

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**КАФЕДРА МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №2**  
**по дисциплине «Базы данных»**  
**Тема: Реализация базы данных в СУБД PostgreSQL**

Студент гр. 1303 \_\_\_\_\_ Насонов Я. К.  
Преподаватель \_\_\_\_\_ Заславский М. М.

Санкт-Петербург  
2023

### **Цель работы.**

Создать локальную базу данных PostgreSQL, создать необходимые таблицы, заполнить их данными. После этого написать запросы, с помощью которых выводится необходимая информация из таблиц и добавляется новая.

### **Задание.**

#### Вариант 15

Пусть требуется создать программную систему, предназначенную для работника деканата. Такая система должна обеспечивать хранение сведений о группах и студентах, а также о результатах текущей сессии. Таким образом, для каждого студента должны храниться такие данные, как фамилия, имя, отчество студента, номер его зачетной книжки, адрес постоянной прописки и адрес, по которому студент проживает, получает или нет стипендию, а также оценки, полученные в текущей сессии, и отметки о сданных зачетах. Сведения о группе – это номер группы, факультет, кафедра, специальность, к которым она относится, год формирования группы.

Работник деканата может вносить в БД следующие изменения:

- Удалить или добавить в базу студента;
- Поменять студенту номер группы, специальность, кафедру, номер зачетки;
- Занести оценки, полученные студентами на экзаменах по каждому предмету;
- По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки.

Работнику деканата могут потребоваться следующие сведения:

- Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?
- Средний балл студентов каждой группы указанного факультета?
- Средний балл по каждому предмету?
- Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?

Необходимо развернуть PostgreSQL локально:

- Написать запросы для создания таблиц из предыдущей лабораторной работы
- Заполнить тестовыми данными: 5-10 строк на каждую таблицу, обязательно наличие связи между ними, данные приближены к реальности.
- Написать запросы к БД, отвечающие на вопросы из предыдущей лабораторной работы
- Исходный код выложить на [www.db-fiddle.com](http://www.db-fiddle.com) для проверки работоспособности
- Исходный код в виде .sql файла запустить в виде PR в репо
- В отчете описать:
  - Цель
  - Текст задания в соответствии с вариантом
  - Скриншоты работы с СУБД PostgreSQL (psql / DBeaver / Datagrip, ...)
  - Скриншоты на каждый запрос (или группу запросов) на изменение/таблицы с выводом результатов (ответ)
  - Исходный код в приложении
  - Ссылку на исходный код [www.db-fiddle.com](http://www.db-fiddle.com) в приложении
  - Ссылка на PR в приложении
  - Вывод



## Выполнение работы.

### 1. Создание БД:

Для создания и заполнения таблиц базы данных написаны соответствующие запросы, также указаны в приложении Б:

```
create table faculty
(
    faculty_id int primary key generated by default as identity,
    faculty_name varchar(100)
);
```

```
✓ insert into faculty (faculty_name)
✓ values ('Art'),
        ('Music'),
        ('Science'),
        ('Physics'),
        ('Language');
```

	 faculty_id ÷	 faculty_name ÷
1	1	Art
2	2	Music
3	3	Science
4	4	Physics
5	5	Language

```
create table department
(
    department_id int primary key generated by default as identity,
    faculty_id int references faculty (faculty_id) on delete cascade,
    department_name varchar(100)
);
```

```

✓ insert into department (faculty_id, department_name)
✓ VALUES (1, 'Modern art'),
          (1, 'Medieval art'),
          (2, 'Classical music'),
          (2, 'Instrumental music'),
          (3, 'Modern science'),
          (3, 'Ancient science'),
          (4, 'Atomic physics'),
          (4, 'Basic physics'),
          (5, 'Native language'),
          (5, 'Foreign language');

```

	department_id	faculty_id	department_name
1	1	1	Modern art
2	2	1	Medieval art
3	3	2	Classical music
4	4	2	Instrumental music
5	5	3	Modern science
6	6	3	Ancient science
7	7	4	Atomic physics
8	8	4	Basic physics
9	9	5	Native language
10	10	5	Foreign language

```

create table speciality
(
    speciality_id int primary key generated by default as identity,
    department_id int references department (department_id) on delete cascade,
    speciality_name varchar(100)
);

```

```

insert into speciality (department_id, speciality_name)
values (1, 'Street art'),
      (1, 'Cynical art'),
      (2, 'Romanesque art'),
      (2, 'Gothic art'),
      (3, 'Symphony music'),
      (3, 'Opera music'),
      (4, 'Symphonic music'),
      (4, 'Chamber music'),
      (5, 'Advanced modern science'),
      (6, 'Profound ancient science'),
      (7, 'Deep atomic physics'),
      (8, 'Elementary basic physics'),
      (9, 'Culture of native language'),
      (10, 'Nuances of foreign language');

```

	speciality_id	department_id	speciality_name
1	1	1	Street art
2	2	1	Cynical art
3	3	2	Romanesque art
4	4	2	Gothic art
5	5	3	Symphony music
6	6	3	Opera music
7	7	4	Symphonic music
8	8	4	Chamber music
9	9	5	Advanced modern science
10	10	6	Profound ancient science
11	11	7	Deep atomic physics
12	12	8	Elementary basic physics
13	13	9	Culture of native language
14	14	10	Nuances of foreign language

```

create table stud_group
(
    group_id      int primary key generated by default as identity,
    speciality_id int references speciality (speciality_id) on delete cascade,
    group_number  int,
    creation_date int
);

insert into stud_group (speciality_id, group_number, creation_date)
values (1, 15, 2019),
      (1, 17, 2013),
      (1, 19, 2015),
      (2, 20, 2008),
      (3, 31, 2003),
      (3, 39, 2008),
      (4, 46, 2015),
      (4, 47, 2019),
      (4, 48, 2008),
      (5, 53, 2003),
      (5, 57, 2008),
      (6, 68, 2008),
      (7, 70, 2015),
      (7, 72, 2019),
      (7, 75, 2003),
      (7, 79, 2013),
      (8, 87, 2015),
      (8, 89, 2019),

      (9, 912, 2008),
      (10, 927, 2015),
      (11, 931, 2003),
      (12, 948, 2015),
      (13, 957, 2008),
      (14, 962, 2019);

```

	group_id	speciality_id	group_number	creation_date
1	1	1	15	2019
2	2	1	17	2013
3	3	1	19	2015
4	4	2	20	2008
5	5	3	31	2003
6	6	3	39	2008
7	7	4	46	2015
8	8	4	47	2019
9	9	4	48	2008
10	10	5	53	2003
11	11	5	57	2008
12	12	6	68	2008
13	13	7	70	2015
14	14	7	72	2019
15	15	7	75	2003
16	16	7	79	2013
17	17	8	87	2015
18	18	8	89	2019
19	19	9	912	2008
20	20	10	927	2015
21	21	11	931	2003
22	22	12	948	2015
23	23	13	957	2008
24	24	14	962	2019

```

create table student
(
    student_id      int primary key generated by default as identity,
    group_id        int references stud_group (group_id) on delete cascade,
    full_name       varchar(200),
    credit_book_number int,
    permanent_address varchar(100),
    current_address  varchar(100),
    scholarship      int default 0
),

