

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЕТ

по лабораторной работе №4-5
на тему

РЕАЛИЗАЦИЯ SQL-ЗАПРОСОВ НА ВЫБОРКУ ДАННЫХ

Студент:

Кутняк А. В.

Руководитель:

Игнатович А. О.

Минск 2024

1 ЦЕЛЬ

1.1 Лабораторная работа №4

Цель лабораторной работы – реализация простых запросов на выборку данных на языке SQL с использованием предложений SELECT, FROM (JOINS), WHERE и ORDER BY оператора SELECT.

Необходимо реализовать не менее 15 запросов с использованием каждого из следующих операторов не менее двух раз: LIKE, ORDER BY, CROSS JOIN, INNER\OUTER JOIN, LEFT\RIGHT\FULL JOIN, DISTINCT\UNIQUE, ALL, ASC, DESC, NULL.

1.2 Лабораторная работа №5

Цель лабораторной работы – реализация запросов на выборку данных на языке SQL с использованием подзапросов, агрегатных функций, а также группировки данных (предложение GROUP BY оператора SELECT) и операций над множествами (UNION, INTERSECT, MINUS).

Необходимо реализовать не менее 15 запросов с использованием каждого из следующих операторов не менее двух раз: UNION, INTERSECT, AVG, COUNT, SUM\MINUS, MIN\MAX, IN, EXISTS, AS, WITH.

2 ХОД ВЫПОЛНЕНИЯ

2.1 Лабораторная работа №4

2.1.1 Получить список всех двухместных номеров

```
select all
    name as "Apartments",
    cost as "Cost"
from
    apartment_type
where
    capacity = 2;
```

	□ "Apartments"	□ "Cost"
1	Двухместный однокомнатный номер «Твин»	95
2	Двухместный двухкомнатный «Дабл»	120
3	Двухместный трехкомнатный «Апартаменты»	280

Рисунок 2.1 – Результат выполнения запроса

2.1.2 Найти открытые акционерные общества среди компаний

```
select all
  name as "Company",
  taxpayer_id as "Taxpayer ID"
from
  company_identity
where
  name like '000 %'
order by
  company_identity;
```

	<input type="checkbox"/> "Company"	<input type="checkbox"/> "Taxpayer ID"
1	000 "СИЛЬВЕР СОЛЮШНЗ"	0xxxxxxxx

Рисунок 2.2 – Результат выполнения запроса

2.1.3 Найти даты проживания всех постояльцев

```
select
    individual.name as "Guest",
    reservation.check_in as "Check in date",
    reservation.check_out as "Check out date"
from
    individual_identity individual
inner join
    guest on individual.id = guest.individual_id
inner join
    apartment_reservation reservation on
    guest.reservation_id = reservation.id
order by
    individual.name;
```

	"Guest"	"Check in date"	"Check out date"
1	ГОГОЛЬ НИКОЛАЙ ВАСИЛЬЕВИЧ	2024-02-03	2024-02-06
2	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	2024-02-09	2024-02-16
3	МАЯКОВСКИЙ ВЛАДИМИР ВЛАДИМИРОВИЧ	2024-02-09	2024-02-16
4	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	2024-02-02	2024-02-09
5	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	2024-02-03	2024-02-06
6	ФЕТ АФАНАСИЙ АФАНАСЬЕВИЧ	2024-02-14	2024-02-29
7	ЦВЕТАЕВА МАРИНА ИВАНОВНА	2024-02-09	2024-02-16

Рисунок 2.3 – Результат выполнения запроса

2.1.4 Сравнить все возможные виды апартаментов

```

select
    A.name as "First",
    B.name as "Second",
    A.cost || ' - ' ||
    B.cost || ' = ' ||
    A.cost - B.cost as "dCost"
from
    apartment_type A
cross join
    apartment_type B
where
    A.id != B.id and A.cost - B.cost >= 0;

-- 5.
select
    apartment.no as "No",
    apartment_type.name as "Type"
from
    apartment
left join
    apartment_type on apartment.type_id = apartment_type.id
where
    apartment.free = true;

