



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»

РТУ МИРЭА

**Институт информационных технологий (ИИТ)
Кафедра цифровой трансформации (ЦТ)**

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2
по дисциплине «Разработка баз данных»

Студент группы *ИКБО-50-23. Павлов Н.С..*

(подпись)

Преподаватель *Мажей Я. В.*

(подпись)

Москва 2025 г.

СОДЕРЖАНИЕ

| | |
|----------------------------|----|
| 1. ПОСТАНОВКА ЗАДАЧИ | 3 |
| 2 ВЫПОЛНЕНИЕ РАБОТЫ | 5 |
| 2.1 ИСХОДНЫЕ ТАБЛИЦЫ | 5 |
| 2.2 ЗАДАНИЕ 1 | 12 |
| 2.3 ЗАДАНИЕ 2 | 15 |

1. ПОСТАНОВКА ЗАДАЧИ

Цель: Научиться извлекать и комбинировать данные из нескольких связанных таблиц с помощью соединений (JOIN) и теоретико-множественных операторов (UNION, INTERSECT, EXCEPT), а также освоить продвинутые паттерны, такие как «само-соединение» и «анти-соединение».

Задачи:

Задание 1: демонстрация различных типов соединений.

На основе индивидуальной схемы данных, составить и выполнить пять аналитических запросов, демонстрирующих различные типы соединений.

Каждый запрос должен решать осмысленную задачу в рамках вашей предметной области.

В начале отчёта должны быть приложены скриншоты всех используемых таблиц индивидуальной схемы данных.

1. Запрос с INNER JOIN: подсчитайте количество связанных записей между таблицами (например, «сколько лекарств у каждого производителя?»)
2. Запрос с LEFT JOIN: проанализируйте наличие или отсутствие связей (например, «сколько лекарств у каждого производителя, включая тех, у кого лекарств нет?»)
3. Запрос с RIGHT JOIN и WHERE... IS NULL (паттерн «анти соединение»): найдите и подсчитайте записи без связей (например, «сколько лекарств не имеют производителя в базе?»)
4. Запрос с FULL JOIN: получите общую статистику – сколько всего связанных записей, и сколько записей без связей.
5. Запрос с CROSS JOIN: сформировать декартово произведение всех записей одной таблицы со всеми записями другой, создав тем самым все возможные комбинации строк между ними.

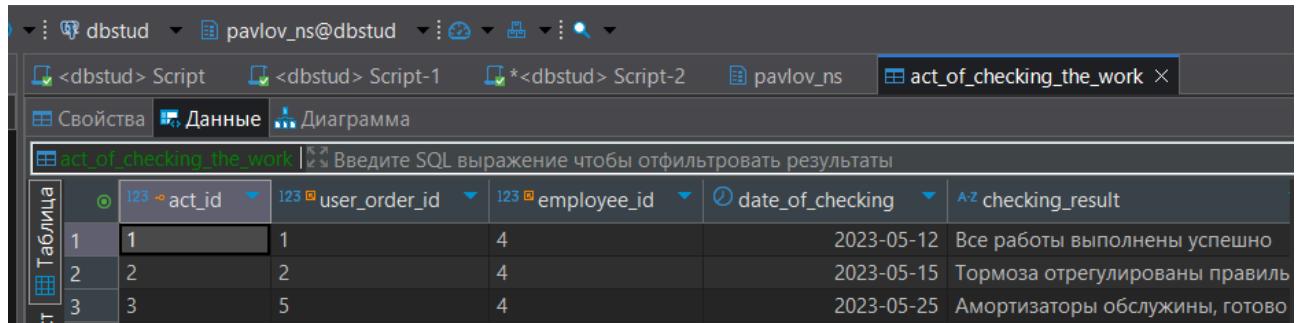
Задание 2: применение теоретико-множественных операторов.