```



```

insert into student (group_id, full_name, credit_book_number, permanent_address, current_address)
values (1, 'Steve G.T.', 152, '3rd street, 8', 'Walnut street, 9'),
      (1, 'David J.J.', 152, '1st street, 5', '2nd street, 7'),
      (1, 'Sarah T.T.', 152, '2nd street, 4', 'Walnut street, 8'),
      (4, 'Carol R.M.', 152, 'River street, 6', 'River street, 6'),
      (4, 'James W.F.', 152, '1st street, 5', '2nd street, 7'),
      (4, 'William Q.V.', 152, '3rd street, 6', '3rd street, 6'),
      (4, 'Michael L.B.', 152, 'River street, 6', 'River street, 9'),
      (6, 'Richard N.T.', 152, 'Park street, 9', 'River street, 9'),
      (7, 'Joseph O.U.', 152, 'Main street, 5', '2nd street, 6'),
      (7, 'Thomas P.W.', 152, '3rd street, 8', 'Walnut street, 9'),
      (7, 'Charles R.D.', 152, '1st street, 5', '1st street, 5'),
      (8, 'Paul B.C.', 152, 'Walnut street, 5', '2nd street, 1'),
      (8, 'Joshua Y.F.', 152, 'Oak street, 8', '3rd street, 4'),
      (11, 'Kevin E.H.', 152, 'River street, 6', 'Walnut street, 4'),
      (14, 'Betty T.J.', 152, 'Walnut street, 5', '2nd street, 3'),
      (14, 'Lisa V.K.', 152, '3rd street, 5', '3rd street, 5'),
      (14, 'Sarah N.U.', 152, '2nd street, 5', '1st street, 2'),
      (14, 'Susan D.Y.', 152, 'Walnut street, 8', 'Walnut street, 5'),
      (14, 'Ashley E.R.', 152, '1st street, 5', '2nd street, 5'),
      (15, 'Emily T.N.', 152, '1st street, 7', '1st street, 7'),
      (18, 'Amy J.B.', 152, 'River street, 5', 'Walnut street, 5'),
      (18, 'Gary W.V.', 152, '2nd street, 1', 'River street, 4'),
      (21, 'Eric E.D.', 152, 'Walnut street, 5', 'Walnut street, 3'),
      (22, 'Emma R.E.', 152, 'River street, 2', 'River street, 5');

```

	student_id	group_id	full_name	credit_book_number	permanent_address	current_address	scholarship
1	2	1	David J.J.	152	1st street, 5	2nd street, 7	0
2	3	1	Sarah T.T.	152	2nd street, 4	Walnut street, 8	0
3	5	4	James W.F.	152	1st street, 5	2nd street, 7	0
4	6	4	William Q.V.	152	3rd street, 6	3rd street, 6	0
5	7	4	Michael L.B.	152	River street, 6	River street, 9	0
6	9	7	Joseph O.U.	152	Main street, 5	2nd street, 6	0
7	11	7	Charles R.D.	152	1st street, 5	1st street, 5	0
8	12	8	Paul B.C.	152	Walnut street, 5	2nd street, 1	0
9	13	8	Joshua Y.F.	152	Oak street, 8	3rd street, 4	0
10	14	11	Kevin E.H.	152	River street, 6	Walnut street, 4	0
11	17	14	Sarah N.U.	152	2nd street, 5	1st street, 2	0
12	19	14	Ashley E.R.	152	1st street, 5	2nd street, 5	0
13	20	15	Emily T.N.	152	1st street, 7	1st street, 7	0
14	21	18	Amy J.B.	152	River street, 5	Walnut street, 5	0
15	22	18	Gary W.V.	152	2nd street, 1	River street, 4	0
16	24	22	Emma R.E.	152	River street, 2	River street, 5	0
17	8	6	Richard N.T.	152	Park street, 9	River street, 9	0
18	1	1	Steve G.T.	152	3rd street, 8	Walnut street, 9	1000
19	10	7	Thomas P.W.	152	3rd street, 8	Walnut street, 9	1000
20	18	14	Susan D.Y.	152	Walnut street, 8	Walnut street, 5	1000
21	23	21	Eric E.D.	152	Walnut street, 5	Walnut street, 3	1500
22	15	14	Betty T.J.	152	Walnut street, 5	2nd street, 3	2000
23	16	14	Lisa V.K.	152	3rd street, 5	3rd street, 5	2000

```



create table subject
(
    subject_id int primary key generated by default as identity ,
    subject_name varchar(100)
);

```

```

insert into subject (subject_name)
values ('History of art'),
      ('Rules of art'),
      ('Types of art'),
      ('Practical art'),
      ('History of music'),
      ('Rules of music'),
      ('Types of music'),
      ('Practical music'),
      ('English'),
      ('Math');

```

	 subject_id ÷	 subject_name ÷
1	1	History of art
2	2	Rules of art
3	3	Types of art
4	4	Practical art
5	5	History of music
6	6	Rules of music
7	7	Types of music
8	8	Practical music
9	9	English
10	10	Math

```

create table mark (
    mark_id int primary key generated by default as identity ,
    student_id int references student (student_id) on delete cascade,
    subject_id int references mark (mark_id) on delete cascade,
    mark_num int
);