```

	□ "First"	□ "Second"	□ "dCost"
1	Одноместный однокомнатный номер «Сингл А»	Одноместный однокомнатный номер «Сингл В»	70 - 50 = 20
2	Двухместный однокомнатный номер «Твин»	Одноместный однокомнатный номер «Сингл А»	95 - 70 = 25
3	Двухместный однокомнатный номер «Твин»	Одноместный однокомнатный номер «Сингл В»	95 - 50 = 45
4	Двухместный двухкомнатный «Дабл»	Одноместный однокомнатный номер «Сингл А»	120 - 70 = 50
5	Двухместный двухкомнатный «Дабл»	Одноместный однокомнатный номер «Сингл В»	120 - 50 = 70
6	Двухместный двухкомнатный «Дабл»	Двухместный однокомнатный номер «Твин»	120 - 95 = 25
7	Двухместный двухкомнатный «Дабл»	Трёхместный двухкомнатный номер	120 - 115 = 5
8	Двухместный трехкомнатный «Апартаменты»	Одноместный однокомнатный номер «Сингл А»	280 - 70 = 210
9	Двухместный трехкомнатный «Апартаменты»	Одноместный однокомнатный номер «Сингл В»	280 - 50 = 230
10	Двухместный трехкомнатный «Апартаменты»	Двухместный однокомнатный номер «Твин»	280 - 95 = 185
11	Двухместный трехкомнатный «Апартаменты»	Двухместный двухкомнатный «Дабл»	280 - 120 = 160
12	Двухместный трехкомнатный «Апартаменты»	Трёхместный двухкомнатный номер	280 - 115 = 165
13	Трёхместный двухкомнатный номер	Одноместный однокомнатный номер «Сингл А»	115 - 70 = 45
14	Трёхместный двухкомнатный номер	Одноместный однокомнатный номер «Сингл В»	115 - 50 = 65
15	Трёхместный двухкомнатный номер	Двухместный однокомнатный номер «Твин»	115 - 95 = 20

Рисунок 2.4 – Результат выполнения запроса

2.1.5 Получить информацию о свободных номерах

```
select
    apartment.no as "No",
    apartment_type.name as "Type"
from
    apartment
left join
    apartment_type on apartment.type_id = apartment_type.id
where
    apartment.free = true;
```

	□ "No" ▾	□ "Type" ▾
1	101	Одноместный однокомнатный номер «Сингл А»
2	402	Двухместный двухкомнатный «Дабл»
3	603	Трёхместный двухкомнатный номер

Рисунок 2.5 – Результат выполнения запроса

2.1.6 Получить информацию о бронировании номеров

```
select
    apartment.no as "No",
    apartment_type.name as "Type",
    case
        when customer.individual_id is not null then individual.name
        when customer.company_id is not null then company.name
        end as "Customer",
    reservation.date as "Reservation date"
from
    customer
full outer join
    apartment_reservation reservation on
    customer.id = reservation.customer_id
full outer join
    apartment on reservation.apartment_no = apartment.no
left join
    apartment_type on apartment.type_id = apartment_type.id
left join
    individual_identity individual on
    customer.individual_id = individual.id
left join
    company_identity company on
    customer.company_id = company.id
order by
    reservation.date desc;
```

	№ "No"	№ "Type"	№ "Customer"	№ "Reservation date"
1	502	Двухместный трехкомнатный «Апартаменты»	<null>	<null>
2	302	Двухместный однокомнатный номер «Твин»	<null>	<null>
3	101	Одноместный однокомнатный номер «Сингл А»	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	2024-03-01
4	201	Одноместный однокомнатный номер «Сингл В»	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	2024-02-12
5	603	Трёхместный двухкомнатный номер	ООО "СИЛЬВЕР СОЛЮШНЗ"	2024-02-04
6	402	Двухместный двухкомнатный «Дабл»	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	2024-02-02
7	101	Одноместный однокомнатный номер «Сингл А»	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	2024-02-01

