

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

Отчет по лабораторной работе № 1

Дисциплина: Системы управления базами данных

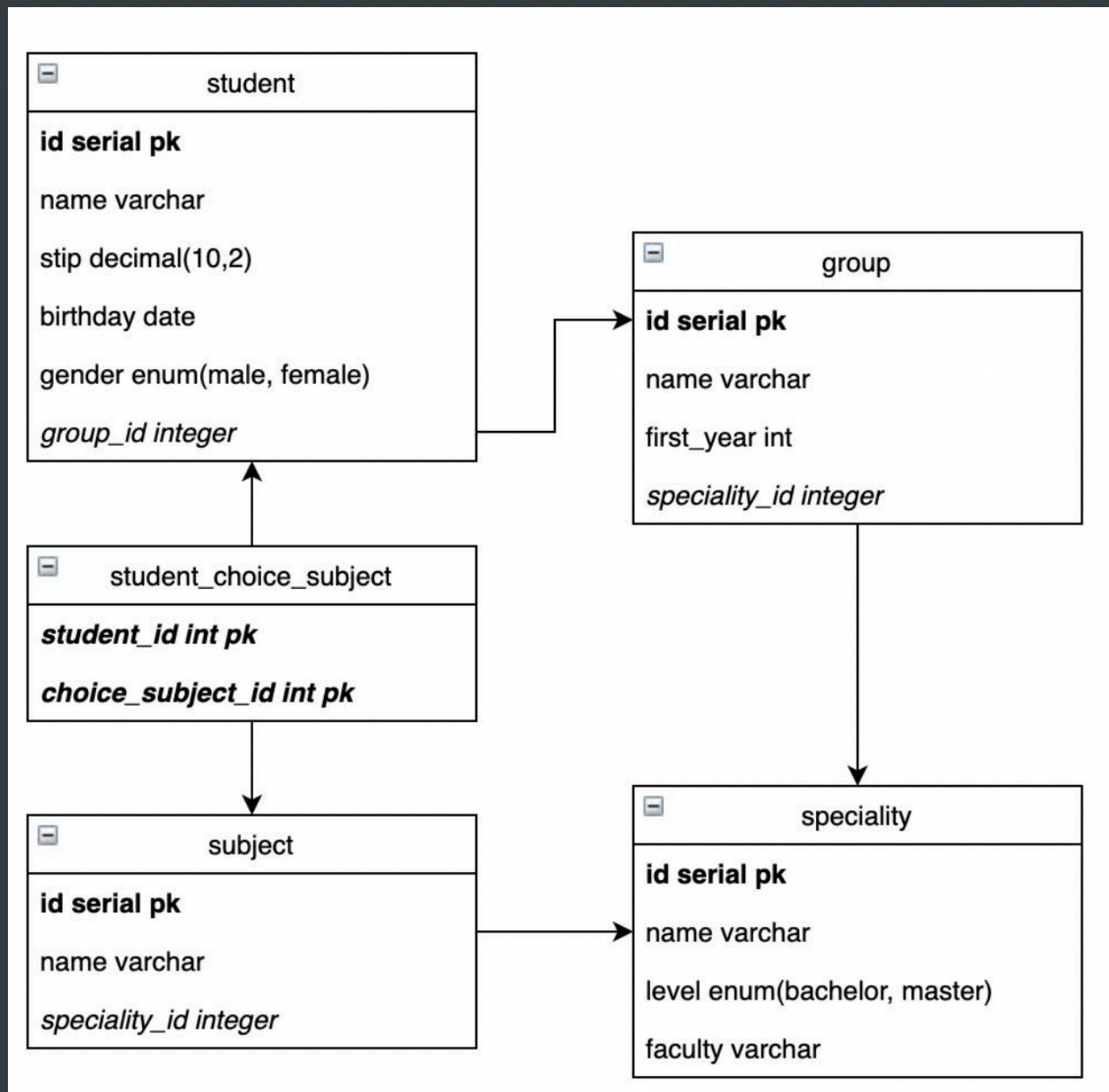
Студент: Логинов Сергей Андреевич

Группа: НФИбд-01-18

МОСКВА 2021г

Задание

- Создать с учетом первичных и внешних ключей следующие таблицы:



- Заполнить таблицы: **student** - 20 записей, **group** - 4 записи, **subject** - 5 записей, **speciality** - 2 записи
- Вывести количество мужчин и женщин в каждой группе
- Вывести количество студентов в каждой специальности упорядоченно по убыванию

Выполнение

1. Создадим требуемые таблицы:

Создадим тип enum, который представляет собой набор из 4 нужных нам значений для пола и степени:

```
5 CREATE TYPE public.enum AS ENUM
6     ('male', 'female', 'bachelor', 'master');
7
8 ALTER TYPE public.enum
9     OWNER TO postgres;
10
```

