

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ  
Факультет компьютерных систем и сетей  
Кафедра электронных вычислительных машин  
Дисциплина: Базы данных

Тема «Грузоперевозки»  
Лабораторная работа №5  
Реализация SQL-запросов на выборку данных с группировкой результатов

Студент:  
Преподаватель:

Е.О. Лукьянов  
Д.В. Куприянова

МИНСК 2025

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 SQL-ЗАПРОСЫ.....	4
1.1 Транспортные средства.....	4
1.2 Транспортные операции.....	6
1.3 Грузы.....	8
1.4 Клиенты.....	10
1.5 Водители.....	12
1.6 Оплаты.....	13
1.7 Маршруты.....	14
ЗАКЛЮЧЕНИЕ.....	19

## ВВЕДЕНИЕ

В данной лабораторной работе будет получен опыт по созданию SQL-запросы для выборки данных из реляционных баз данных, используя подзапросы, агрегатные функции, а также группировку данных с помощью оператора GROUP BY и операции над множествами (UNION, INTERSECT, MINUS). Работа включает несколько этапов, начиная с получения заданий от преподавателя на основе схемы данных, созданной в лабораторной работе №2, и реализованной в виде таблиц в СУБД в лабораторной работе №3.

Основная цель данной лабораторной работы — формирование практических навыков написания SQL-запросов, которые эффективно взаимодействуют с данными. При выполнении заданий важно учитывать правила написания запросов, включая ограничения на использование скалярных подзапросов и необходимость предварительного изучения данных в таблицах. Каждый запрос будет реализован в виде одного оператора SQL SELECT, который может включать подзапросы и группировку данных.

Эта лабораторная работа поможет углубить понимание работы с SQL и подготовит нас к более сложным задачам анализа и обработки данных в реляционных базах данных.

# 1 SQL-ЗАПРОСЫ

## 1.1 Транспортные средства

Задание: *Вывести модель и суммарную грузоподъёмность всех транспортных средств по каждой модели по возрастанию.*

Скрипт для вывода модели и суммарной грузоподъёмности всех транспортных средств по каждой модели по возрастанию.

```
SELECT      "Модель",          SUM ("Грузоподъёмность")      AS
Суммарная_грузоподъемность
FROM "Транспортные средства"
GROUP BY "Модель"
ORDER BY Суммарная_грузоподъемность ASC;
```

Таблица «Транспортные средства» до скрипта представлена на рисунке 1.1.

	id транспортного средства [PK] integer	Производитель character varying (20)	Модель character varying (20)	Грузоподъёмность double precision	Регистрационный номер character varying (10)	id транспортной операции integer
1	1	Mercedes	Sprinter	3500	A123BC	1
2	2	Volvo	FH16	8000	B234CD	2
3	3	MAN	TGS	6500	C345DE	3
4	4	Scania	R450	7500	D456EF	4
5	5	Renault	Master	3500	E567FG	5
6	6	Iveco	EuroCargo	5000	F678GH	6
7	7	Mercedes	Atego	4000	G789HI	7
8	8	Volvo	FMX	7000	H890IJ	8
9	9	MAN	TGX	8500	I901JK	9
10	10	Scania	S500	7600	J012KL	10
11	11	Renault	T460	6800	K123LM	11
12	12	Iveco	Stralis	7200	L234MN	12
13	13	Mercedes	Actros	9500	M345NO	13
14	14	Volvo	VNL	8000	N456OP	14
15	15	MAN	Lions Coach	9000	O567PQ	15
16	16	Scania	G410	7000	P678QR	16
17	17	Renault	T520	6500	Q789RS	17
18	18	Iveco	Daily	3000	R890ST	18
19	19	Mercedes	Vito	2800	S901TU	19
20	20	Volvo	V90	5500	T012UV	20
21	21	MAN	F2000	6400	U123VW	21
22	22	Scania	L Series	5300	V234WX	22
23	23	Renault	K Range	8500	W345XY	23
24	24	Iveco	EuroTech	7800	X456YZ	24
25	25	Mercedes	Benz	4000	Y567ZA	25
26	26	Volvo	Globetrotter	9500	Z678AB	26
27	27	MAN	ClubTruck	6600	A789BC	27
28	28	Scania	V8	9200	B890CD	28
29	29	Renault	Range T	6700	C901DE	29
30	30	Iveco	EuroStar	7300	D012EF	30
31	31	Mercedes	Zetros	10000	E123FG	31

Рисунок 1.1 – Таблица «Транспортные средства» до скрипта

Таблица «Транспортные средства» после скрипта представлена на рисунке 1.2.