На основе индивидуальной схемы данных составить и выполнить три запроса, демонстрирующих практическое применение операторов UNION, INTERSECT и EXCEPT.

1. UNION: составить единый список из данных двух разных таблиц (столбцы должны быть совместимы по типу).
2. INTERSECT: найти общие записи, которые удовлетворяют двум разным условиям или находятся в двух разных наборах данных.
3. EXCEPT: найти записи, которые присутствуют в одном наборе данных, но отсутствуют в другом.

2 ВЫПОЛНЕНИЕ РАБОТЫ

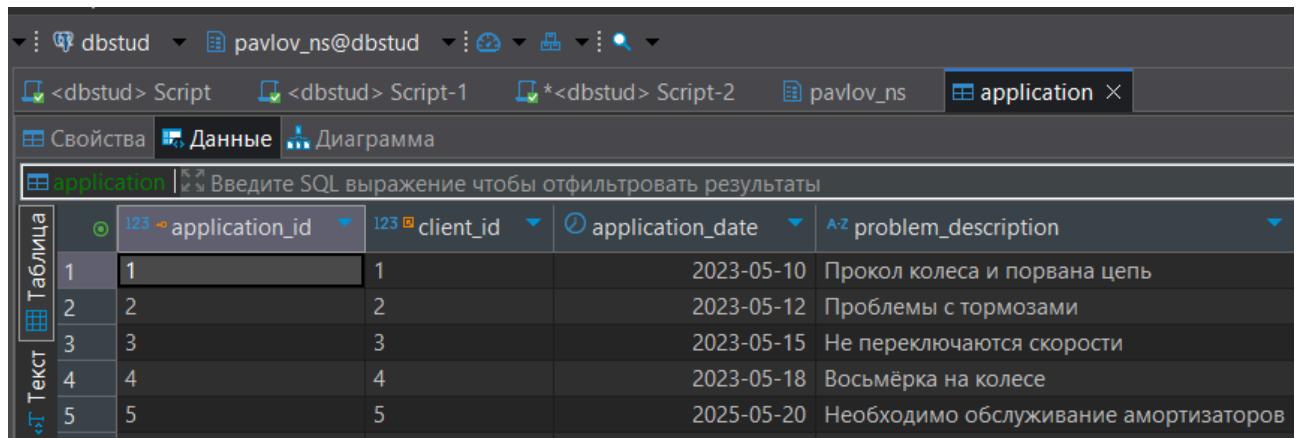
2.1 ИСХОДНЫЕ ТАБЛИЦЫ



The screenshot shows the 'act_of_checking_the_work' table in dbstud. The table has columns: act_id, user_order_id, employee_id, date_of_checking, and checking_result. The data is as follows:

| Таблица | act_id | user_order_id | employee_id | date_of_checking | checking_result |
|---------|--------|---------------|-------------|------------------|--------------------------------|
| | 1 | 1 | 4 | 2023-05-12 | Все работы выполнены успешно |
| | 2 | 2 | 4 | 2023-05-15 | Тормоза отрегулированы правила |
| | 3 | 5 | 4 | 2023-05-25 | Амортизаторы обслужины, готово |

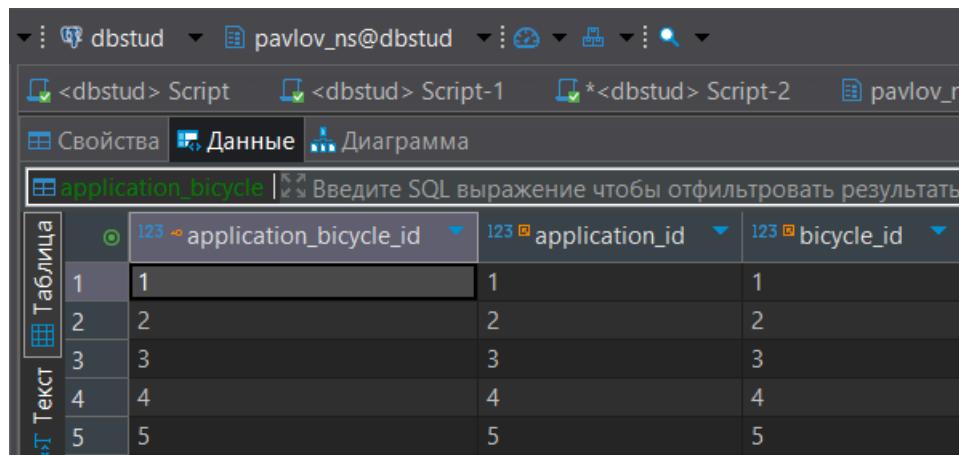
Рисунок 1 – Таблица act_of_checking_the_work