Группы. Первичный ключ - id_group, внешний - id_speciality:

pgAdmin File Object Tools Help

Browser Properties SQL

Databases (4)

lab01

- Casts
- Catalogs
- Event Triggers
- Extensions
- Foreign Data Wrappers
- Languages
- Publications

Schemas (1)

public

- Collations
- Domains
- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Procedures
- Sequences

Tables (5)

- groups
- speciality
- student
- student_choice_subject
- subject

```
1 -- Table: public.groups
2
3 -- DROP TABLE IF EXISTS public.groups;
4
5 CREATE TABLE IF NOT EXISTS public.groups
6 (
7     name character varying(100) COLLATE pg_catalog."default" NOT NULL,
8     first_year integer NOT NULL,
9     id_speciality integer NOT NULL,
10    id_group integer NOT NULL DEFAULT nextval('group_id_seq'::regclass),
11    CONSTRAINT group_pk PRIMARY KEY (id_group),
12    CONSTRAINT fk_group_spec FOREIGN KEY (id_speciality)
13        REFERENCES public.speciality (id_speciality) MATCH SIMPLE
14        ON UPDATE NO ACTION
15        ON DELETE NO ACTION
16        NOT VALID
17 )
18
19 TABLESPACE pg_default;
20
21 ALTER TABLE IF EXISTS public.groups
22     OWNER to postgres;
```

Специальности. Первичный ключ - id_speciality:

pgAdmin File Object Tools Help

Browser Properties SQL

Servers PostgreSQL 14 Databases (4) lab01 Casts Catalogs Event Triggers Extensions Foreign Data Wrappers Languages Publications Schemas (1) public Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Procedures Sequences Tables (5) groups speciality student student_choice_subject subject

```
1 -- Table: public.speciality
2
3 -- DROP TABLE IF EXISTS public.speciality;
4
5 CREATE TABLE IF NOT EXISTS public.speciality
6 (
7     name character varying(100) COLLATE pg_catalog."default" NOT NULL,
8     faculty character varying(100) COLLATE pg_catalog."default" NOT NULL,
9     id_speciality integer NOT NULL DEFAULT nextval('speciality_id_seq'::regclass),
10    level enum,
11    CONSTRAINT speciality_pk PRIMARY KEY (id_speciality)
12 )
13
14 TABLESPACE pg_default;
15
16 ALTER TABLE IF EXISTS public.speciality
17     OWNER to postgres;
```

Студенты. Первичный ключ - id_student, внешний - id_group:

pgAdmin File Object Tools Help

Browser Properties SQL

Servers PostgreSQL 14 Databases (4) lab01 Casts Catalogs Event Triggers Extensions Foreign Data Wrappers Languages Publications Schemas (1) public Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Procedures Sequences Tables (5) groups speciality student student_choice_subject subject

```
1 -- Table: public.student
2
3 -- DROP TABLE IF EXISTS public.student;
4
5 CREATE TABLE IF NOT EXISTS public.student
6 (
7     name character varying(100) COLLATE pg_catalog."default" NOT NULL,
8     stip numeric(10,0),
9     birthday date NOT NULL,
10    id_group integer NOT NULL,
11    id_student integer NOT NULL DEFAULT nextval('students_id_seq'::regclass),
12    gender enum,
13    CONSTRAINT students_pk PRIMARY KEY (id_student),
14    CONSTRAINT fk_students_groups FOREIGN KEY (id_group)
15        REFERENCES public.groups (id_group) MATCH SIMPLE
16        ON UPDATE NO ACTION
17        ON DELETE NO ACTION
18        NOT VALID
19 )
20
21 TABLESPACE pg_default;
22
23 ALTER TABLE IF EXISTS public.student
24     OWNER to postgres;
```

Выбранные студентами дисциплины. Первичный ключ здесь составной - student_id + choice_subject_id:

pgAdmin File Object Tools Help

Browser Properties SQL

Servers PostgreSQL 14 Databases (4) lab01 Casts Catalogs Event Triggers Extensions Foreign Data Wrappers Languages Publications Schemas (1) public Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Procedures Sequences Tables (5) groups speciality student student_choice_subject subject

```
1 -- Table: public.student_choice_subject
2
3 -- DROP TABLE IF EXISTS public.student_choice_subject;
4
5 CREATE TABLE IF NOT EXISTS public.student_choice_subject
6 (
7     student_id integer NOT NULL,
8     choice_subject_id integer NOT NULL,
9     CONSTRAINT student_choice_subject_pkey PRIMARY KEY (student_id, choice_s
10    CONSTRAINT fk_student FOREIGN KEY (student_id)
11        REFERENCES public.student (id_student) MATCH SIMPLE
12        ON UPDATE NO ACTION
13        ON DELETE NO ACTION
14        NOT VALID,
15    CONSTRAINT fk_subject FOREIGN KEY (choice_subject_id)
16        REFERENCES public.subject (id_subject) MATCH SIMPLE
17        ON UPDATE NO ACTION
18        ON DELETE NO ACTION
19        NOT VALID
20 )
21
22 TABLESPACE pg_default;
23
24 ALTER TABLE IF EXISTS public.student_choice_subject
25     OWNER to postgres;
```