	Модель character varying (20)	Суммарная_грузоподъемность double precision
1	Vito	2800
2	Daily	3000
3	Master	3500
4	Sprinter	3500
5	Atego	4000
6	Benz	4000
7	EuroCargo	5000
8	L Series	5300
9	V90	5500
10	F2000	6400
11	T520	6500
12	TGS	6500
13	ClubTruck	6600
14	Range T	6700
15	T460	6800
16	G410	7000
17	FMX	7000
18	Stralis	7200
19	EuroStar	7300
20	R450	7500
21	S500	7600
22	EuroTech	7800
23	FH16	8000
24	VNL	8000
25	TGX	8500
26	K Range	8500
27	Lions Coach	9000
28	V8	9200
29	Globetrotter	9500
30	Actros	9500
31	Zetros	10000

Рисунок 1.2 – Результат скрипта

Задание: *Вывести производителя и максимальную грузоподъемность среди его моделей по убыванию.*

Скрипт для вывода производителей и максимальной грузоподъемности среди их моделей по убыванию.

```
SELECT      "Производитель",      MAX ("Грузоподъемность")      AS  
Максимальная_грузоподъемность  
FROM "Транспортные средства"  
GROUP BY "Производитель"  
ORDER BY Максимальная_грузоподъемность DESC;
```

Таблица «Транспортные средства» до скрипта представлена на рисунке 1.1.

Таблица «Транспортные средства» после скрипта представлена на рисунке 1.3.

	Производитель character varying (20)	Максимальная_грузоподъемность double precision
1	Mercedes	10000
2	Volvo	9500
3	Scania	9200
4	MAN	9000
5	Renault	8500
6	Iveco	7800

Рисунок 1.3 – Результат скрипта

## 1.2 Транспортные операции

Задание: *Вывести суммарную стоимость всех оплат по каждому статусу транспортной операции.*

Скрипт для вывода суммарной стоимости всех оплат по каждому статусу транспортной операции.

```
SELECT то."Статус", SUM(о."Сумма") AS Суммарная_стоимость  
FROM "Транспортные операции" то  
JOIN "Оплаты" о ON то."id транспортной операции" = о."id  
транспортной операции"  
GROUP BY то."Статус";
```

Таблица «Транспортные операции» до скрипта представлена на рисунке 1.4.

	id транспортной операции [PK] integer	Статус character varying (15)	Ожидаемое время доставки time without time zone	Дата прибытия date	id маршрута integer
1	1	В процессе	12:30:00	2025-02-05	3
2	2	Завершена	14:30:00	2025-02-06	4
3	3	В процессе	15:00:00	2025-02-07	5
4	4	Завершена	17:30:00	2025-02-08	6
5	5	В процессе	11:00:00	2025-02-09	7
6	6	Завершена	16:30:00	2025-02-10	8
7	7	В процессе	12:45:00	2025-02-11	9
8	8	Завершена	13:30:00	2025-02-12	10
9	9	В процессе	14:00:00	2025-02-13	1
10	10	Завершена	15:30:00	2025-02-14	2
11	11	В процессе	16:15:00	2025-02-15	3
12	12	Завершена	17:00:00	2025-02-16	4
13	13	В процессе	11:30:00	2025-02-17	5
14	14	Завершена	14:45:00	2025-02-18	6
15	15	В процессе	13:15:00	2025-02-19	7
16	16	Завершена	15:00:00	2025-02-20	8
17	17	В процессе	14:30:00	2025-02-21	9
18	18	Завершена	12:00:00	2025-02-22	10
19	19	В процессе	11:45:00	2025-02-23	1
20	20	Завершена	13:30:00	2025-02-24	2
21	21	В процессе	16:00:00	2025-02-25	3
22	22	Завершена	14:30:00	2025-02-26	4
23	23	В процессе	17:00:00	2025-02-27	5
24	24	Завершена	11:00:00	2025-02-28	6
25	25	В процессе	15:30:00	2025-03-01	7
26	26	Завершена	13:00:00	2025-03-02	8
27	27	В процессе	16:30:00	2025-03-03	9
28	28	Завершена	12:15:00	2025-03-04	10
29	29	В процессе	14:00:00	2025-03-05	1
30	30	Завершена	15:30:00	2025-03-06	2
31	31	В процессе	16:00:00	2025-03-07	3

Рисунок 1.4 – Таблица «Транспортные операции» до скрипта

Таблица «Транспортные операции» после скрипта представлена на рисунке 1.5.