The screenshot shows the 'application' table in dbstud. The table has columns: application_id, client_id, application_date, and problem_description. The data is as follows:

| Таблица | application_id | client_id | application_date | problem_description |
|---------|----------------|-----------|------------------|---------------------------------------|
| | 1 | 1 | 2023-05-10 | Прокол колеса и порвана цепь |
| | 2 | 2 | 2023-05-12 | Проблемы с тормозами |
| | 3 | 3 | 2023-05-15 | Не переключаются скорости |
| | 4 | 4 | 2023-05-18 | Восьмёрка на колесе |
| | 5 | 5 | 2025-05-20 | Необходимо обслуживание амортизаторов |

Рисунок 2 – Таблица application



The screenshot shows the 'application_bicycle' table in dbstud. The table has columns: application_bicycle_id, application_id, and bicycle_id. The data is as follows:

| Таблица | application_bicycle_id | application_id | bicycle_id |
|---------|------------------------|----------------|------------|
| | 1 | 1 | 1 |
| | 2 | 2 | 2 |
| | 3 | 3 | 3 |
| | 4 | 4 | 4 |
| | 5 | 5 | 5 |

Рисунок 3 – Таблица application_bicycle

Свойства Данные Диаграмма

bicycle Введите SQL выражение чтобы отфильтровать результаты

| Таблица | bicycle_id | brand | model | year_of_release | color |
|---------|------------|-------------|-----------------------|-----------------|----------------|
| 1 | 1 | Trek | Marlin 5 | 2 022 | Чёрный |
| 2 | 2 | Giant | Talon 4 | 2 021 | Синий |
| 3 | 3 | Specialized | Rockhopper | 2 023 | Красный |
| 4 | 4 | Cannondale | Trail 8 | 2 020 | Зелёный |
| 5 | 5 | Scott | Scale 930 | 2 024 | Темно-синий |
| 6 | 34 | Scott | Spark RC SL | 2 024 | Красный карбон |
| 7 | 35 | Scott | Addict RC Ultimate | 2 024 | Чёрный карбон |
| 8 | 36 | Scott | Gambler | 2 024 | Неоново-жёлтый |
| 9 | 37 | Specialized | S-Works Tarmac SL8 | 2 024 | Мираж/Сатфин |
| 10 | 38 | Specialized | S-Works Epic EVO | 2 024 | Зелёный карбон |
| 11 | 39 | Specialized | S-Works Stumpjumper | 2 024 | Камуфляж |
| 12 | 40 | Specialized | S-Works Turbo Levo SL | 2 024 | Серый карбон |
| 13 | 41 | Specialized | S-Works Aethos | 2 024 | Матовая медь |
| 14 | 42 | Trek | Session 9.9 XX AXS | 2 024 | Радужный флейм |