Дисциплины. Первичный ключ - id_subject, внешний - speciality_id:

The screenshot shows the pgAdmin 4 application window. The left pane is the 'Browser' tab, displaying a hierarchical list of database objects under 'Servers' > 'PostgreSQL 14' > 'lab01'. The 'Tables (5)' section is expanded, showing 'groups', 'speciality', 'student', 'student_choice_subject', and 'subject'. The right pane is the 'SQL' tab, containing the following PostgreSQL SQL code:

```
1 -- Table: public.subject
2
3 -- DROP TABLE IF EXISTS public.subject;
4
5 CREATE TABLE IF NOT EXISTS public.subject
6 (
7     name character varying(100) COLLATE pg_catalog."default" NOT NULL,
8     speciality_id integer NOT NULL,
9     id_subject integer NOT NULL DEFAULT nextval('subject_id_seq'::regclass),
10    CONSTRAINT subject_pk PRIMARY KEY (id_subject),
11    CONSTRAINT fk_subject_spec FOREIGN KEY (speciality_id)
12        REFERENCES public.speciality (id_speciality) MATCH SIMPLE
13        ON UPDATE NO ACTION
14        ON DELETE NO ACTION
15        NOT VALID
16 )
17
18 TABLESPACE pg_default;
19
20 ALTER TABLE IF EXISTS public.subject
21     OWNER to postgres;
```

2. Заполняем таблицы записями:

Студенты:

	name character varying (100)	stip numeric (10)	birthday date	id_group integer	id_student [PK] integer	gender enum
1	sergey	2000	2000-01-05	1	101	male
2	pavel	[null]	1998-03-05	1	102	male
3	dmitriy	5000	2000-11-23	1	103	male
4	anastasia	[null]	2000-04-06	1	104	female
5	natalia	[null]	2000-03-16	1	105	female
6	veronika	5000	1999-06-01	1	106	female
7	diana	2500	1999-09-12	1	107	female
8	malik	[null]	2000-02-18	2	201	male
9	daniil	3000	1999-10-14	2	202	male
10	anna	[null]	2000-07-11	2	203	female
11	artyom	1500	1998-08-04	2	204	male
12	ilya	1500	2000-10-20	2	205	male
13	hasan	[null]	2000-01-31	3	301	male
14	sergey	2000	2001-05-05	3	302	male
15	egor	2500	2000-09-13	3	303	male
16	stanislav	2500	2000-05-14	3	304	male
17	alexandra	2000	1999-07-23	3	305	female
18	maria	10000	1999-10-12	4	401	female
19	ivan	2000	1997-08-07	4	402	male
20	saveliy	2000	1997-03-13	4	403	male

Группы:

	name character varying (100)	first_year integer	id_speciality integer	id_group [PK] integer
1	nfi-01	2018	1	1
2	nfi-02	2018	1	2
3	ni-01	2017	2	3
4	ni-02	2018	2	4

Специальности:

	name character varying (100) 	faculty character varying (100) 	id_speciality [PK] integer 	level enum 
1	fundamental informatics	ffmien		1 bachelor
2	applied informatics	ffmien		2 bachelor

Дисциплины:

	name character varying (100) 	speciality_id integer 	id_subject [PK] integer 
1	AI	2	1
2	computer graphic	2	2
3	subd	1	3
4	MM	1	11
5	math	1	12

3. Вывести количество мужчин и женщин в каждой группе:

Количество мужчин:

lab01/postgres@PostgreSQL 14 ▾

Query Editor Query History

```
1 select groups.name, sum(case when gender = 'male' then 1 else 0 end) as man from student
2   inner join groups on student.id_group = groups.id_group
3 group by groups.name order by man desc
```

Data Output Explain Notifications Messages

	name character varying (100)	man bigint
1	nfi-02	4
2	ni-01	4
3	nfi-01	3
4	ni-02	2

Количество женщин:

lab01/postgres@PostgreSQL 14 ▾

Query Editor Query History

```
1 select groups.name, sum(case when gender = 'female' then 1 else 0 end) as woman from stud
2   inner join groups on student.id_group = groups.id_group
3 group by groups.name order by woman
```

Data Output Explain Notifications Messages

	name character varying (100)	woman bigint
1	ni-02	1
2	nfi-02	1
3	ni-01	1
4	nfi-01	4

4. Вывести количество студентов в каждой специальности упорядоченно по убыванию

lab01/postgres@PostgreSQL 14 ▾

Query Editor Query History

```
1 select speciality.name, count(id_student) from student
2     inner join groups on student.id_group = groups.id_group
3     inner join speciality on groups.id_speciality = speciality.id_speciality
4 group by speciality.name order by count(id_student) desc
```

Data Output Explain Notifications Messages

	name character varying (100)	count bigint
1	fundamental informatics	12
2	applied informatics	8