МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №  43

ОТЧЕТ

ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

### Старший преподаватель                                                               Путилова Н. В.

должность, уч. Степень, звание   подпись, дата                    инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №7

Хранимые процедуры

по курсу: ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

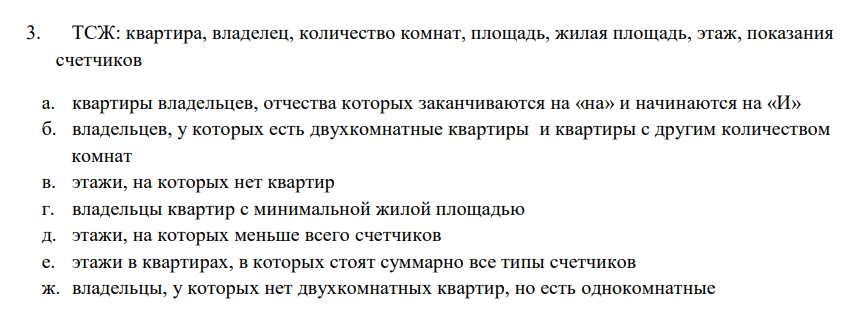
РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 4136                                                                        Бобрович Н. С.

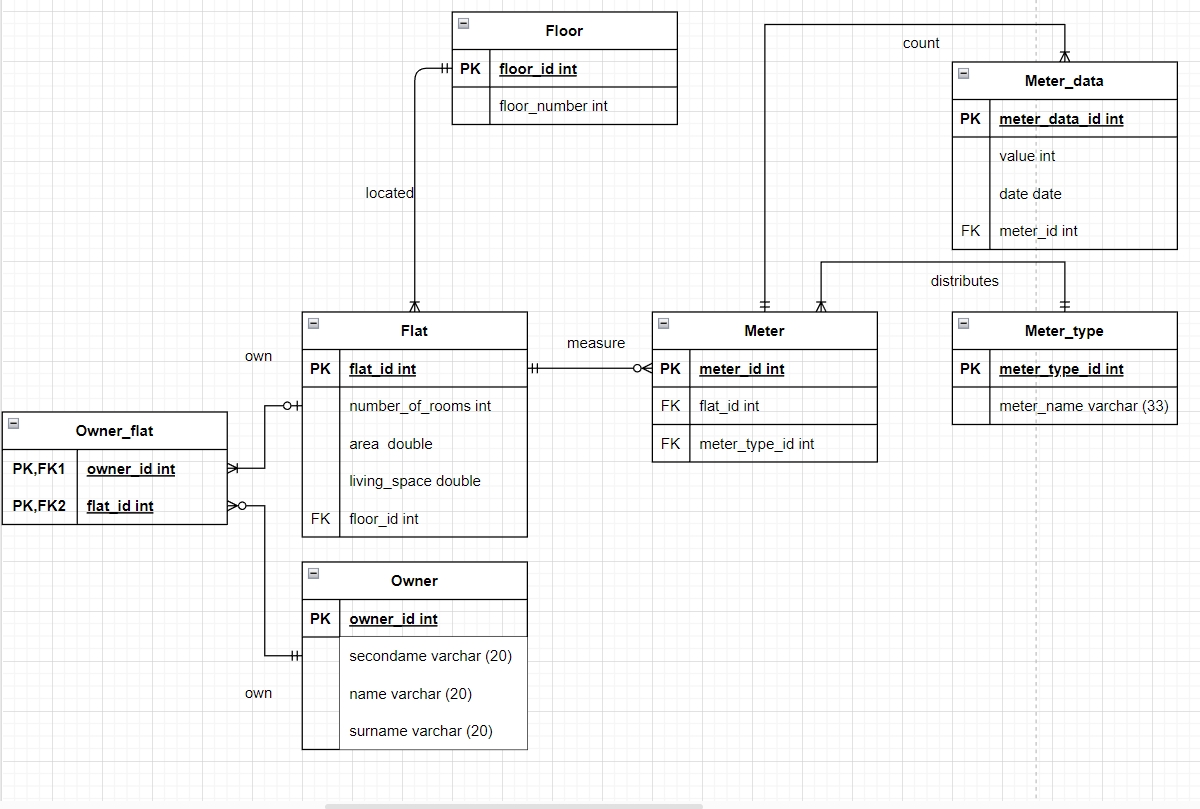
                                                                         подпись, дата                      инициалы, фамилия

Санкт-Петербург 2023

Вариант 3:



Физическая модель БД:



Назначение, тексты ХП и их вызовов:

Создание ХП:

**Процедура вставки с пополнением справочника.**

delimiter //

CREATE PROCEDURE p1 (floor\_number\_ INT, number\_of\_rooms\_ INT, area\_ DECIMAL(18,2), living\_space\_ DECIMAL(18,2))

BEGIN

declare floor\_id\_new int;

declare flat\_id\_new int;

if exists (select \* from floor where floor\_number = floor\_number\_)

then select floor\_id into floor\_id\_new from floor where floor\_number = floor\_number\_;

else begin

set floor\_id\_new = (select ifnull (max (floor\_id) + 1,0) from floor);