Рисунок 4 – Таблица bicycle

Свойства Данные Диаграмма

client Введите SQL выражение чтобы отфильтровать результаты

| Таблица | client_id | surname | name | fathers_name | phone | email |
|---------|-----------|----------|---------|--------------|--------------|------------------------|
| 1 | 1 | Иванов | Иван | Иванович | +79161234567 | ivanov@mail.ru |
| 2 | 2 | Петров | Пётр | Петрович | +79162345678 | petrov@mail.ru |
| 3 | 3 | Сидоров | Алексей | [NULL] | +79163456789 | sidorov@mail.ru |
| 4 | 4 | Смирнова | Мария | Дмитриевна | +79164567890 | smirnova@mail.ru |
| 5 | 5 | Павлов | Никита | Сергеевич | +79611557092 | nikniknik388@gmail.com |
| 6 | 36 | Петров | Иван | Сергеевич | +79103750275 | petrov@gmail.ru |
| 7 | 37 | Смирнова | Варвара | [NULL] | +79166427502 | smirnova@gmail.ru |

Рисунок 5 – Таблица client

Свойства Данные Диаграмма

employee Введите SQL выражение чтобы отфильтровать результаты

| Таблица | employee_id | position_id | surname | name | fathers_name | phone | registration_address | employment_date | contract_due_date |
|---------|-------------|-------------|----------|---------|--------------|--------------|--------------------------|-----------------|-------------------|
| 1 | 1 | 1 | Волков | Андрей | Николаевич | +79166789012 | ул. Главная, 123, Москва | 2020-01-15 | 2025-01-15 |
| 2 | 2 | 2 | Лебедева | Ольга | Сергеевна | +79167890123 | ул. Дубовая, 45, Москва | 2021-03-20 | 2024-03-20 |
| 3 | 3 | 3 | Соколов | Дмитрий | [NULL] | +79168901234 | ул. Сосновая, 78, Москва | 2022-05-10 | 2023-05-10 |
| 4 | 4 | 4 | Козлова | Елена | Владимировна | +79169012345 | ул. Вязовая, 32, Москва | 2019-11-05 | 2024-11-05 |
| 5 | 5 | 5 | Морозов | Игорь | Анатольевич | +79160123456 | ул. Кленовая, 65, Москва | 2023-02-15 | 2023-08-15 |

Рисунок 6 – Таблица employee

Screenshot of MySQL Workbench showing the 'estimate' table. The table has columns: estimate_id, user_order_id, price_of_work, price_of_spare_parts, and total_price. Data rows show various estimates for user orders.

| | estimate_id | user_order_id | price_of_work | price_of_spare_parts | total_price |
|---|-------------|---------------|---------------|----------------------|-------------|
| 1 | 1 | 1 | | 1 300 | 1 700 |
| 2 | 2 | 2 | | 1 000 | 800 |
| 3 | 3 | 3 | | 1 200 | 2 500 |
| 4 | 4 | 4 | | 1 500 | 50 |
| 5 | 5 | 5 | | 1 500 | 1 300 |
| | | | | | 3 000 |
| | | | | | 1 800 |
| | | | | | 3 700 |
| | | | | | 1 550 |
| | | | | | 2 800 |

Рисунок 7 – Таблица estimate

Screenshot of MySQL Workbench showing the 'malfunction' table. The table has columns: malfunction_id, title, criticality, and description. Data rows list various vehicle malfunctions.

| | malfunction_id | title | criticality | description |
|---|----------------|--------------------------|-------------|--------------------------------------|
| 1 | 1 | Прокол колеса | Низкая | Прокол камеры |
| 2 | 2 | Обрыв цепи | Средняя | Цепь порвана или повреждена |
| 3 | 3 | Неисправность тормозов | Высокая | Тормоза работают неудовлетворительно |
| 4 | 4 | Проблемы с переключением | Средняя | Скорости переключаются неправильно |
| 5 | 5 | Восьмёрка на колесе | Средняя | Колесо требует правки |

Рисунок 8 – Таблица malfunction

Screenshot of MySQL Workbench showing the 'position' table. The table has columns: position_id, title, and salary. Data rows list job positions and their salaries.