```

```
insert into mark (student_id, subject_id, mark_num)
values
    (1, 1, 5),
    (1, 2, 4),
    (1, 3, 5),
    (1, 4, 4),

    (2, 2, 3),
    (2, 3, null),
    (2, 4, 3),

    (4, 1, 4),
    (4, 2, 3),
    (4, 3, 3),
    (4, 4, 4),

    (5, 1, 5),
    (5, 2, 2),
    (1, 4, 3),

    (7, 1, 2),
    (7, 2, null),
    (7, 3, null),
    (7, 4, 3),

    (10, 1, 4),
    (10, 2, 3),
    (10, 4, 4),
```

(15, 5, 5),

(15, 6, 5),

(15, 7, 5),

(15, 8, 5),

(16, 6, 5),

(16, 7, 5),

(16, 8, 4),

(18, 5, 3),

(18, 6, 5),

(18, 7, 4),

(19, 5, 3),

(19, 6, 4),

(19, 7, 3),

(19, 8, 4),

(21, 5, 2),

(21, 8, 3),

(23, 6, 4),

(23, 7, 4),

(23, 8, 4);

	mark_id ÷	student_id ÷	subject_id ÷	mark_num ÷
1	1	1	1	5
2	2	1	2	4
3	3	1	3	5
4	4	1	4	4
5	5	2	2	3
6	6	2	3	<null>
7	7	2	4	3
8	12	5	1	5
9	13	5	2	2
10	14	1	4	3
11	15	7	1	2
12	16	7	2	<null>
13	17	7	3	<null>
14	18	7	4	3
15	19	10	1	4
16	20	10	2	3
17	21	10	4	4
18	22	15	5	5
19	23	15	6	5
20	24	15	7	5
21	26	16	6	5
22	27	16	7	5
23	29	18	5	3
24	30	18	6	5
25	31	18	7	4
26	32	19	5	3
27	33	19	6	4
28	34	19	7	3
29	36	21	5	2
30	38	23	6	4
31	39	23	7	4

### Запросы к БД:

Запросы к БД находятся в приложении Б.

1. Удалить или добавить в базу студента.

```
-- task1: Удалить или добавить в базу студента
```

```
insert into student (group_id, full_name, credit_book_number, permanent_address, current_address, scholarship)
values (3, 'James T.R.', 1543, '1st street, 7', '1st street, 7', 0);
```

```
delete
from student
where student_id = 25;
```

21	23	21 Eric E.D.	152 Walnut street, 5	Walnut street, 3	1500
22	15	14 Betty T.J.	152 Walnut street, 5	2nd street, 3	2000
23	16	14 Lisa V.K.	152 3rd street, 5	3rd street, 5	2000
24	26	3 James T.R.	1543 1st street, 7	1st street, 7	0

2. Поменять студенту номер группы, специальность, кафедру, номер зачетки.

```
-- task2: Поменять студенту номер группы, специальность, кафедру, номер зачетки
```

```
update student
set group_id = 2
where student_id = 10;
```

23	26	3 James T.R.	1543 1st street, 7	1st street, 7	0
24	10	2 Thomas P.W.	152 3rd street, 8	Walnut street, 9	1000

3. Занести оценки, полученные студентами на экзаменах по каждому предмету

```
-- task3: Занести оценки, полученные студентами на экзаменах по каждому предмету
```

```
-- Сделано при инициализации fill.sql
```

```

insert into mark (student_id, subject_id, mark_num)
values
    (1, 1, 5),
    (1, 2, 4),
    (1, 3, 5),
    (1, 4, 4),

    (2, 2, 3),
    (2, 3, null),
    (2, 4, 3),

    (4, 1, 4),
    (4, 2, 3),
    (4, 3, 3),
    (4, 4, 4),

    (5, 1, 5),
    (5, 2, 2),
    (1, 4, 3),

```

	mark_id ÷	student_id ÷	subject_id ÷	mark_num ÷
1	1	1	1	5
2	2	1	2	4
3	3	1	3	5
4	4	1	4	4
5	5	2	2	3
6	6	2	3	<null>
7	7	2	4	3
8	12	5	1	5
9	13	5	2	2
10	14	1	4	3
11	15	7	1	2
12	16	7	2	<null>
13	17	7	3	<null>
14	18	7	4	3
15	19	10	1	4
16	20	10	2	3
17	21	10	4	4
18	22	15	5	5
19	23	15	6	5