	Статус character varying (15)	Суммарная_стоимость double precision
1	Завершена	3100
2	В процессе	3390

Рисунок 1.5 – Результат скрипта

Задание: *Вывести среднюю стоимость оплаты для каждого маршрута по возрастанию.*

Скрипт для вывода средней стоимости оплаты для каждого маршрута по возрастанию.

```
SELECT м."Точка отправления", м."Точка прибытия",
ROUND (AVG (о."Сумма")) AS Средняя_стоимость
FROM "Маршруты" м
JOIN "Транспортные операции" то ON м."id маршрута" = то."id маршрута"
JOIN "Оплаты" о ON то."id транспортной операции" = о."id транспортной операции"
GROUP BY м."Точка отправления", м."Точка прибытия"
ORDER BY Средняя_стоимость ASC;
```

Таблица «Транспортные операции» до скрипта представлена на рисунке 1.4.

Таблица «Транспортные операции» после скрипта представлена на рисунке 1.6.

	Точка отправления character varying (20)	Точка прибытия character varying (20)	Средняя_стоимость double precision
1	Минск	Гродно	172
2	Минск	Витебск	177
3	Гомель	Брест	187
4	Брест	Гомель	190
5	Минск	Гомель	213
6	Брест	Минск	217
7	Минск	Брест	223
8	Гомель	Минск	230
9	Гомель	Гродно	240
10	Гомель	Витебск	257

Рисунок 1.6 – Результат скрипта

### 1.3 Грузы

Задание: *Вывести средний вес грузов по каждой точке отправления.*

Скрипт для вывода среднего веса грузов по каждой точке отправления.

```
SELECT м."Точка отправления", ROUND (AVG (г."Вес")) AS
Средний_вес
FROM "Грузы" г
JOIN "Транспортные операции" то ON г."id транспортной
операции" = то."id транспортной операции"
JOIN "Маршруты" м ON то."id маршрута" = м."id маршрута"
GROUP BY м."Точка отправления";
```



Таблица «Грузы» до скрипта представлена на рисунке 1.7.

	id груза [PK] integer	Название character varying (100)	Вес double precision	Объём double precision	Дата погрузки date	id транспортной операции integer
1	1	Продукция сельского хозяйства	63.4	[null]	2025-02-06	7
2	2	Строительные материалы	246.12	94.95	2025-02-24	2
3	3	Продукты питания	999.29	[null]	2025-02-06	10
4	4	Офисная мебель	745.34	56.26	2025-02-26	7
5	5	Электронные устройства	473.8	42.29	2025-02-15	7
6	6	Стеклянные изделия	582.5	[null]	2025-02-10	1
7	7	Автозапчасти	320.75	12.6	2025-02-18	3
8	8	Текстильная продукция	654.32	87.1	2025-02-22	9
9	9	Пластиковые трубы	123.98	18.9	2025-02-13	5
10	10	Деревянные панели	245.6	[null]	2025-02-19	4
11	11	Бытовая техника	413.12	73.55	2025-02-16	8
12	12	Металлические конструкции	765.25	90.3	2025-02-21	6
13	13	Продукция агропромышленности	290.78	[null]	2025-02-14	2
14	14	Химикаты	510.89	55.44	2025-02-20	10
15	15	Косметика и парфюмерия	820.6	[null]	2025-02-17	1
16	16	Автозапчасти	325.43	21.2	2025-02-23	7
17	17	Книги и печатная продукция	105.2	[null]	2025-02-25	3
18	18	Детские игрушки	600.34	66.8	2025-02-08	9
19	19	Мебель для дома	413.67	[null]	2025-02-09	4
20	20	Фармацевтические препараты	256.9	80.5	2025-02-12	6
21	21	Упаковочные материалы	391.12	[null]	2025-02-11	8
22	22	Продукция для животных	112.34	23.6	2025-02-05	5
23	23	Стеклянные бутылки	800.78	45.2	2025-02-07	2
24	24	Строительные товары	567.9	[null]	2025-02-10	7
25	25	Моторные жидкости	434.2	58.3	2025-02-18	3
26	26	Пищевая продукция	111.5	[null]	2025-02-13	10
27	27	Запчасти для сельскохозяйственной техники	222.3	40.2	2025-02-20	4
28	28	Инструменты для ремонта	689.7	[null]	2025-02-25	9
29	29	Грузовая техника	543.12	85.1	2025-02-22	1
30	30	Молочные продукты	678.34	[null]	2025-02-19	8
31	31	Шины и колесные диски	234.78	37.5	2025-02-21	5

Рисунок 1.7 – Таблица «Грузы» до скрипта

Таблица «Грузы» после скрипта представлена на рисунке 1.8.