| | position_id | title | salary |
|---|-------------|------------------|--------|
| 1 | 1 | Главный механик | 80 000 |
| 2 | 2 | Старший механик | 60 000 |
| 3 | 3 | Младший механик | 40 000 |
| 4 | 4 | Менеджер сервиса | 70 000 |
| 5 | 5 | Администратор | 35 000 |

Рисунок 9 – Таблица position

| | 123 ↗ spare_part_id | A-Z title | A-Z specifications | 123 ↗ price | 123 ↗ supplier_id |
|---|---------------------|-------------------|------------------------------------|-------------|-------------------|
| 1 | 1 | Камера | 29 дюймов, автомобильный ниппель | 500 | 1 |
| 2 | 2 | Цепь | 9 скоростей, 116 звеньев | 1 200 | 2 |
| 3 | 3 | Колодки тормозные | Дисковые, резиновые | 800 | 3 |
| 4 | 4 | Переключатель | Задний, 9 скоростей | 2 500 | 4 |
| 5 | 5 | Спицы | Нержавеющая сталь, 260 мм | 50 | 5 |
| 6 | 34 | Камера | 29 дюймов, велосипедный ниппель | 1 500 | 1 |
| 7 | 35 | Цепь | 9 скоростей, 116 звеньев | 2 200 | 2 |
| 8 | 36 | Камера | 27.5 дюймов, автомобильный ниппель | 1 800 | 3 |

Рисунок 10 – Таблица spare_part

| | 123 ↗ status_id | A-Z title | A-Z description |
|---|-----------------|--------------------|---|
| 1 | 1 | Принята | Заявка принята, но ещё не обработана |
| 2 | 2 | В работе | Ремонт выполняется |
| 3 | 3 | Ожидание запчастей | Ожидание поставки необходимых запчастей |
| 4 | 4 | Готово к выдаче | Ремонт завершён, можно забирать |
| 5 | 5 | Отменена | Заказ отменён |

Рисунок 11 – Таблица status

| | 123 ↗ supplier_id | A-Z organization_title | A-Z address | A-Z phone |
|---|-------------------|------------------------|-----------------------------------|--------------|
| 1 | 1 | Велоомплект | ул. Промышленная, 123, Москва | +74951234567 |
| 2 | 2 | Велотехника | ул. Торговая, 45, Санкт-Петербург | +78122345678 |
| 3 | 3 | БайкТулс | пр. Детальный, 78, Казань | +78433456789 |
| 4 | 4 | СпортЗапчасть | б-р Компонентов, 32, Москва | +73832567890 |
| 5 | 5 | Гиромастер | ул. Звёздочная, 65, Москва | +73434567890 |

Рисунок 12 – Таблица supplier

The screenshot shows the MySQL Workbench interface with the database 'dbstud' and user 'pavlov_ns@dbstud'. The 'transaction' table is selected. The table has three columns: 'transaction_id' (primary key), 'user_order_id', and 'payment_date'. The data is as follows:

| | transaction_id | user_order_id | payment_date |
|---|----------------|---------------|--------------|
| 1 | 1 | 1 | 2023-05-12 |
| 2 | 2 | 2 | 2023-05-15 |
| 3 | 3 | 5 | 2024-05-25 |

Рисунок 13 – Таблица transaction

The screenshot shows the MySQL Workbench interface with the database 'dbstud' and user 'pavlov_ns@dbstud'. The 'user_order' table is selected. The table has six columns: 'user_order_id' (primary key), 'application_id', 'status_id', 'start_date', and 'end_date'. The data is as follows:

| | user_order_id | application_id | status_id | start_date | end_date |
|---|---------------|----------------|-----------|------------|------------|
| 1 | 1 | 1 | 2 | 2023-05-10 | 2023-05-12 |
| 2 | 2 | 2 | 2 | 2023-05-12 | 2023-05-15 |
| 3 | 3 | 3 | 3 | 2023-05-15 | 2023-05-20 |
| 4 | 4 | 4 | 1 | 2023-05-18 | 2023-05-22 |
| 5 | 5 | 5 | 4 | 2024-05-20 | 2024-05-25 |

