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 №10

Объектно-реляционные базы данных. Манипуляция данными и пользовательские операторы .

по курсу: ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

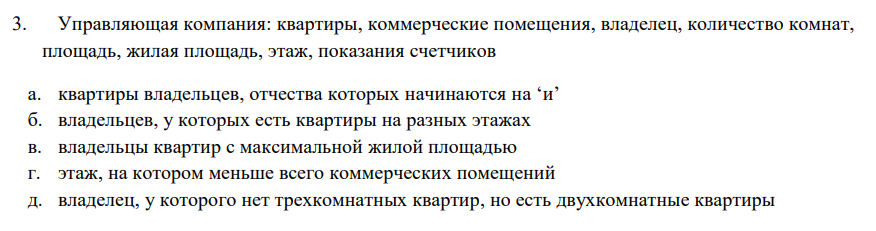
РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 4136                                                                        Бобрович Н. С.

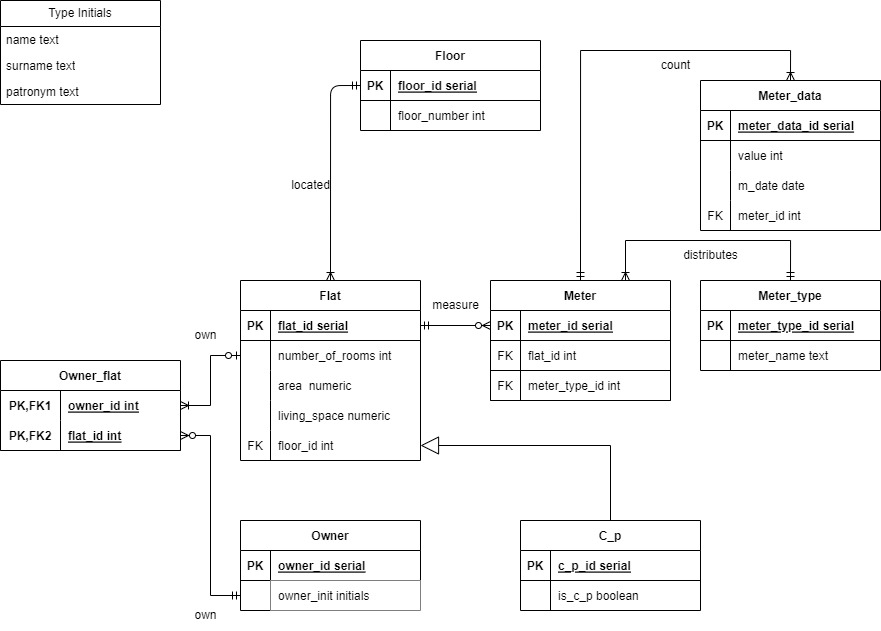
                                                                         подпись, дата                      инициалы, фамилия

Санкт-Петербург 2024

**Вариант 3:**



**Физическая модель БД:**



**Наборы данных, содержащихся в таблицах БД:**

INSERT INTO floor (floor\_number) VALUES

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

INSERT INTO flat (number\_of\_rooms, area, living\_space, floor\_id) VALUES

(3, 80.5, 60.2, 1), (2, 65.2, 45.7, 1), (1, 40.1, 30.8, 2),

(4, 95.3, 70.1, 4), (3, 75.8, 55.4, 5);

INSERT INTO c\_p (number\_of\_rooms, area, living\_space, floor\_id, is\_c\_p) VALUES

(3, 80.6, 60.3, 1, true);

INSERT INTO owner (owner\_init) VALUES

(ROW('Иванов', 'Иван', 'Иванович')), (ROW('Петрова', 'Мария', 'Сергеевна')),

(ROW('Смирнов', 'Алексей', 'Владимирович'));

INSERT INTO owner\_flat (owner\_id, flat\_id) VALUES

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

INSERT INTO meter\_type (meter\_name) VALUES

('Газ'), ('Вода');