4. По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки

```
/* task4: По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам,
   которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты,
   получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки. */
```

```
create temporary table has_2_null
as (select student_id
    from mark
    where mark_num = 2
       or mark_num is null
    group by student_id);

create temporary table has_3
as (select student_id
    from mark
    where mark_num = 3
    group by student_id);

create temporary table has_4
as (select student_id
    from mark
    where mark_num = 4
    group by student_id);
```



```
create temporary table one_3
as (select student_id
    from mark
    where mark_num = 3
    group by student_id
    having count(mark_num) = 1);
```

```
create temporary table from_another_city
as (select student_id
    from student
    where current_address <> student.permanent_address);
```

```
create temporary table more_two_4
as (select student_id
    from mark
    where mark_num = 4
    group by student_id
    having count(mark_num) > 2);
```

```

update student
set scholarship = 1000
where student_id in (select student_id
                     from mark
                     where student_id not in (select * from has_2_null)
                     and (student_id not in (select * from has_3) or
                         (student_id in (select * from one_3) and
                          student_id in (select * from from_another_city)))
                     group by student_id
                     order by student_id);

```

```

update student
set scholarship = 1500
where student_id in (select student_id
                     from mark
                     where student_id not in (select * from has_2_null)
                     and student_id not in (select * from has_3)
                     and student_id not in (select * from more_two_4));

```

```

update student
set scholarship = 2000
where student_id in (select student_id
                     from mark
                     where student_id not in (select * from has_2_null)
                     and student_id not in (select * from has_3)
                     and student_id not in (select * from has_4));

```

Задача 5: Студенты, обучающиеся на образовательной кафедре и не получающие стипендии

student_id	group_id	full_name	credit_book_number	permanent_address	current_address	scholarship
1	2	1 David J.J.		152 1st street, 5	2nd street, 7	0
2	3	1 Sarah T.T.		152 2nd street, 4	Walnut street, 8	0
3	5	4 James W.F.		152 1st street, 5	2nd street, 7	0
4	6	4 William Q.V.		152 3rd street, 6	3rd street, 6	0
5	7	4 Michael L.B.		152 River street, 6	River street, 9	0
6	9	7 Joseph O.U.		152 Main street, 5	2nd street, 6	0
7	11	7 Charles R.D.		152 1st street, 5	1st street, 5	0
8	12	8 Paul B.C.		152 Walnut street, 5	2nd street, 1	0
9	13	8 Joshua Y.F.		152 Oak street, 8	3rd street, 4	0
10	14	11 Kevin E.H.		152 River street, 6	Walnut street, 4	0
11	17	14 Sarah N.U.		152 2nd street, 5	1st street, 2	0
12	19	14 Ashley E.R.		152 1st street, 5	2nd street, 5	0
13	20	15 Emily T.N.		152 1st street, 7	1st street, 7	0
14	21	18 Amy J.B.		152 River street, 5	Walnut street, 5	0
15	22	18 Gary W.V.		152 2nd street, 1	River street, 4	0
16	24	22 Emma R.E.		152 River street, 2	River street, 5	0
17	8	6 Richard N.T.		152 Park street, 9	River street, 9	0
18	26	3 James T.R.		1543 1st street, 7	1st street, 7	0
19	18	14 Susan D.Y.		152 Walnut street, 8	Walnut street, 5	1000
20	10	2 Thomas P.W.		152 3rd street, 8	Walnut street, 9	1000
21	1	1 Steve G.T.		152 3rd street, 8	Walnut street, 9	1000
22	23	21 Eric E.D.		152 Walnut street, 5	Walnut street, 3	1500
23	15	14 Betty T.J.		152 Walnut street, 5	2nd street, 3	2000
24	16	14 Lisa V.K.		152 3rd street, 5	3rd street, 5	2000

5. Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?

```

/* task5: Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен,
с указанием группы и предмета, по которому оценка отсутствует или равна 2? */

select full_name, group_number, subject_name
from department
    inner join speciality using (department_id)
    inner join stud_group using (speciality_id)
    inner join student using (group_id)
    inner join mark using (student_id)
    inner join subject using (subject_id)
where student_id in (select * from has_2_null)
and department_name = 'Modern art'
order by full_name;

```

	full_name	group_number	subject_name
1	David J.J.	15	Rules of art
2	David J.J.	15	Types of art
3	David J.J.	15	Practical art
4	James W.F.	20	History of art
5	James W.F.	20	Rules of art
6	Michael L.B.	20	History of art
7	Michael L.B.	20	Rules of art
8	Michael L.B.	20	Types of art
9	Michael L.B.	20	Practical art

6. Средний балл студентов каждой группы указанного факультета?

