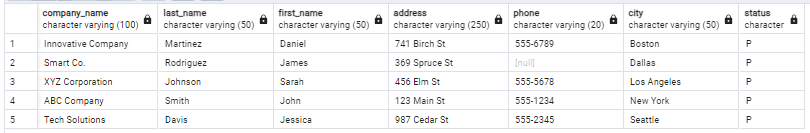
SELECT c.company\_name, c.last\_name, c.first\_name, c.address, c.phone, c.city, o.status

FROM customers c

JOIN orders o ON c.customer\_id = o.customer\_id

WHERE o.status = 'P'

ORDER BY c.city, c.last\_name;



2. Получить информацию о покупателях (фамилия, имя, адрес, телефон, город), чьи заказы были отменены. Список отсортировать по городу и фамилиям.

SELECT c.city, COUNT(DISTINCT c.customer\_id) AS customer\_count

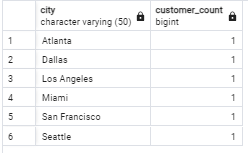
FROM customers c

JOIN orders o ON c.customer\_id = o.customer\_id

WHERE o.paid\_date IS NOT NULL

GROUP BY c.city

ORDER BY customer\_count DESC;



**Вывод:**

После выполнения данной лабораторной работы приобретены практические навыки работы с данными в реляционных базах данных, освоены основные операторы SQL для выборки, фильтрации и сортировки данных. Этот опыт позволяет лучше понимать структуру данных в реляционных базах и составлять запросы для получения нужной информации.