	Точка отправления character varying (20)	Средний_вес double precision
1	Гомель	294
2	Брест	457
3	Минск	571

Рисунок 1.8 – Результат скрипта

*Задание: Вывести количество грузов с объёмом более 50м<sup>3</sup> по каждой точке прибытия, при условии что их количество больше 1.*

*Скрипт для вывода количества грузов с объёмом более 50м<sup>3</sup> по каждой точке прибытия, при условии что их количество больше 1.*

```

SELECT  m."Точка прибытия",    COUNT(g."id груза")    AS
Количество_грузов
FROM "Грузы" g
JOIN  "Транспортные операции" то ON g."id транспортной
операции" = то."id транспортной операции"
JOIN "Маршруты" m ON то."id маршрута" = m."id маршрута"
WHERE g."Объём" > 50
GROUP BY m."Точка прибытия"
HAVING COUNT(g."id груза") > 1;

```

Таблица «Грузы» до скрипта представлена на рисунке 1.7.

Таблица «Грузы» после скрипта представлена на рисунке 1.9.

	Точка прибытия character varying (20) 🔒	Количество_грузов bigint 🔒
1	Гомель	3
2	Витебск	3
3	Минск	2

Рисунок 1.9 – Результат скрипта

## 1.4 Клиенты

*Задание: Вывести количество транспортных операций для клиентов из Минска и Гомеля.*

Скрипт для вывода количества транспортных операций для клиентов из Минска и Гомеля.

```

SELECT
CASE
    WHEN k."Адрес" LIKE '%Минск%' THEN 'Минск'
    WHEN k."Адрес" LIKE '%Гомель%' THEN 'Гомель'
END AS Город,
COUNT(kto."id транспортной операции") AS Количество_операций
FROM "Клиенты" k
JOIN "Клиенты-Транспортные операции" kto ON k."id клиента" =
kto."id клиента"
WHERE k."Адрес" LIKE '%Минск%' OR k."Адрес" LIKE '%Гомель%'
GROUP BY CASE
    WHEN k."Адрес" LIKE '%Минск%' THEN 'Минск'
    WHEN k."Адрес" LIKE '%Гомель%' THEN 'Гомель'
END;

```

Таблица «Клиенты» до скрипта представлена на рисунке 1.10.