INSERT INTO floor (floor\_id, floor\_number) VALUES (floor\_id\_new, floor\_number\_);

end;

end if;

set flat\_id\_new = ( select ifnull(max(flat\_id)+1,0) from flat);

insert into flat (flat\_id, number\_of\_rooms, area, living\_space, floor\_id)

VALUES (flat\_id\_new, number\_of\_rooms\_, area\_, living\_space\_,floor\_id\_new);

END;//

delimiter ;

**Процедура удаления с очисткой справочника.**

delimiter //

create procedure p2(flat\_id\_del int)

BEGIN

declare floor\_id\_del int;

select floor\_id into floor\_id\_del from flat where flat\_id = flat\_id\_del;

delete from flat where flat\_id = flat\_id\_del;

if not exists(select \* from flat where floor\_id = floor\_id\_del)

then delete from floor where floor\_id = floor\_id\_del;

end if;

END;//

delimiter ;

**Процедура каскадного удаления.**

delimiter //

create procedure p3(floor\_id\_del int)

BEGIN

#delete from meter\_data where meter\_id in

#(select meter\_id from meter where flat\_id in

#(select flat\_id from flat where floor\_id = floor\_id\_del));

DELETE meter\_data

FROM meter\_data

JOIN meter ON meter\_data.meter\_id = meter.meter\_id

JOIN flat ON meter.flat\_id = flat.flat\_id

WHERE flat.floor\_id = floor\_id\_del;

delete from meter where flat\_id in

(select flat\_id from flat where floor\_id = floor\_id\_del);

delete from owner\_flat where flat\_id in

(select flat\_id from flat where floor\_id = floor\_id\_del);

delete from flat where floor\_id = floor\_id\_del;

delete from floor where floor\_id = floor\_id\_del;

END;//

delimiter ;**Процедура вычисления и возврат значения агрегатной функции.**

delimiter //

create procedure p4(out cnt\_floor int)

BEGIN

select ifnull(count(floor\_id),0) into cnt\_floor from floor;

END;//

delimiter ;

delimiter //

create function p4\_1() returns int NOT DETERMINISTIC

BEGIN

declare cnt\_floor int DEFAULT 0;

set cnt\_floor = (select ifnull(count(floor\_id),0) from floor);

return cnt\_floor;

END;//

delimiter ;

**Формирование статистики во временной таблице.**

delimiter //

create procedure p5()

BEGIN

create temporary table if not exists owner\_stat

(

id\_stat int auto\_increment primary key,

flat\_id int,

count\_owner int,

count\_owner\_avg double default 0,

diff\_cnt\_avg double default 0

);

insert into owner\_stat (flat\_id, count\_owner)

select flat.flat\_id, count(owner\_id) as count\_owner from flat

left join owner\_flat on flat.flat\_id = owner\_flat.flat\_id group by flat.flat\_id;

update owner\_stat set count\_owner\_avg =

(select avg(count\_owner) from

(select flat.flat\_id, count(owner\_id) as count\_owner from flat

left join owner\_flat on flat.flat\_id = owner\_flat.flat\_id group by flat.flat\_id)q);

update owner\_stat set diff\_cnt\_avg = count\_owner - count\_owner\_avg;

select \* from owner\_stat;

select avg (diff\_cnt\_avg \* diff\_cnt\_avg) from owner\_stat;

drop table owner\_stat;

END;//

delimiter ;

Вызовы ХП (соответственно их созданию):

call p1 (3, 5, 70.33, 68.11);

call p2 (5);

call p3 (4);

call p4(@cnt);

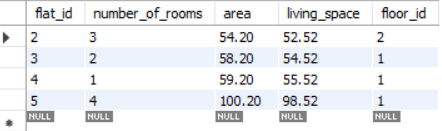
select @cnt;

select p4\_1();

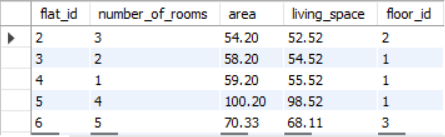
call p5 ();

Наборы данных, возвращаемые ХП:

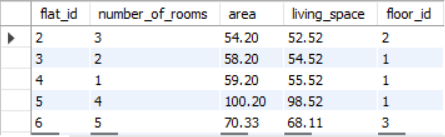
До добавления:



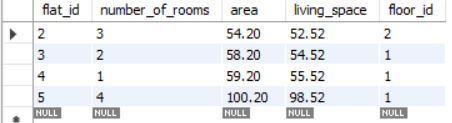
После:



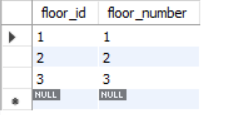
До удаления:



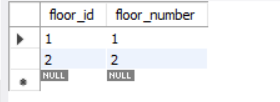
После:



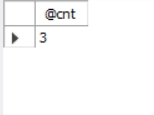
До каскадного удаления:

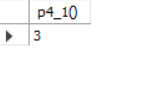


После:



Результаты ХП 4:





Результаты ХП 5:

