

Министерство цифрового развития

Федеральное государственное бюджетное образовательное учреждение высшего
образования

«Сибирский государственный университет телекоммуникаций
и информатики» (СибГУТИ)

Кафедра прикладной математики и кибернетики

Отчёт
по Расчётно-графической работе

Вариант 24
“Фитнес-студия”

Выполнил:
студент группы ИП-216
Русецкий А.С.

Работу проверил:
Старший преподаватель
Дьячкова Марина Сергеевна

Новосибирск 2025 г.

1. Описание базы данных

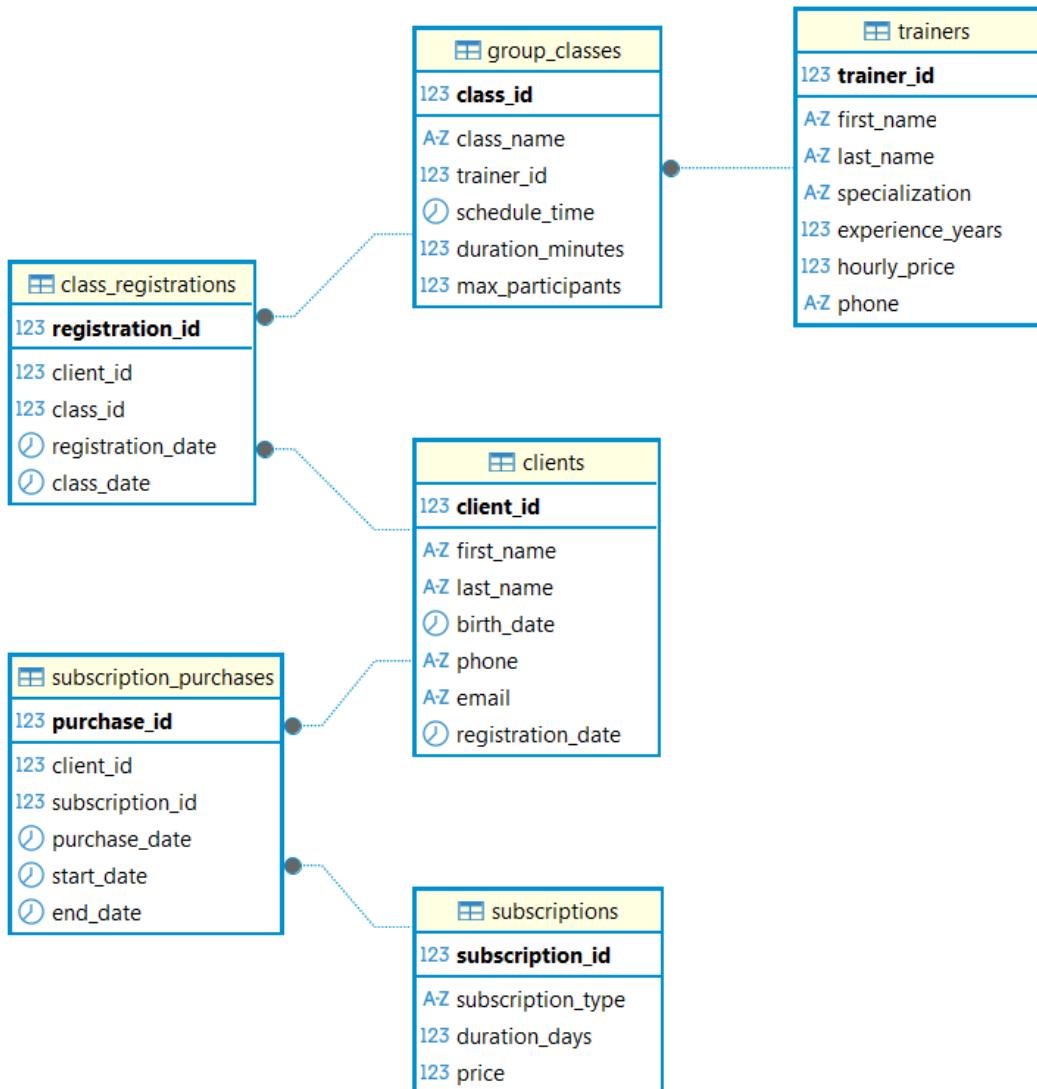


Таблица 1. Сущности и связи

Сущность 1	Сущность 2	Тип связи (1:1, 1:M, M:N)	Описание связи
Тренеры	Групповые занятия	1:M	Один тренер проводит несколько групповых занятий
Клиенты	Покупки абонементов	1:M	Один клиент может совершить несколько покупок абонементов
Клиенты	Записи на занятия	1:M	Один клиент может записаться на несколько групповых занятий
Абонементы	Покупки абонементов	1:M	Одни типы абонемента могут быть проданы несколько раз
Групповые занятия	Записи на занятия	1:M	На одно групповое занятие может записаться несколько клиентов

Таблица 2. Атрибуты сущностей

Сущность, название таблицы в БД	Название атрибута	Название атрибута в БД (имя поля)	Тип данных	Not Null	Primary key	Foreign key	Дополнительные ограничения целостности
Тренеры trainers	Номер тренера Имя тренера Фамилия тренера Специализация Опыт Почасовая ставка Телефон	trainer_id first_name last_name specialization experience_years hourly_price phone	Int (serial) varchar(50) varchar(50) varchar(100) int decimal(10,2) varchar(20)	1 1 1 1 1 1 1	P		>= 0 > 0
Клиенты clients	Номер клиента Имя клиента Фамилия клиента Дата рождения Телефон Email Дата регистрации	client_id first_name last_name birth_date phone email registration_date	Int (serial) varchar(50) varchar(50) date varchar(20) varchar(100) date	1 1 1 1 1 1 1	P		Проверка формата email
Абонементы subscriptions	Номер абонемента Тип абонемента Продолжительность Стоимость	subscription_id subscription_type duration_days price	Int (serial) varchar(50) int decimal(10,2)	1 1 1 1	P		>= 0 >= 0
Групповые занятия group_classes	Номер занятия Название занятия Номер тренера Время проведения	class_id class_name trainer_id schedule_time	Int (serial) varchar(100) int time	1 1 1 1	P	F	

	Продолжительность Максимальное кол-во участников	duration_minutes max_participants	int int	1 1			>0 >0
Покупки абонементов subscription_purchases	Номер покупки Номер клиента Номер абонемента Дата покупки Дата начала Дата окончания	purchase_id client_id subscription_id purchase_date start_date end_date	Int (serial) int int date date date	1 1 1 1 1 1	P F F		
Записи на занятия class_registrations	Номер записи Номер клиента Номер занятия Дата регистрации Дата занятия	registration_id client_id class_id registration_date class_date	Int (serial) int int date date	1 1 1 1 1	P F F		“end_date” > “start_date”

1.2. Данные БД (содержимое таблиц)

Таблица trainers

	123 ↗ trainer_id	AZ first_name	AZ last_name	AZ specialization	123 experience_years	123 hourly_price	Az phone
Таблица	1	Артём	Русецкий	Силовые тренировки	5	2 000	+7-903-111-2233
Текст	2	Эмил	Алиев	Йога	7	1 500	+7-916-222-3344
Текст	3	Иван	Разорвин	Стретчинг	3	1 200	+7-925-333-4455
Текст	4	Андрей	Смаев	Кроссфит	6	1 800	+7-903-444-5566
Текст	5	Нарек	Марянин	Пилатес	4	1 300	+7-916-555-6677

Таблица clients

	123 ↗ client_id	AZ first_name	AZ last_name	⌚ birth_date	Az phone	Az email	⌚ registration_date
Таблица	1	Артур	Литвинов	2004-11-01	+7-903-666-7788	artur@mail.ru	2025-01-10
Текст	2	Филипп	Андрющенко	2004-08-22	+7-916-777-8899	filipp@gmail.com	2025-02-15
Текст	3	Михаил	Горбунов	2000-01-01	+7-925-888-9900	gorbunov@mail.ru	2025-01-20
Текст	4	Иван	Лобода	2005-05-14	+7-903-999-0011	loboda@gmail.com	2025-03-05
Текст	5	Павел	Швецов	2006-07-18	+7-916-000-1122	pashka@mail.ru	2025-02-28

Таблица subscriptions

	123 ↗ subscription_id	AZ subscription_type	123 duration_days	123 price
Таблица	1	Разовый визит	1	500
Текст	2	Месячный стандарт	30	3 000
Текст	3	Месячный премиум	30	5 000
Текст	4	Квартальный	90	8 000
Текст	5	Годовой	365	25 000

Таблица group_classes

	123 ↗ class_id	AZ class_name	123 ↗ trainer_id	⌚ schedule_time	123 duration_minutes	123 max_participants
Таблица	1	Утренняя йога	2	08:00:00	60	15
Текст	2	Силовая тренировка	1	18:00:00	90	10
Текст	3	Пилатес для начинающих	5	17:00:00	60	12
Текст	4	Кроссфит	4	19:00:00	75	8
Текст	5	Вечерний стретчинг	3	20:00:00	45	20
Текст	6	Йога для продвинутых	2	19:00:00	75	12
Текст	7	Вечерняя йога	2	20:30:00	60	15
Текст	8	Силовой тренинг для начинающих	1	09:00:00	60	8
Текст	9	Функциональный тренинг	1	17:00:00	90	10
Текст	10	Пилатес для беременных	5	10:00:00	45	6
Текст	11	Стретчинг утренний	3	07:00:00	30	25

Таблица subscription_purchases

	123 ↗ purchase_id	123 ↗ client_id	123 ↗ subscription_id	⌚ purchase_date	⌚ start_date	⌚ end_date
Таблица	1	1	2	2025-09-01	2025-09-01	2025-10-01
Текст	2	2	3	2025-09-05	2025-09-05	2025-10-05
Текст	3	3	1	2025-09-10	2025-09-10	2025-09-11
Текст	4	4	4	2025-09-15	2025-09-15	2025-12-15
Текст	5	5	2	2025-09-20	2025-09-20	2025-10-20
Текст	6	6	1	2025-09-25	2025-09-25	2025-09-26

Таблица class_registrations

SQL select * from class_registrations | Введите SQL выражение чтобы отфильтровать результаты

Таблица	registration_id	client_id	class_id	registration_date	class_date
1	1	1	1	2025-09-26	2025-09-27
2	2	1	3	2025-09-26	2025-09-28
3	3	2	2	2025-09-27	2025-09-28
4	4	2	4	2025-09-27	2025-09-29
5	5	3	1	2025-09-28	2025-09-29
6	6	4	5	2025-09-29	2025-09-30
7	7	5	3	2025-09-30	2025-10-01

2. Запросы:

1.1 Вывести имя и фамилию клиентов, которые купили более одного абонемента

```
select first_name, last_name from clients
where client_id in (
    select client_id from subscription_purchases
    group by client_id
    having count(purchase_id) > 1
);
```

SQL select first_name, last_name from clients where | Введе

Таблица	A-Z first_name	A-Z last_name
1	Артур	Литвинов

1.2 Вывести тренеров, которые проводят более 2 групповых занятий

```
select first_name, last_name, specialization from trainers
where trainer_id in (
    select trainer_id from group_classes
    group by trainer_id
    having count(class_id) > 2
);
```

SQL select first_name, last_name, specialization from | Введе

Таблица	A-Z first_name	A-Z last_name	A-Z specialization
1	Артём	Русецкий	Силовые тренировки
2	Эмил	Алиев	Йога

2.1 Вывести имена и фамилии клиентов, названия групповых занятий и даты занятий для всех записей на занятия

```
select clients.first_name, clients.last_name,
       group_classes.class_name, class_registrations.class_date
  from clients
 join class_registrations ON clients.client_id =
 class_registrations.client_id
 join group_classes ON class_registrations.class_id =
 group_classes.class_id;
```

Текст

select clients.first_name, clients.last_name, gr | Введите SQL выражение чтобы отфильтровать результат

1	Артур	Литвинов	Утренняя йога	2025-09-27
2	Артур	Литвинов	Пилатес для начинающих	2025-09-28
3	Филипп	Андрющенко	Силовая тренировка	2025-09-28
4	Филипп	Андрющенко	Кроссфит	2025-09-29
5	Михаил	Горбунов	Утренняя йога	2025-09-29
6	Иван	Лобода	Вечерний стретчинг	2025-09-30
7	Павел	Швецов	Пилатес для начинающих	2025-10-01

2.2 Вывести информацию о тренерах, названиях групповых занятий и общее количество записей на эти занятия

```
select trainers.first_name, trainers.last_name,
       group_classes.class_name,
       count(class_registrations.registration_id) as
total_registrations
  from trainers
 join group_classes on trainers.trainer_id =
 group_classes.trainer_id
 join class_registrations on group_classes.class_id =
 class_registrations.class_id
 group by trainers.first_name, trainers.last_name,
          group_classes.class_name,
          trainers.trainer_id, group_classes.class_id;
```

Текст

select trainers.first_name, trainers.last_name, gr | Введите SQL выражение чтобы отфильтровать результаты

1	Нарек	Марянян	Пилатес для начинающих	2
2	Артём	Русецкий	Силовая тренировка	1
3	Эмил	Алиев	Утренняя йога	2
4	Андрей	Смаев	Кроссфит	1
5	Иван	Разорвин	Вечерний стретчинг	1