-- task6: Средний балл студентов каждой группы указанного факультета?

```
select group_number, round(avg(mark_num), 2)
from faculty
    inner join department using (faculty_id)
    inner join speciality using (department_id)
    inner join stud_group using (speciality_id)
    inner join student using (group_id)
    inner join mark using (student_id)
where faculty_name = 'Art'
group by group_number
order by group_number;
```

	group_number ÷	round ÷
1	15	3.86
2	17	3.67
3	20	3

7. Средний балл по каждому предмету?

-- task7: Средний балл по каждому предмету?

```
select subject_name, round(avg(mark_num), 2)
from mark
    inner join subject using (subject_id)
group by subject_name
order by subject_name;
```

	subject_name	round
1	History of art	4
2	History of music	3.25
3	Practical art	3.4
4	Rules of art	3
5	Rules of music	4.6
6	Types of art	5
7	Types of music	4.2

8. Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?

```
-- task8: Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?

select full_name
from department inner join speciality using(department_id)
inner join stud_group using (speciality_id)
inner join student using (group_id)
where scholarship <> 0;
```

	full_name
1	Steve G.T.
2	Susan D.Y.
3	Eric E.D.
4	Betty T.J.
5	Lisa V.K.
6	Thomas P.W.

### Вывод.

При выполнении работы создана локальная база данных PostgreSQL, созданы необходимые таблицы, заполнены данными. После этого написаны запросы, с помощью которых выводится необходимая информация из таблиц и добавляется новая.

## **ПРИЛОЖЕНИЕ А**

PullRequest: <https://github.com/moevm/sql-2023-1303/pull/24>

db-fiddle : <https://www.db-fiddle.com/f/vG4PgaVB9W7Rxsobwd6tkz/2>

## ПРИЛОЖЕНИЕ Б

```
-- creating tables

create table if not exists faculty
(
    faculty_id    int primary key generated by default as identity,
    faculty_name  varchar(100)
);

create table if not exists department
(
    department_id int primary key generated by default as identity,
    faculty_id    int references faculty (faculty_id) on delete
cascade,
    department_name varchar(100)
);

create table speciality
(
    speciality_id int primary key generated by default as identity,
    department_id int references department (department_id) on
delete cascade,
    speciality_name varchar(100)
);

create table stud_group
(
    group_id        int primary key generated by default as identity,
    speciality_id int references speciality (speciality_id) on delete
cascade,
    group_number    int,
    creation_date   int
);

create table student
(
    student_id    int primary key generated by default as identity,
    group_id      int references stud_group (group_id) on delete
cascade,
    full_name     varchar(200),
    credit_book_number int,
    permanent_address varchar(100),
    current_address  varchar(100),
    scholarship     int default 0
);

create table subject
(
    subject_id int primary key generated by default as identity ,
    subject_name varchar(100)
);

create table mark (
    mark_id int primary key generated by default as identity ,
```

```

        student_id int references student (student_id) on delete cascade,
        subject_id int references mark (mark_id) on delete cascade,
        mark_num int
    );

```

```

-- filling data

```

```

insert into faculty (faculty_name)
values ('Art'),
       ('Music'),
       ('Science'),
       ('Physics'),
       ('Language');

```

```

insert into department (faculty_id, department_name)
VALUES (1, 'Modern art'),
       (1, 'Medieval art'),
       (2, 'Classical music'),
       (2, 'Instrumental music'),
       (3, 'Modern science'),
       (3, 'Ancient science'),
       (4, 'Atomic physics'),
       (4, 'Basic physics'),
       (5, 'Native language'),
       (5, 'Foreign language');