	Id клиента [PK] integer	Название организации character varying (100)	Адрес character varying (100)	Номер телефона character varying (20)	Адрес электронной почты character varying (50)
1	1	БелТрансЛогистик	Минск, ул. Логистическая, 10	+375291112201	contact@btl.by
2	2	ГрузЭкспресс	Гомель, ул. Центральная, 5	+375291112202	info@gruzexpress.by
3	3	ТрансАвто	Витебск, ул. Советская, 15	+375291112203	sales@transauto.by
4	4	ЛогистикСервис	Брест, ул. Транспортная, 12	+375291112204	logistic@ls.by
5	5	ГлобалКарго	Гродно, ул. Складская, 8	+375291112205	global@cargo.by
6	6	ТрансЛогист	Минск, ул. Промышленная, 25	+375291112206	info@translogist.by
7	7	ЭкспрессКарго	Гомель, ул. Погрузочная, 30	+375291112207	support@expresscargo.by
8	8	ФрахтСервис	Витебск, ул. Складская, 14	+375291112208	contact@frachtservice.by
9	9	СкороДоставка	Брест, ул. Магистральная, 22	+375291112209	info@skorodostavka.by
10	10	Европейские перевозки	Гродно, ул. Логистическая, 5	+375291112210	service@eurocargo.by
11	11	БелТрансСнаб	Минск, ул. Техническая, 17	+375291112211	contact@btanab.by
12	12	ЛогистикПро	Гомель, ул. Грузовая, 40	+375291112212	info@logisticpro.by
13	13	ТранспортПлюс	Витебск, ул. Дальняя, 11	+375291112213	support@transportplus.by
14	14	КаргоБел	Брест, ул. Перевозочная, 19	+375291112214	service@cargobel.by
15	15	БелТрансСервис	Гродно, ул. Погрузочная, 7	+375291112215	contact@beltransservice.by
16	16	ФорвардТранс	Минск, ул. Логистическая, 33	+375291112216	info@forwardtrans.by
17	17	ТрансГарант	Гомель, ул. Путевая, 8	+375291112217	support@transgarant.by
18	18	ГрузПартнер	Витебск, ул. Магистральная, 21	+375291112218	service@gruzpartner.by
19	19	Скоростная логистика	Брест, ул. Грузовая, 26	+375291112219	info@fastlogistics.by
20	20	БелТрансЭкспресс	Гродно, ул. Доставкаочная, 12	+375291112220	contact@beltransexpress.by
21	21	ТранспЛайн	Минск, ул. Грузовая, 37	+375291112221	support@transpline.by
22	22	ЭкспедиторСервис	Гомель, ул. Магистральная, 13	+375291112222	service@expeditor.by
23	23	ЛогистБел	Витебск, ул. Перевозочная, 18	+375291112223	contact@logistbel.by
24	24	ФрахтЛайн	Брест, ул. Центральная, 9	+375291112224	info@frachtline.by
25	25	МегаТранс	Гродно, ул. Логистическая, 15	+375291112225	support@megatrans.by
26	26	ЕвроТранс	Минск, ул. Международная, 29	+375291112226	service@eurotrans.by
27	27	Транспортное агентство	Гомель, ул. Транспортная, 20	+375291112227	contact@transportagency.by
28	28	Грузовая логистика	Витебск, ул. Погрузочная, 32	+375291112228	info@gruzlogistic.by
29	29	ФрахтБел	Брест, ул. Грузовая, 16	+375291112229	support@frachtbel.by
30	30	КаргоСервис	Гродно, ул. Транспортная, 23	+375291112230	service@cargoservice.by
31	31	БелАвтоЛогистик	Минск, ул. Автопарк, 5	+375291112231	info@belautolog.by
32	32	ГрузИнтерТранс	Гомель, ул. Грузовая, 10	+375291112232	support@gruzintertrans.by
33	33	Логистический Центр	Витебск, ул. Промышленная, 14	+375291112233	contact@logisticcenter.by

Рисунок 1.10 – Таблица «Клиенты» до скрипта

Таблица «Клиенты» после скрипта представлена на рисунке 1.11.

	Город text	Количество_операций bigint
1	Гомель	9
2	Минск	9

Рисунок 1.11 – Результат скрипта

## 1.5 Водители

Задание: Вывести возрастные категории водителей и количество водителей в каждой категории, где количество больше 10, отсортировать по убыванию.

Скрипт для вывода возрастных категорий водителей и количество водителей в каждой категории, где количество больше 10, отсортировать по убыванию.

```
SELECT
CASE
    WHEN "Возраст" < 30 THEN 'До 30 лет'
    WHEN "Возраст" BETWEEN 30 AND 40 THEN '30-40 лет'
    WHEN "Возраст" > 40 THEN 'Старше 40 лет'
END AS Возрастная_категория,
COUNT(*) AS Количество_водителей
FROM "Водители"
GROUP BY CASE
    WHEN "Возраст" < 30 THEN 'До 30 лет'
    WHEN "Возраст" BETWEEN 30 AND 40 THEN '30-40 лет'
    WHEN "Возраст" > 40 THEN 'Старше 40 лет'
END
HAVING COUNT(*) > 10
ORDER BY Количество_водителей DESC;
```

Таблица «Водители» до скрипта представлена на рисунке 1.12.