3.1 Создать представление, которое показывает тренеров, у которых общее количество записей на их групповые занятия превышает среднее количество записей на одного тренера в фитнес-студии.

```
CREATE VIEW popular_trainers AS
SELECT trainers.trainer_id, trainers.first_name,
trainers.last_name,
trainers.specialization,
COUNT(class_registrations.registration_id) as
total_registrations
FROM trainers
JOIN group_classes ON trainers.trainer_id =
group_classes.trainer_id
JOIN class_registrations ON group_classes.class_id =
class_registrations.class_id
GROUP BY trainers.trainer_id, trainers.first_name,
trainers.last_name, trainers.specialization
HAVING COUNT(class_registrations.registration_id) > (
    SELECT AVG(reg_count)
    FROM (
        SELECT COUNT(cr.registration_id) as reg_count
        FROM group_classes gc
        JOIN class_registrations cr ON gc.class_id =
cr.class_id
        GROUP BY gc.trainer_id
    ) as trainer_registrations
);
```

select * from popular_trainers;

The screenshot shows a SQL editor interface. At the top, there is a search bar with the placeholder "Введите SQL выражение чтобы отфильтровать результаты". Below it is a table titled "Таблица" (Table) with the following data:

	123 trainer_id	AZ first_name	AZ last_name	AZ specialization	123 total_registrations
1	5	Нарек	Мариян	Пилатес	2
2	2	Эмил	Алиев	Йога	2

3.2 Создать представление, которое показывает полную информацию о групповых занятиях с данными тренеров

```
CREATE VIEW classes_schedule AS
SELECT group_classes.class_id, group_classes.class_name,
group_classes.schedule_time,
group_classes.duration_minutes,
group_classes.max_participants,
trainers.first_name, trainers.last_name,
trainers.specialization,
trainers.experience_years
FROM group_classes
JOIN trainers ON group_classes.trainer_id = trainers.trainer_id
ORDER BY group_classes.schedule_time;
select * from classes_schedule;
```

select * from classes_schedule

	123 class_id	AZ class_name	123 schedule_time	123 duration_minutes	123 max_participants	AZ first_name	AZ last_name	AZ specialization	123 experience_years
1	11	Стретчинг утренний	07:00:00	30	25	Иван	Разорвин	Стретчинг	3
2	1	Утренняя йога	08:00:00	60	15	Эмил	Алиев	Йога	7
3	8	Силовой тренинг для начинающих	09:00:00	60	8	Артём	Русецкий	Силовые тренировки	5
4	10	Пилатес для беременных	10:00:00	45	6	Нарек	Марянян	Пилатес	4
5	3	Пилатес для начинающих	17:00:00	60	12	Нарек	Марянян	Пилатес	4
6	9	Функциональный тренинг	17:00:00	90	10	Артём	Русецкий	Силовые тренировки	5
7	2	Силовая тренировка	18:00:00	90	10	Артём	Русецкий	Силовые тренировки	5
8	6	Йога для продвинутых	19:00:00	75	12	Эмил	Алиев	Йога	7
9	4	Кроссфит	19:00:00	75	8	Андрей	Смаев	Кроссфит	6
10	5	Вечерний стретчинг	20:00:00	45	20	Иван	Разорвин	Стретчинг	3
11	7	Вечерняя йога	20:30:00	60	15	Эмил	Алиев	Йога	7

3. Создание таблиц

Таблица trainers

```
create table trainers (
    trainer_id serial primary key,
    first_name varchar(50) not null,
    last_name varchar(50) not null,
    specialization varchar(100) not null,
    experience_years integer check (experience_years >= 0),
    hourly_price decimal(10,2) not null check (hourly_price > 0),
    phone varchar(20)
);
```

```
insert into trainers (first_name, last_name, specialization,
experience_years, hourly_price, phone) values
('Артём', 'Русецкий', 'Силовые тренировки', 5, 2000.00, '+7-903-111-2233'),
('Эмил', 'Алиев', 'Йога', 7, 1500.00, '+7-916-222-3344'),
('Иван', 'Разорвин', 'Стретчинг', 3, 1200.00, '+7-925-333-4455'),
('Андрей', 'Смаев', 'Кроссфит', 6, 1800.00, '+7-903-444-5566'),
('Нарек', 'Марянян', 'Пилатес', 4, 1300.00, '+7-916-555-6677');
```