Рисунок 14 – Таблица user_order

The screenshot shows the MySQL Workbench interface with the database 'dbstud' and user 'pavlov_ns@dbstud'. The 'user_order_employee' table is selected. The table has three columns: 'user_order_employee_id' (primary key), 'user_order_id', and 'employee_id'. The data is as follows:

| | user_order_employee_id | user_order_id | employee_id |
|---|------------------------|---------------|-------------|
| 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 2 |
| 5 | 5 | 5 | 1 |

Рисунок 15 – Таблица user_order_employee

The screenshot shows the dbstud database interface with the 'Свойства' (Properties) tab selected. Below it is a table named 'user_order_malfunction' with three columns: 'user_order_malfunction_id', 'user_order_id', and 'malfunction_id'. The data consists of six rows:

| | user_order_malfunction_id | user_order_id | malfunction_id |
|---|---------------------------|---------------|----------------|
| 1 | 1 | 1 | 1 |
| 2 | 2 | 1 | 2 |
| 3 | 3 | 2 | 3 |
| 4 | 4 | 3 | 4 |
| 5 | 5 | 4 | 5 |
| 6 | 6 | 5 | 3 |

Рисунок 16 – Таблица user_order_malfunction

The screenshot shows the dbstud database interface with the 'Свойства' (Properties) tab selected. Below it is a table named 'user_order_work' with three columns: 'user_order_work_id', 'user_order_id', and 'work_id'. The data consists of seven rows:

| | user_order_work_id | user_order_id | work_id |
|---|--------------------|---------------|---------|
| 1 | 1 | 1 | 1 |
| 2 | 2 | 1 | 2 |
| 3 | 3 | 2 | 3 |
| 4 | 4 | 3 | 4 |
| 5 | 5 | 4 | 5 |
| 6 | 6 | 5 | 1 |
| 7 | 7 | 5 | 3 |

Рисунок 17 – Таблица user_order_work

The screenshot shows the dbstud database interface with the 'Свойства' (Properties) tab selected. Below it is a table named 'warehouse' with two columns: 'warehouse_id' and 'spare_part_id'. The data consists of five rows:

| | warehouse_id | spare_part_id |
|---|--------------|---------------|
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 3 |
| 4 | 4 | 4 |
| 5 | 5 | 5 |

Рисунок 18 – Таблица warehouse

Свойства Данные Диаграмма

work | Введите SQL выражение чтобы отфильтровать результаты

| | 123 ↗ work_id | A-Z title | A-Z description_of_the_execution | 123 lead_time | 123 price |
|---|---------------|------------------------|---|---------------|-----------|
| 1 | 1 | Замена камеры | Снять старую камеру, установить новую | 30 | 500 |
| 2 | 2 | Замена цепи | Установить новую цепь, подогнать по длине | 45 | 800 |
| 3 | 3 | Регулировка тормозов | Настройка колодок и тросиков | 60 | 1 000 |
| 4 | 4 | Настройка переключения | Регулировка переключателя и манеток | 60 | 1 200 |
| 5 | 5 | Правка колеса | Регулировка спиц для устранения восьмёрки | 90 | 1 500 |

Рисунок 19 – Таблица work

Свойства Данные Диаграмма

work_spare_part | Введите SQL выражение чтобы отфильтровать результаты

| | 123 ↗ work_spare_part_id | 123 ↗ work_id | 123 ↗ spare_part_id |
|---|--------------------------|---------------|---------------------|
| 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 4 |
| 5 | 5 | 5 | 5 |

Рисунок 20 – Таблица work_spare_part

2.2 ЗАДАНИЕ 1

The screenshot shows a database interface with a query editor and a results grid.