	id водителя [PK] integer	ФИО character varying (60)	Возраст integer	Стаж вождения integer	Номер водительского удостоверения character varying (30)	id транспортной операции integer
1	1	Иванов Иван Иванович	35	15	AB1234567	1
2	2	Петров Петр Петрович	40	20	CD2345678	2
3	3	Сидоров Алексей Николаевич	29	10	EF3456789	3
4	4	Козлов Дмитрий Олегович	45	25	GH4567890	4
5	5	Морозов Сергей Викторович	38	18	IJ5678901	5
6	6	Антонов Антон Антонович	33	12	KL6789012	6
7	7	Белов Виктор Сергеевич	50	30	MN7890123	7
8	8	Гаврилов Андрей Павлович	41	22	OP8901234	8
9	9	Дмитриев Олег Евгеньевич	36	17	QR9012345	9
10	10	Егоров Василий Николаевич	28	9	ST0123456	10
11	11	Жуков Михаил Петрович	47	26	UV1234567	11
12	12	Зайцев Николай Андреевич	32	14	WX2345678	12
13	13	Исеев Станислав Геннадьевич	44	23	YZ3456789	13
14	14	Капустин Федор Алексеевич	39	19	AB4567890	14
15	15	Лазарев Геннадий Владимирович	42	21	CD5678901	15
16	16	Михайлов Артем Сергеевич	31	11	EF6789012	16
17	17	Никитин Сергей Петрович	34	13	GH7890123	17
18	18	Орлов Виктор Андреевич	48	28	IJ8901234	18
19	19	Павлов Алексей Дмитриевич	37	16	KL9012345	19
20	20	Романов Артем Валентинович	29	10	MN0123456	20
21	21	Семенов Иван Григорьевич	46	27	OP1234567	21
22	22	Тихонов Олег Вячеславович	30	8	QR2345678	22
23	23	Ушаков Николай Игоревич	43	24	ST3456789	23
24	24	Федоров Петр Васильевич	35	15	UV4567890	24
25	25	Харитонов Максим Геннадьевич	49	29	WX5678901	25
26	26	Цветков Андрей Павлович	38	18	YZ6789012	26
27	27	Чернов Александр Олегович	45	26	AB7890123	27
28	28	Шестаков Виктор Аркадьевич	41	22	CD8901234	28
29	29	Щербаков Артем Владимирович	33	12	EF9012345	29
30	30	Юдин Олег Григорьевич	40	20	GH0123456	30
31	31	Яковлев Сергей Иванович	50	30	IJ1234567	31

Рисунок 1.12 – Таблица «Водители» до скрипта

Таблица «Водители» после скрипта представлена на рисунке 1.13.

	Возрастная_категория text	Количество_водителей bigint
1	30-40 лет	15
2	Старше 40 лет	13

Рисунок 1.13 – Результат скрипта

## 1.6 Оплаты

Задание: *Вывести максимальную сумму оплаты по каждому способу оплаты.*

Скрипт для вывода максимальной суммы оплаты по каждому способу оплаты.

```
SELECT MAX(Стоимость) AS максимальная_стоимость, Способ
FROM Оплата
GROUP BY Способ
```

Таблица «Оплаты» до скрипта представлена на рисунке 1.14.

	id оплаты [PK] integer	Сумма double precision	Дата оплаты date	Способ оплаты character varying (20)	id клиента integer	id транспортной операции integer
1	1	150	2025-03-01	Банковский перевод	1	1
2	2	250	2025-03-02	Наличные	2	2
3	3	300	2025-03-03	Кредитная карта	3	3
4	4	100	2025-03-04	Электронный кошелек	4	4
5	5	200	2025-03-05	Банковский перевод	5	5
6	6	350	2025-03-06	Наличные	6	6
7	7	120	2025-03-07	Кредитная карта	7	7
8	8	180	2025-03-08	Электронный кошелек	8	8
9	9	220	2025-03-09	Банковский перевод	9	9
10	10	160	2025-03-10	Наличные	10	10
11	11	240	2025-03-11	Кредитная карта	11	11
12	12	130	2025-03-12	Электронный кошелек	12	12
13	13	190	2025-03-13	Банковский перевод	13	13
14	14	210	2025-03-14	Наличные	14	14
15	15	300	2025-03-15	Кредитная карта	15	15
16	16	140	2025-03-16	Электронный кошелек	16	16
17	17	230	2025-03-17	Банковский перевод	17	17
18	18	260	2025-03-18	Наличные	18	18
19	19	280	2025-03-19	Кредитная карта	19	19
20	20	350	2025-03-20	Электронный кошелек	20	20
21	21	110	2025-03-21	Банковский перевод	21	21
22	22	150	2025-03-22	Наличные	22	22
23	23	200	2025-03-23	Кредитная карта	23	23
24	24	250	2025-03-24	Электронный кошелек	24	24
25	25	220	2025-03-25	Банковский перевод	25	25
26	26	280	2025-03-26	Наличные	26	26
27	27	300	2025-03-27	Кредитная карта	27	27
28	28	130	2025-03-28	Электронный кошелек	28	28
29	29	140	2025-03-29	Банковский перевод	29	29
30	30	160	2025-03-30	Наличные	30	30
31	31	190	2025-03-31	Кредитная карта	31	31

Рисунок 1.14 – Таблица «Оплаты» до скрипта

Таблица «Оплаты» после скрипта представлена на рисунке 1.15.