```

```

insert into speciality (department_id, speciality_name)
values (1, 'Street art'),
       (1, 'Cynical art'),
       (2, 'Romanesque art'),
       (2, 'Gothic art'),
       (3, 'Symphony music'),
       (3, 'Opera music'),
       (4, 'Symphonic music'),
       (4, 'Chamber music'),
       (5, 'Advanced modern science'),
       (6, 'Profound ancient science'),
       (7, 'Deep atomic physics'),
       (8, 'Elementary basic physics'),
       (9, 'Culture of native language'),
       (10, 'Nuances of foreign language');

```

```

insert into stud_group (speciality_id, group_number, creation_date)
values (1, 15, 2019),
       (1, 17, 2013),
       (1, 19, 2015),
       (2, 20, 2008),
       (3, 31, 2003),
       (3, 39, 2008),
       (4, 46, 2015),
       (4, 47, 2019),
       (4, 48, 2008),
       (5, 53, 2003),
       (5, 57, 2008),
       (6, 68, 2008),

```



```

(7, 70, 2015),
(7, 72, 2019),
(7, 75, 2003),
(7, 79, 2013),
(8, 87, 2015),
(8, 89, 2019),

(9, 912, 2008),
(10, 927, 2015),
(11, 931, 2003),
(12, 948, 2015),
(13, 957, 2008),
(14, 962, 2019);

```

```

insert into student (group_id, full_name, credit_book_number,
permanent_address, current_address)
values (1, 'Steve G.T.', 152, '3rd street, 8', 'Walnut street, 9'),
(1, 'David J.J.', 152, '1st street, 5', '2nd street, 7'),
(1, 'Sarah T.T.', 152, '2nd street, 4', 'Walnut street, 8'),
(4, 'Carol R.M.', 152, 'River street, 6', 'River street, 6'),
(4, 'James W.F.', 152, '1st street, 5', '2nd street, 7'),
(4, 'William Q.V.', 152, '3rd street, 6', '3rd street, 6'),
(4, 'Michael L.B.', 152, 'River street, 6', 'River street, 9'),
(6, 'Richard N.T.', 152, 'Park street, 9', 'River street, 9'),
(7, 'Joseph O.U.', 152, 'Main street, 5', '2nd street, 6'),
(7, 'Thomas P.W.', 152, '3rd street, 8', 'Walnut street, 9'),
(7, 'Charles R.D.', 152, '1st street, 5', '1st street, 5'),
(8, 'Paul B.C.', 152, 'Walnut street, 5', '2nd street, 1'),
(8, 'Joshua Y.F.', 152, 'Oak street, 8', '3rd street, 4'),
(11, 'Kevin E.H.', 152, 'River street, 6', 'Walnut street, 4'),
(14, 'Betty T.J.', 152, 'Walnut street, 5', '2nd street, 3'),
(14, 'Lisa V.K.', 152, '3rd street, 5', '3rd street, 5'),
(14, 'Sarah N.U.', 152, '2nd street, 5', '1st street, 2'),
(14, 'Susan D.Y.', 152, 'Walnut street, 8', 'Walnut street,
5'),
(14, 'Ashley E.R.', 152, '1st street, 5', '2nd street, 5'),
(15, 'Emily T.N.', 152, '1st street, 7', '1st street, 7'),
(18, 'Amy J.B.', 152, 'River street, 5', 'Walnut street, 5'),
(18, 'Gary W.V.', 152, '2nd street, 1', 'River street, 4'),
(21, 'Eric E.D.', 152, 'Walnut street, 5', 'Walnut street, 3'),
(22, 'Emma R.E.', 152, 'River street, 2', 'River street, 5');

```

```

insert into subject (subject_name)
values ('History of art'),
('Rules of art'),
('Types of art'),
('Practical art'),
('History of music'),
('Rules of music'),
('Types of music'),
('Practical music'),
('English'),
('Math');

```

```

insert into mark (student_id, subject_id, mark_num)