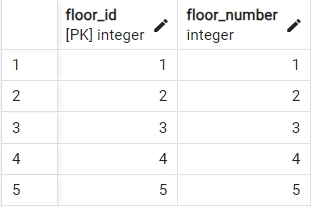
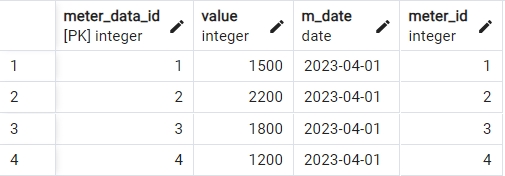
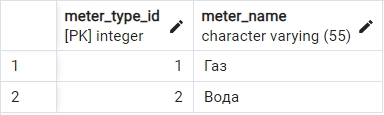
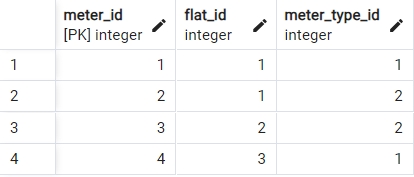
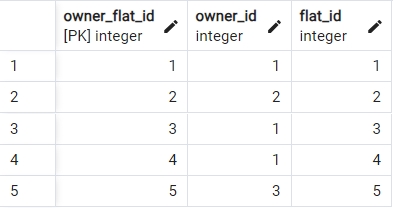
INSERT INTO meter (flat\_id, meter\_type\_id) VALUES

(1, 1), (1, 2), (2, 2), (3, 1);

INSERT INTO meter\_data (value, m\_date, meter\_id) VALUES

(1500, '2023-04-01', 1), (2200, '2023-04-01', 2), (1800, '2023-04-01', 3),

(1200, '2023-04-01', 4);



**Текст запросов на SQL:**

**Только потомок:**

SELECT c.\*

FROM c\_p c

WHERE c.is\_c\_p = true;



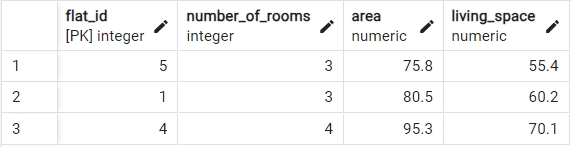
**Только предок:**

SELECT f.flat\_id, f.number\_of\_rooms, f.area, f.living\_space

FROM only flat f

WHERE f.living\_space > 50

ORDER BY f.living\_space;



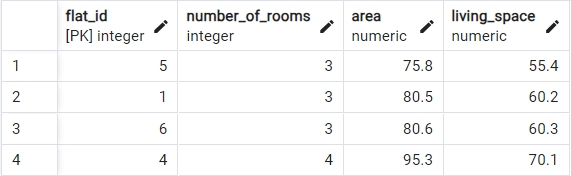
**Предок и потомок:**

SELECT f.flat\_id, f.number\_of\_rooms, f.area, f.living\_space

FROM flat f

WHERE f.living\_space > 50

ORDER BY f.living\_space;



**Пользовательский оператор:**

CREATE FUNCTION calc\_dead\_area(flat\_area numeric, living\_area numeric)

RETURNS numeric

AS $$

BEGIN

RETURN flat\_area - living\_area;

END;

$$ LANGUAGE plpgsql;

CREATE OPERATOR @@ (

LEFTARG = numeric,

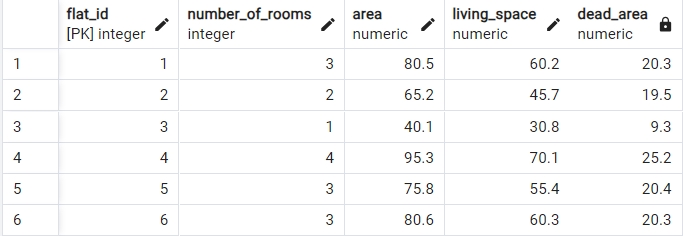
RIGHTARG = numeric,

PROCEDURE = calc\_dead\_area

);

SELECT flat\_id, number\_of\_rooms, area, living\_space, area @@ living\_space AS dead\_area

FROM flat;



**Пользовательская функция:**

CREATE FUNCTION initials\_agg\_state(agg\_state initials[], rec initials)

RETURNS initials[]

AS $$

BEGIN

IF rec.surname IS NOT NULL AND rec.name IS NOT NULL AND rec.patronym IS NOT NULL THEN

RETURN agg\_state || rec;

ELSE

RETURN agg\_state;

END IF;

END;

$$ LANGUAGE plpgsql;

CREATE AGGREGATE initials\_agg(initials)

(

sfunc = initials\_agg\_state,

stype = initials[],

initcond = '{}'

);

SELECT initials\_agg(owner\_init) AS owner\_initials

FROM owner;