	Максимальная_сумма double precision	Способ оплаты character varying (20)
1	230	Банковский перевод
2	350	Электронный кошелек
3	350	Наличные
4	300	Кредитная карта

Рисунок 1.15– Результат скрипта

Задание: *Вывести сумму оплат, которые будут произведены после 2025-03-17, по каждому способу оплаты.*

Скрипт для вывода суммы оплат, которые будут произведены после 2025-03-17, по каждому способу оплаты.

```
SELECT SUM("Сумма") AS Суммарная_стоимость, "Способ оплаты"
FROM "Оплаты"
WHERE "Дата оплаты" > '2025-03-17'
GROUP BY "Способ оплаты";
```

Таблица «Оплаты» до скрипта представлена на рисунке 1.14.

Таблица «Оплаты» после скрипта представлена на рисунке 1.16.

	Суммарная_стоимость double precision	Способ оплаты character varying (20)
1	470	Банковский перевод
2	730	Электронный кошелек
3	850	Наличные
4	970	Кредитная карта

Рисунок 1.16– Результат скрипта

## 1.7 Маршруты

Задание: *Вывести среднее расстояние маршрутов между каждой парой городов, округленное до целого числа, отсортированные по убыванию.*

Скрипт для вывода среднего расстояния маршрутов между каждой парой городов, округленного до целого числа, отсортированные по убыванию.

```
SELECT
    "Точка отправления",
    "Точка прибытия",
    ROUND(AVG("Расстояние")) AS Среднее_расстояние
FROM "Маршруты"
```

GROUP BY "Точка отправления", "Точка прибытия"  
ORDER BY Среднее\_расстояние DESC;

Таблица «Маршруты» до скрипта представлена на рисунке 1.17.

	Id маршрута [PK] integer	Точка отправления character varying (20)	Точка прибытия character varying (20)	Расстояние double precision	Время в пути time without time zone
1	1	Минск	Гомель	300.5	05:30:00
2	2	Минск	Брест	400.2	06:00:00
3	3	Минск	Гродно	270.8	04:45:00
4	4	Минск	Витебск	250.3	04:00:00
5	5	Гомель	Минск	300.5	05:30:00
6	6	Гомель	Брест	450	07:00:00
7	7	Гомель	Гродно	330.4	06:10:00
8	8	Гомель	Витебск	380	06:40:00
9	9	Брест	Минск	400.2	06:00:00
10	10	Брест	Гомель	450	07:00:00
11	11	Брест	Гродно	160.3	03:20:00
12	12	Брест	Витебск	380.4	06:20:00
13	13	Гродно	Минск	270.8	04:45:00
14	14	Гродно	Гомель	330.4	06:10:00
15	15	Гродно	Брест	160.3	03:20:00
16	16	Гродно	Витебск	300	05:30:00
17	17	Витебск	Минск	250.3	04:00:00
18	18	Витебск	Гомель	380	06:40:00
19	19	Витебск	Брест	380.4	06:20:00
20	20	Витебск	Гродно	300	05:30:00
21	21	Минск	Смолевичи	40	00:45:00
22	22	Гомель	Смолевичи	320	05:00:00
23	23	Брест	Смолевичи	430	07:30:00
24	24	Гродно	Смолевичи	350	06:10:00
25	25	Витебск	Смолевичи	370	06:20:00
26	26	Минск	Прага	1500	20:00:00
27	27	Гомель	Прага	1600	21:00:00
28	28	Брест	Прага	1400	19:00:00
29	29	Гродно	Прага	1550	20:30:00
30	30	Витебск	Прага	1650	21:30:00
31	31	Минск	Берлин	1300	18:00:00
32	32	Гомель	Берлин	1400	19:00:00
33	33	Брест	Берлин	1200	17:30:00
34	34	Гродно	Берлин	1350	18:30:00
35	35	Витебск	Берлин	1450	19:30:00

Рисунок 1.17– Таблица «Маршруты» до скрипта

Таблица «Маршруты» после скрипта представлена на рисунке 1.18.