Query:

```
SELECT
    sup.organization_title AS organization_title,
    COUNT(sp.spare_part_id) AS sp_count
FROM
    spare_part AS sp
INNER JOIN
    supplier AS sup
ON
    sp.supplier_id = sup.supplier_id
GROUP BY
    sup.supplier_id
```

Results:

| Таблица | organization_title | sp_count |
|---------|--------------------|----------|
| 1 | СпортЗапчасть | 1 |
| 2 | Велотехника | 2 |
| 3 | БайкТулс | 2 |
| 4 | Велокомплект | 2 |
| 5 | Гиромастер | 1 |

Рисунок 21 – Запрос с INNER JOIN

The screenshot shows a database interface with a query editor and a results grid.

Query:

```
SELECT
    c.surname,
    c.name,
    COUNT(a.application_id) AS count_of_application
FROM
    client AS c
LEFT JOIN
    application AS a
ON
    c.client_id = a.client_id
GROUP BY
    c.client_id
ORDER BY
    count_of_application DESC
```

Results:

| Таблица | surname | name | count_of_application |
|---------|----------|---------|----------------------|
| 1 | Смирнова | Мария | 1 |
| 2 | Петров | Петр | 1 |
| 3 | Иванов | Иван | 1 |
| 4 | Павлов | Никита | 1 |
| 5 | Сидоров | Алексей | 1 |
| 6 | Смирнова | Варвара | 0 |
| 7 | Петров | Иван | 0 |

Рисунок 22 – Запрос с LEFT JOIN

The screenshot shows a database interface with a query editor and a results viewer. The query is:

```

SELECT
    c.surname,
    c.name,
    a.problem_description
FROM
    application AS a
RIGHT JOIN
    client AS c
ON
    a.client_id = c.client_id
WHERE
    a.application_id IS NULL

```

The results table shows two rows:

| | surname | name | problem_description |
|---|----------|---------|---------------------|
| 1 | Смирнова | Варвара | [NULL] |
| 2 | Петров | Иван | [NULL] |

Рисунок 23 – Запрос с RIGHT JOIN

The screenshot shows a database interface with a query editor and a results viewer. The query is:

```

SELECT
    COALESCE(sup.organization_title, 'Не указан') AS manufacturer_name,
    COUNT(sp.spare_part_id) AS count_of_parts
FROM
    supplier AS sup
FULL JOIN
    spare_part AS sp
ON
    sup.supplier_id = sp.supplier_id
GROUP BY
    sup.supplier_id

```

The results table shows five rows:

| | manufacturer_name | count_of_parts |
|---|-------------------|----------------|
| 1 | СпортЗапчасть | 1 |
| 2 | Велотехника | 2 |
| 3 | БайкТулс | 2 |
| 4 | Велокомплект | 2 |
| 5 | Гиромастер | 1 |

Рисунок 24 – Запрос с FULL JOIN

The screenshot shows a database interface with a query editor and a results viewer.

Query Editor:

```

SELECT
    sp.title,
    sp.specifications,
    sup.organization_title
FROM
    spare_part AS sp
CROSS JOIN
    supplier AS sup

```

Results View:

| № | A-Z title | A-Z specifications | A-Z organization_title |
|----|-------------------|----------------------------------|------------------------|
| 1 | Камера | 29 дюймов, автомобильный ниппель | Велокомплект |
| 2 | Камера | 29 дюймов, автомобильный ниппель | Велотехника |
| 3 | Камера | 29 дюймов, автомобильный ниппель | БайкТулс |
| 4 | Камера | 29 дюймов, автомобильный ниппель | СпортЗапчасть |
| 5 | Камера | 29 дюймов, автомобильный ниппель | Гиромастер |
| 6 | Цепь | 9 скоростей, 116 звеньев | Велокомплект |
| 7 | Цепь | 9 скоростей, 116 звеньев | Велотехника |
| 8 | Цепь | 9 скоростей, 116 звеньев | БайкТулс |
| 9 | Цепь | 9 скоростей, 116 звеньев | СпортЗапчасть |
| 10 | Цепь | 9 скоростей, 116 звеньев | Гиромастер |
| 11 | Колодки тормозные | Дисковые, резиновые | Велокомплект |
| 12 | Колодки тормозные | Дисковые, резиновые | Велотехника |
| 13 | Колодки тормозные | Дисковые, резиновые | БайкТулс |
| 14 | Колодки тормозные | Дисковые, резиновые | СпортЗапчасть |
| 15 | Колодки тормозные | Дисковые, резиновые | Гиромастер |
| 16 | Переключатель | Задний, 9 скоростей | Велокомплект |
| 17 | Переключатель | Задний, 9 скоростей | Велотехника |
| 18 | Переключатель | Задний, 9 скоростей | БайкТулс |
| 19 | Переключатель | Задний, 9 скоростей | СпортЗапчасть |
| 20 | Переключатель | Задний, 9 скоростей | Гиромастер |
| 21 | Спицы | Нержавеющая сталь, 260 мм | Велокомплект |
| 22 | Спицы | Нержавеющая сталь, 260 мм | Велотехника |
| 23 | Спицы | Нержавеющая сталь, 260 мм | БайкТулс |
| 24 | Спицы | Нержавеющая сталь, 260 мм | СпортЗапчасть |
| 25 | Спицы | Нержавеющая сталь, 260 мм | Гиромастер |
| 26 | Камера | 29 дюймов, велосипедный ниппель | Велокомплект |

Рисунок 25 – Запрос с CROSS JOIN

2.3 ЗАДАНИЕ 2

The screenshot shows the MySQL Workbench interface. In the top-left corner, the connection details are displayed: dbstud and pavlov_ns@dbstud. Below this, two tabs are open: <dbstud> Script and <dbstud> Script. The left pane contains a tree view of database objects. The main pane displays the following SQL query:

```
SELECT phone
FROM employee
UNION
SELECT phone
FROM client
```

Below the query, there is a toolbar with various icons. The bottom section is titled "Результат 1" (Result 1) and shows the execution of the query. The results are presented in a table:

| Таблица | phone |
|---------|--------------|
| 1 | +79164567890 |
| 2 | +79160123456 |
| 3 | +79167890123 |
| 4 | +79162345678 |
| 5 | +79169012345 |
| 6 | +79166789012 |
| 7 | +79163456789 |
| 8 | +79168901234 |
| 9 | +79161234567 |
| 10 | +79611557092 |
| 11 | +79166427502 |
| 12 | +79103750275 |

Рисунок 26 – Запрос с UNION

The screenshot shows a database interface with a query editor and a results viewer. The query is:

```
SELECT
    surname,
    name
FROM
    employee
INTERSECT
    SELECT
        e.surname,
        e.name
    FROM
        employee AS e
    INNER JOIN
        position AS p
    ON
        e.position_id = p.position_id
    WHERE
        p.title = 'Старший механик'
```

The results table shows one row:

| | surname | name |
|---|----------|-------|
| 1 | Лебедева | Ольга |

Рисунок 27 – Запрос с INTERSECT

The screenshot shows a database interface with a query editor and a results viewer. The query is:

```
SELECT
    surname,
    name
FROM
    employee
EXCEPT
    SELECT
        e.surname,
        e.name
    FROM
        employee AS e
    INNER JOIN
        position AS p
    ON
        e.position_id = p.position_id
    WHERE
        p.title = 'Старший механик'
```

The results table shows four rows:

| | surname | name |
|---|---------|---------|
| 1 | Морозов | Игорь |
| 2 | Волков | Андрей |
| 3 | Соколов | Дмитрий |
| 4 | Козлова | Елена |

Рисунок 28 – Запрос с EXCEPT