Рисунок 2.6 – Результат выполнения запроса

2.1.7 Получить список апартаментов с кондиционером

```
select
    apartment.no as "No",
    apartment_type.name as "Type"
from
    apartment
inner join
    apartment_type on apartment.type_id = apartment_type.id
where
    apartment_type.description like '%' || lower('...') || '%';
```

	<input type="checkbox"/> "No"	<input type="checkbox"/> "Type"
1	101	Одноместный однокомнатный номер «Сингл А»
2	302	Двухместный однокомнатный номер «Твин»
3	402	Двухместный двухкомнатный «Дабл»
4	502	Двухместный трехкомнатный «Апартаменты»

Рисунок 2.7 – Результат выполнения запроса

2.1.8 Получить информацию об услугах, которые никогда не оказывались

```
select
    service.name as "Service",
    service.cost as "Cost"
from
    service
left join
    service_order on service.id = service_order.service_id
where
    service_order.service_id is null
order by
    service.cost desc;
```

	"Service"	"Cost"
1	Абонемент фитнес-зала (7)	40
2	Прокат детской кровати	20
3	Абонемент фитнес-зала (1)	12

Рисунок 2.8 – Результат выполнения запроса

2.1.9 Получаем информацию о номерах, которые когда-либо бронировались

```
select distinct
    apartment.no as "No",
    apartment_type.name as "Type"
from
    apartment_reservation reservation
left join
    apartment on reservation.apartment_no = apartment.no
left join
    apartment_type on apartment.type_id = apartment_type.id;
```

	□ "No" ⇅	□ "Type" ⇅
1	603	Трёхместный двухкомнатный номер
2	402	Двухместный двухкомнатный «Дабл»
3	201	Одноместный однокомнатный номер «Сингл В»
4	101	Одноместный однокомнатный номер «Сингл А»

Рисунок 2.9 – Результат выполнения запроса

2.1.10 Получить информацию об услугах, которые когда-либо заказывались

```
select distinct
    service.name as "Service",
    service.cost as "Cost"
from
    service
right join
    service_order s_order on service.id = s_order.service_id;
```

	"Service"	"Cost"
1	Дополнительная уборка номера	15
2	Услуги прачечной	10
3	Экскурсия в ХХХ	40
4	Дополнительный завтрак	12
5	Вызов такси к гостинице	14

Рисунок 2.10 – Результат выполнения запроса

2.1.11 Найти все номера с «оригинальными» названиями

```
select
    apartment_type.name as "Type"
from
    apartment_type
where
    apartment_type.name like '%<%>%';
```

	Type
1	Одноместный однокомнатный номер «Сингл А»
2	Одноместный однокомнатный номер «Сингл В»
3	Двухместный однокомнатный номер «Твин»
4	Двухместный двухкомнатный «Дабл»
5	Двухместный трехкомнатный «Апартаменты»

Рисунок 2.11 – Результат выполнения запроса

2.1.12 Найти все пересекающиеся бронирования

```
select
    A.check_in || ' - ' || A.check_out
        as "First",
    B.check_in || ' - ' || B.check_out
        as "Second",
    B.check_in || ' - ' || least(A.check_out, B.check_out)
        as "Intersection"
from
    apartment_reservation A
cross join
    apartment_reservation B
where
    A.id != B.id
        and
    A.check_in <= B.check_in
        and
    B.check_in <= A.check_out;
```

	📄 "First"	📄 "Second"	📄 "Intersection"
1	2024-02-02 - 2024-02-09	2024-02-03 - 2024-02-06	2024-02-03 - 2024-02-06
2	2024-02-02 - 2024-02-09	2024-02-09 - 2024-02-16	2024-02-09 - 2024-02-09
3	2024-02-09 - 2024-02-16	2024-02-14 - 2024-02-29	2024-02-14 - 2024-02-16