Таблица clients

```
create table clients (
    client_id serial primary key,
    first_name varchar(50) not null,
    last_name varchar(50) not null,
    birth_date date not null,
    phone varchar(20),
    email varchar(100) check (email ~ '^[A-Za-z0-9]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}$'),
    registration_date date
);
```

```
insert into clients (first_name, last_name, birth_date, phone,
email, registration_date) values
('Артур', 'Литвинов', '2004-11-01', '+7-903-666-7788',
'artur@mail.ru', '2025-01-10'),
('Филипп', 'Андрющенко', '2004-08-22', '+7-916-777-8899',
'filipp@gmail.com', '2025-02-15'),
```

```
( 'Михаил', 'Горбунов', '2000-01-01', '+7-925-888-9900',
'gorbunov@mail.ru', '2025-01-20'),
('Иван', 'Лобода', '2005-05-14', '+7-903-999-0011',
'loboda@gmail.com', '2025-03-05'),
('Павел', 'Швецов', '2006-07-18', '+7-916-000-1122',
'pashka@mail.ru', '2025-02-28');
```

Таблица subscriptions

```
create table subscriptions (
subscription_id serial primary key,
subscription_type varchar(50) not null,
duration_days integer not null check (duration_days > 0),
price decimal(10,2) not null check (price > 0)
);
```

```
Insert into subscriptions (subscription_type, duration_days,
price) values
('Разовый визит', 1, 500.00),
('Месячный стандарт', 30, 3000.00),
('Месячный премиум', 30, 5000.00),
('Квартальный', 90, 8000.00),
('Годовой', 365, 25000.00);
```

Таблица group_classes

```
create table group_classes (
class_id serial primary key,
class_name varchar(100) not null,
trainer_id integer not null,
schedule_time time not null,
duration_minutes integer not null check (duration_minutes > 0),
max_participants integer not null check (max_participants > 0),
foreign key (trainer_id) references trainers(trainer_id)
);
```

```
insert into group_classes (class_name, trainer_id, schedule_time,
duration_minutes, max_participants) values
('Утренняя йога', 2, '08:00:00', 60, 15),
('Силовая тренировка', 1, '18:00:00', 90, 10),
('Пилатес для начинающих', 5, '17:00:00', 60, 12),
('Кроссфит', 4, '19:00:00', 75, 8),
('Вечерний стретчинг', 3, '20:00:00', 45, 20),
('Йога для продвинутых', 2, '19:00:00', 75, 12),
('Вечерняя йога', 2, '20:30:00', 60, 15),
('Силовой тренинг для начинающих', 1, '09:00:00', 60, 8),
('Функциональный тренинг', 1, '17:00:00', 90, 10),
('Пилатес для беременных', 5, '10:00:00', 45, 6),
('Стретчинг утренний', 3, '07:00:00', 30, 25);
```

Таблица subscription_purchases

```
create table subscription_purchases (
    purchase_id serial primary key,
    client_id integer not null,
    subscription_id integer not null,
    purchase_date date,
    start_date date not null,
    end_date date not null check (end_date > start_date),
    foreign key (client_id) references clients(client_id),
    foreign key (subscription_id) references
subscriptions(subscription_id)
    unique (client_id, subscription_id, start_date)
);

insert into subscription_purchases (client_id, subscription_id,
purchase_date, start_date, end_date) values
(1, 2, '2025-09-01', '2025-09-01', '2025-10-01'),
(2, 3, '2025-09-05', '2025-09-05', '2025-10-05'),
(3, 1, '2025-09-10', '2025-09-10', '2025-09-11'),
(4, 4, '2025-09-15', '2025-09-15', '2025-12-15'),
(5, 2, '2025-09-20', '2025-09-20', '2025-10-20'),
(1, 1, '2025-09-25', '2025-09-25', '2025-09-26');
```

Таблица class_registrations

```
create table class_registrations (
    registration_id serial primary key,
    client_id integer not null,
    class_id integer not null,
    registration_date date,
    class_date date not null,
    foreign key (client_id) references clients(client_id),
    foreign key (class_id) references group_classes(class_id)
);

insert into class_registrations (client_id, class_id,
registration_date, class_date) values
(1, 1, '2025-09-26', '2025-09-27'),
(1, 3, '2025-09-26', '2025-09-28'),
(2, 2, '2025-09-27', '2025-09-28'),
(2, 4, '2025-09-27', '2025-09-29'),
(3, 1, '2025-09-28', '2025-09-29'),
(4, 5, '2025-09-29', '2025-09-30'),
(5, 3, '2025-09-30', '2025-10-01');
```