```

```

values
  (1, 1, 5),
  (1, 2, 4),
  (1, 3, 5),
  (1, 4, 4),

  (2, 2, 3),
  (2, 3, null),
  (2, 4, 3),

  (4, 1, 4),
  (4, 2, 3),
  (4, 3, 3),
  (4, 4, 4),

  (5, 1, 5),
  (5, 2, 2),
  (1, 4, 3),

  (7, 1, 2),
  (7, 2, null),
  (7, 3, null),
  (7, 4, 3),

  (10, 1, 4),
  (10, 2, 3),
  (10, 4, 4),

  (15, 5, 5),
  (15, 6, 5),
  (15, 7, 5),
  (15, 8, 5),

  (16, 6, 5),
  (16, 7, 5),
  (16, 8, 4),

  (18, 5, 3),
  (18, 6, 5),
  (18, 7, 4),

  (19, 5, 3),
  (19, 6, 4),
  (19, 7, 3),
  (19, 8, 4),

  (21, 5, 2),
  (21, 8, 3),

  (23, 6, 4),
  (23, 7, 4),
  (23, 8, 4);

```

```
-- some tasks on this data tables
```

```

-- task1: Удалить или добавить в базу студента

insert into student (group_id, full_name, credit_book_number,
permanent_address, current_address, scholarship)
values (3, 'James T.R.', 1543, '1st street, 7', '1st street, 7', 0);

delete
from student
where student_id = 25;

-- task2: Поменять студенту номер группы, специальность, кафедру,
номер зачетки

update student
set group_id = 6
where student_id = 8;

-- task3: Занести оценки, полученные студентами на экзаменах по
каждому предмету

-- Сделано при инициализации fill.sql

/* task4: По результатам сессии начислить стипендии студентам, не
имеющим троек или иногородним студентам,
которые имеют не более одной тройки. Право на 50 % повышение
стипендии имеют студенты,
получившие в сессию не более двух четверок, а на 100 % повышение
- студенты, сдавшие сессию на все пятерки. */

create temporary table has_2_null
as (select student_id
from mark
where mark_num = 2
or mark_num is null
group by student_id);

create temporary table has_3
as (select student_id
from mark
where mark_num = 3
group by student_id);

create temporary table has_4
as (select student_id
from mark
where mark_num = 4
group by student_id);

create temporary table one_3
as (select student_id
from mark
where mark_num = 3
group by student_id
having count(mark_num) = 1);

```

```

create temporary table from_another_city
as (select student_id
    from student
    where current_address <> student.permanent_address);

create temporary table more_two_4
as (select student_id
    from mark
    where mark_num = 4
    group by student_id
    having count(mark_num) > 2);

update student
set scholarship = 1000
where student_id in (select student_id
    from mark
    where student_id not in (select * from
has_2_null)
    and (student_id not in (select * from has_3)
or
    (student_id in (select * from one_3) and
    student_id in (select * from
from_another_city)))
    group by student_id
    order by student_id);

update student
set scholarship = 1500
where student_id in (select student_id
    from mark
    where student_id not in (select * from
has_2_null)
    and student_id not in (select * from has_3)
    and student_id not in (select * from
more_two_4));

update student
set scholarship = 2000
where student_id in (select student_id
    from mark
    where student_id not in (select * from
has_2_null)
    and student_id not in (select * from has_3)
    and student_id not in (select * from has_4));

/* task5: Студенты, обучающиеся на определенной кафедре и не сдавшие
хотя бы один экзамен,
    с указанием группы и предмета, по которому оценка отсутствует или
равна 2? */

select full_name, group_number, subject_name
from department
    inner join speciality using (department_id)
    inner join stud_group using (speciality_id)

```

```

        inner join student using (group_id)
        inner join mark using (student_id)
        inner join subject using (subject_id)
where student_id in (select * from has_2_null)
    and department_name = 'Modern art'
order by full_name;

```

-- task6: Средний балл студентов каждой группы указанного факультета?

```

select group_number, round(avg(mark_num), 2)
from faculty
    inner join department using (faculty_id)
    inner join speciality using (department_id)
    inner join stud_group using (speciality_id)
    inner join student using (group_id)
    inner join mark using (student_id)
where faculty_name = 'Art'
group by group_number
order by group_number;

```

-- task7: Средний балл по каждому предмету?

```

select subject_name, round(avg(mark_num), 2)
from mark
    inner join subject using (subject_id)
group by subject_name
order by subject_name;

```

-- task8: Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?

```

select full_name
from department inner join speciality using(department_id)
inner join stud_group using (speciality_id)
inner join student using (group_id)
where scholarship <> 0;

```