Рисунок 2.12 – Результат выполнения запроса

2.1.13 Найти постояльцев, проживающих в апартаментах, заказанных компанией

```
select
    individual.name as "Guest",
    apartment.no as "No",
    company.name as "Customer"
from
    guest
inner join
    apartment_reservation reservation on
    guest.reservation_id = reservation.id
inner join
    customer on
    reservation.customer_id = customer.id
inner join
    company_identity company on
    customer.company_id = company.id
inner join
    individual_identity individual on
    guest.individual_id = individual.id
inner join
    apartment on
    reservation.apartment_no = apartment.no;
```

	□ "Guest"	□ "No"	□ "Customer"
1	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	603	000 "СИЛЬВЕР СОЛЮШНЗ"
2	МАЯКОВСКИЙ ВЛАДИМИР ВЛАДИМИРОВИЧ	603	000 "СИЛЬВЕР СОЛЮШНЗ"
3	ЦВЕТАЕВА МАРИНА ИВАНОВНА	603	000 "СИЛЬВЕР СОЛЮШНЗ"

Рисунок 2.13 – Результат выполнения запроса

2.1.14 Найти все длительные бронирования

```
select
    reservation.apartment_no
        as "No",
    reservation.check_in || ' - ' || reservation.check_out
        as "Period"
from
    apartment_reservation reservation
where
    reservation.check_out - reservation.check_in >= 7;
```

	<input type="checkbox"/> "No"	<input type="checkbox"/> "Period"
1	101	2024-02-02 - 2024-02-09
2	603	2024-02-09 - 2024-02-16
3	201	2024-02-14 - 2024-02-29

Рисунок 2.14 – Результат выполнения запроса

2.1.15 Получить список услуг, оказываемых в день заказа

```
select distinct
    service.name as "Service",
    service_order.service_date as "Date"
from
    service_order
inner join
    service on service_order.service_id = service.id
where
    service_order.order_date = service_order.service_date
order by
    service_order.service_date, service.name;
```

	"Service"	"Date"
1	Дополнительная уборка номера	2024-02-04
2	Вызов такси к гостинице	2024-02-12

Рисунок 2.15 – Результат выполнения запроса

2.2 Лабораторная работа №5

2.2.1 Получить список всех идентификационных записей

```
select
    name as "Name",
    passport_id as "Id"
from
    individual_identity
union
select
    name as "Name",
    taxpayer_id as "Id"
from
    company_identity;
```

	<input type="checkbox"/> "Name"	<input type="checkbox"/> "Id"
1	ЦВЕТАЕВА МАРИНА ИВАНОВНА	6xxxxxxxxxxxxx
2	ФЕТ АФАНАСИЙ АФАНАСЬЕВИЧ	7xxxxxxxxxxxxx
3	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	1xxxxxxxxxxxxx
4	ГОГОЛЬ НИКОЛАЙ ВАСИЛЬЕВИЧ	3xxxxxxxxxxxxx
5	ЗАО "ДАММИ СЭМПЛ"	1xxxxxxxx
6	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	2xxxxxxxxxxxxx
7	МАЯКОВСКИЙ ВЛАДИМИР ВЛАДИМИРОВИЧ	5xxxxxxxxxxxxx
8	ООО "СИЛЬВЕР СОЛЮШНЗ"	0xxxxxxxx
9	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	4xxxxxxxxxxxxx

Рисунок 2.16 – Результат выполнения запроса

2.2.2 Получить количество бронирований для каждого номера

```
select
    apartment_no as "No",
    count(*) as "Reservations"
from
    apartment_reservation
group by
    apartment_no;
```