	Точка отправления character varying (20) 🔒	Точка прибытия character varying (20) 🔒	Среднее_расстояние double precision 🔒
1	Витебск	Прага	1650
2	Гомель	Прага	1600
3	Гродно	Прага	1550
4	Минск	Прага	1500
5	Витебск	Берлин	1450
6	Брест	Прага	1400
7	Гомель	Берлин	1400
8	Гродно	Берлин	1350
9	Минск	Берлин	1300
10	Брест	Берлин	1200
11	Гомель	Брест	450
12	Брест	Гомель	450
13	Брест	Смолевичи	430
14	Минск	Брест	400
15	Брест	Минск	400
16	Гомель	Витебск	380
17	Витебск	Брест	380
18	Брест	Витебск	380
19	Витебск	Гомель	380
20	Витебск	Смолевичи	370
21	Гродно	Смолевичи	350
22	Гродно	Гомель	330
23	Гомель	Гродно	330
24	Гомель	Смолевичи	320
25	Гродно	Витебск	300
26	Минск	Гомель	300
27	Гомель	Минск	300
28	Витебск	Гродно	300
29	Минск	Гродно	271
30	Гродно	Минск	271
31	Минск	Витебск	250
32	Витебск	Минск	250
33	Брест	Гродно	160
34	Гродно	Брест	160
35	Минск	Смолевичи	40

Рисунок 1.18 – Результат скрипта



Задание: *Вывести количество маршрутов для каждой точки отправления, где время в пути превышает 5 часов, отсортировать по возрастанию количества.*

Скрипт для вывода количества маршрутов для каждой точки отправления, где время в пути превышает 5 часов, отсортированные по возрастанию количества.

```
SELECT
    "Точка отправления",
    COUNT(*) AS Количество_маршрутов
FROM "Маршруты"
WHERE "Время в пути" > '05:00:00'
GROUP BY "Точка отправления"
ORDER BY Количество_маршрутов ASC;
```

Таблица «Маршруты» до скрипта представлена на рисунке 1.17.

Таблица «Маршруты» после скрипта представлена на рисунке 1.19.

	Точка отправления character varying (20)	Количество_маршрутов bigint
1	Минск	4
2	Гродно	5
3	Гомель	6
4	Витебск	6
5	Брест	6

Рисунок 1.19 – Результат скрипта

Задание: *Вывести 3 самых длинных маршрута по расстоянию для каждой точки отправления, с указанием точки прибытия и расстояния.*

Скрипт для вывода 3 самых длинных маршрутов по расстоянию для каждой точки отправления, с указанием точки прибытия и расстояния.

```
SELECT *
FROM (
    SELECT
        "Точка отправления",
        "Точка прибытия",
        "Расстояние",
        ROW_NUMBER() OVER (PARTITION BY "Точка отправления"
        ORDER BY "Расстояние" DESC) AS rank
    FROM "Маршруты"
) ranked_routes
WHERE rank <= 3;
```

Таблица «Маршруты» до скрипта представлена на рисунке 1.17.

Таблица «Маршруты» после скрипта представлена на рисунке 1.20.

	Точка отправления character varying (20) 🔒	Точка прибытия character varying (20) 🔒	Расстояние double precision 🔒	rank bigint 🔒
1	Брест	Прага	1400	1
2	Брест	Берлин	1200	2
3	Брест	Гомель	450	3
4	Витебск	Прага	1650	1
5	Витебск	Берлин	1450	2
6	Витебск	Брест	380.4	3
7	Гомель	Прага	1600	1
8	Гомель	Берлин	1400	2
9	Гомель	Брест	450	3
10	Гродно	Прага	1550	1
11	Гродно	Берлин	1350	2
12	Гродно	Смолевичи	350	3
13	Минск	Прага	1500	1
14	Минск	Берлин	1300	2
15	Минск	Брест	400.2	3

Рисунок 1.20 – Результат скрипта

## **ЗАКЛЮЧЕНИЕ**

В ходе выполнения лабораторной работы были успешно реализованы SQL-запросы для выборки данных из реляционной базы данных туристического агентства. Основной целью работы стало освоение базовых операций языка SQL, включая фильтрацию, сортировку, соединение таблиц и использование скалярных функций.

Ключевые результаты:

1. Фильтрация и сортировка данных.
2. Соединение таблиц (JOIN).
3. Работа с датами.
4. Валидация данных.

Работа подтвердила, что грамотное проектирование структуры базы данных и использование индексов значительно упрощают выполнение сложных запросов. Освоенные навыки позволяют эффективно извлекать и анализировать данные, что критически важно для автоматизации процессов в туристическом бизнесе (сбор информации о клиентах и сотрудниках, управление платежами, планирование ресурсов).

Выполненная работа стала важным этапом в освоении технологий работы с базами данных и заложила основу для изучения более сложных SQL-операций, таких как группировка, агрегация и оптимизация запросов.