	<input type="checkbox"/> "No" ▾	<input type="checkbox"/> "Reservations" ▾
1	101	2
2	201	1
3	402	1
4	603	1

Рисунок 2.17 – Результат выполнения запроса

2.2.3 Получить список самых дешевых типов апартаментов

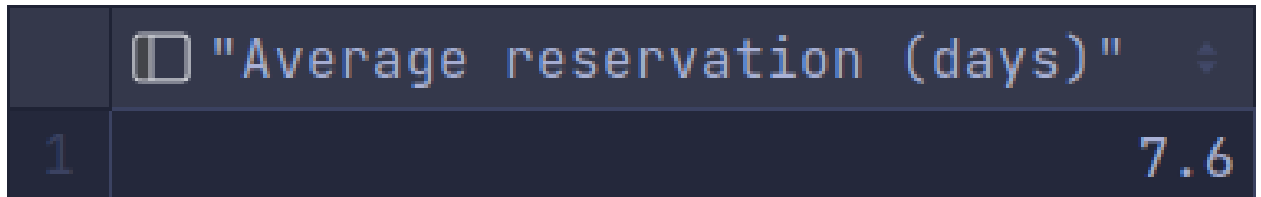
```
select
    name as "Apartment",
    cost as "Cost"
from
    apartment_type
where
    cost = (select min(cost) from apartment_type);
```

	□ "Apartment"	□ "Cost"
1	Одноместный однокомнатный номер «Сингл В»	50

Рисунок 2.18 – Результат выполнения запроса

2.2.4 Получить среднюю длительность каждого бронирования

```
select
    avg(reservation.check_out - reservation.check_in)
    as "Average reservation (days)"
from
    apartment_reservation reservation;
```



The screenshot shows a database query result in a dark-themed interface. The top row is a header with a small square icon and the text "Average reservation (days)". The bottom row is a data row with the number "1" in the first column and the value "7.6" in the second column.

	"Average reservation (days)"
1	7.6

Рисунок 2.19 – Результат выполнения запроса

2.2.5 Получить информацию об оплаченных услугах

```
with
  paid_services as (
    select
      service_order.order_date,
      service_order.service_date,
      service.cost
    from
      service_order
    join
      service on service_order.service_id = service.id
    where
      paid = true
  )
select
  count(*) as "Count",
  avg(service_date - order_date) as "Avg wait time (days)",
  sum(cost) as "Total cost"
from
  paid_services;
```

	☐ "Count"	☐ "Average wait time (days)"	☐ "Total cost"
1	4	0.5	51

Рисунок 2.20 – Результат выполнения запроса

2.2.6 Получись данные постояльцев-заказчиков

```
select
    name as "Customer",
    passport_id as "Passport ID"
from
    individual_identity
where
    id in (
        select individual_id from guest
        intersect
        select customer.individual_id from customer
    );
```

	☐ "Customer"	☐ "Passport ID"
1	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	2xxxxxxxxxxxxxx
2	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	4xxxxxxxxxxxxxx
3	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	1xxxxxxxxxxxxxx

Рисунок 2.21 – Результат выполнения запроса

2.2.7 Получить информацию о самых дорогих услугах

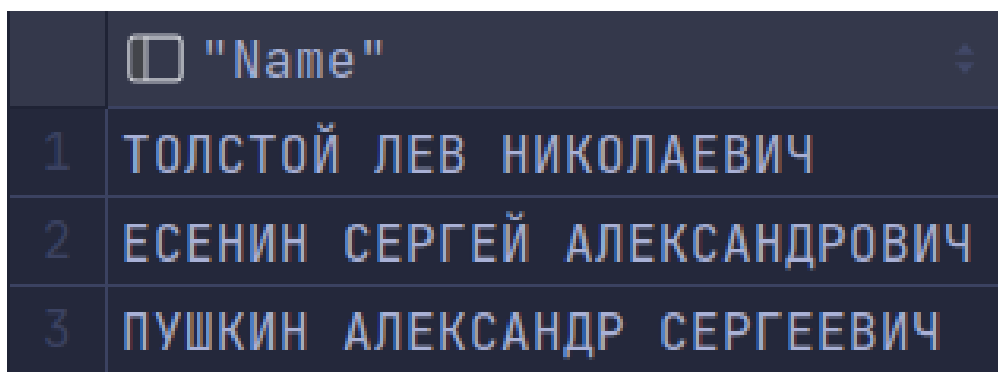
```
select
    service.name as "Name",
    service.description as "Description",
    service.cost as "Cost"
from
    service
where
    service.cost = (select max(cost) from service);
```

	"Name"	"Description"	"Cost"
1	Экскурсия в ХХХ	Групповая поездка с гидом на экскурсию в ХХХ	40
2	Абонемент фитнес-зала (7)	Недельный абонемент в фитнес-зал	40

Рисунок 2.22 – Результат выполнения запроса

2.2.8 Получить список постояльцев, совершивших заказ апартаментов и услуг

```
with
reservation_customers as (
    select individual_id from customer
    where individual_id is not null
),
service_customers as (
    select individual_id from guest
    join service_order on guest.id = service_order.guest_id
)
select
    name as "Name"
from
    individual_identity
where
    id in (
        select individual_id from reservation_customers
        intersect
        select individual_id from service_customers
    );
```



	<input type="checkbox"/> "Name" ⬇
1	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ
2	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ
3	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ

Рисунок 2.23 – Результат выполнения запроса

2.2.9 Получить транзакции всех заказчиков

```
with
reservation_customers as (
    select name from individual_identity individual
    join customer on individual.id = customer.individual_id
    union
    select name from company_identity company
    join customer on company.id = customer.company_id
),
service_customers as (
    select name from individual_identity individual
    where id in (
        select individual_id from guest
        join service_order on
        guest.id = service_order.guest_id
    )
)
select
    name as "Name",
    'Apartment reservation' as "Transaction type"
from
    reservation_customers
union
select name, 'Service order' from service_customers;
```

	<input type="checkbox"/> "Name"	<input type="checkbox"/> "Transaction type"
1	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	Service order
2	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	Service order
3	ООО "СИЛЬВЕР СОЛЮШНЗ"	Apartment reservation
4	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	Apartment reservation
5	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ	Apartment reservation
6	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ	Apartment reservation
7	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ	Service order
8	ЦВЕТАЕВА МАРИНА ИВАНОВНА	Service order
9	ГОГОЛЬ НИКОЛАЙ ВАСИЛЬЕВИЧ	Service order

Рисунок 2.24 – Результат выполнения запроса

2.2.10 Получить список физических лиц, совершивших заказы

```
select
    name
from
    individual_identity individual
where
    exists(
        select * from customer
        join apartment_reservation on
        customer.id = apartment_reservation.customer_id
        where customer.individual_id = individual.id
    )
    or
    exists(
        select * from guest
        join service_order on
        guest.id = service_order.guest_id
        where guest.individual_id = individual.id
    );
```

	name
1	ПУШКИН АЛЕКСАНДР СЕРГЕЕВИЧ
2	ТОЛСТОЙ ЛЕВ НИКОЛАЕВИЧ
3	ГОГОЛЬ НИКОЛАЙ ВАСИЛЬЕВИЧ
4	ЕСЕНИН СЕРГЕЙ АЛЕКСАНДРОВИЧ
5	ЦВЕТАЕВА МАРИНА ИВАНОВНА

Рисунок 2.25 – Результат выполнения запроса

2.2.11 Получить количество постояльцев от каждой компании

```
select
    name as "Company",
    count(guest.id) as "Guests count"
from
    guest
join
    apartment_reservation on
    guest.reservation_id = apartment_reservation.id
join
    customer on
    apartment_reservation.customer_id = customer.id
right join
    company_identity on
    customer.company_id = company_identity.id
group by
    company_identity.name;
```

	□ "Company"	□ "Guests count"
1	ЗАО "ДАММИ СЭМПЛ"	0
2	ООО "СИЛЬВЕР СОЛЮШНЗ"	3

Рисунок 2.26 – Результат выполнения запроса

2.2.12 Вывести типы апартаментов, которые бронировались на неделю

```
with
  week_reservations as (
    select apartment_no from apartment_reservation
    where check_out - check_in >= 7
  ),
  week_apartments as (
    select * from apartment
    where apartment.no in
      (select apartment_no from week_reservations)
  )
select
  name as "Name",
  description as "Description",
  cost as "Cost"
from
  apartment_type
where
  id in (select type_id from week_apartments);
```

	□ "Name"	□ "Description"	□ "Cost"
1	Одноместный однокомнатный номер «Сингл А»	Номер повышенной комфортности.В номере: кровать 140x20...	70
2	Одноместный однокомнатный номер «Сингл В»	Номер, предназначенный для одноместного размещения. В ...	50
3	Трёхместный двухкомнатный номер	Номер, предназначенный для размещения трех человек. Да...	115

Рисунок 2.27 – Результат выполнения запроса

2.2.13 Получить количество постояльцев в каждом номере

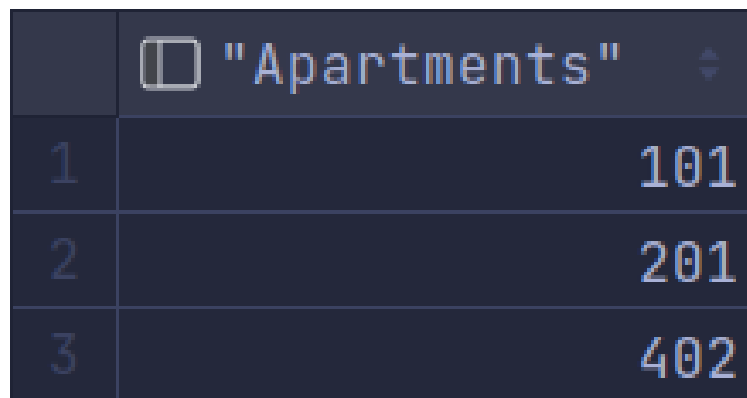
```
select
    apartment_no as "Apartment",
    count(*) as "Guests"
from
    guest
join
    apartment_reservation reservation on
    guest.reservation_id = reservation.id
group by
    reservation.apartment_no;
```

	<input type="checkbox"/> "Apartment"	^	<input type="checkbox"/> "Guests"	↕
1		101		1
2		201		1
3		402		2
4		603		3

Рисунок 2.28 – Результат выполнения запроса

2.2.14 Получить номера, заказанные физическими лицами

```
select
    no as "Apartments"
from
    apartment
where
    exists(
        select *
        from
            apartment_reservation reservation
        join
            customer on reservation.customer_id = customer.id
        where
            reservation.apartment_no = apartment.no
            and
            customer.individual_id is not null
    );
```

A screenshot of a database query result displayed in a dark-themed interface. The result is a table with two columns. The first column contains the numbers 1, 2, and 3. The second column contains the values 101, 201, and 402. Above the table, there is a header row with a small square icon followed by the text "Apartments" and a dropdown arrow.


	 "Apartments" ▾
1	101
2	201
3	402

Рисунок 2.29 – Результат выполнения запроса

2.2.15 Получить общее число дней оплаченных бронирований

```
select
    count(*) as "Reservations count",
    sum(check_out - check_in) as "Total days count"
from
    apartment_reservation
where
    apartment_reservation.paid = true;
```

	<input type="checkbox"/> "Reservations count"	:	<input type="checkbox"/> "Total days count"	:
1	4		32	

Рисунок 2.30 – Результат выполнения запроса