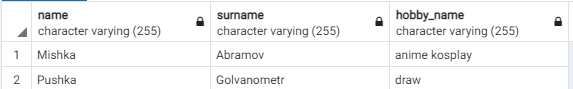
-----------------------1-------------

SELECT stud.name, stud.surname, h.hobby\_name FROM student stud

INNER JOIN student\_hobby sh ON stud.n\_z = sh.id

INNER JOIN hobby h ON h.id = sh.id



---------------------2-------------------

SELECT (NOW ()-sh.started\_at) as doing, \*

FROM student stud

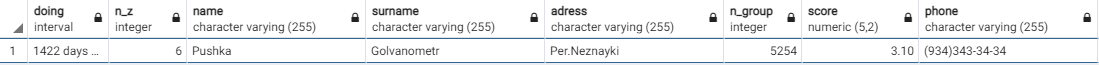
INNER JOIN student\_hobby sh ON stud.n\_z = sh.student\_id

INNER JOIN hobby h ON h.id = sh.hobby\_id

WHERE sh.finished\_at IS NULL

ORDER BY sh.started\_at

LIMIT 1



Там ещё много столбцов после

--------------------3---------------

SELECT stud.name, stud.surname, stud.n\_z, stud.age

FROM student stud LEFT JOIN (

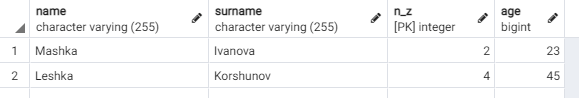
SELECT student\_id, SUM(risk) FROM student\_hobby sh

JOIN hobby h on sh.hobby\_id = h.id

GROUP BY student\_id) as nt

ON stud.id=nt.student\_id

WHERE stud.score>= (select AVG(score) FROM student) and nt.sum>9



---------------------------4----------------

SELECT stud.surname, stud.name, stud.n\_z, stud.age

FROM student stud;

SELECT h.hobby\_name

FROM hobby h

SELECT 12 \* extract(year from (sh.finished\_at, sh.started\_at)) as doing

FROM student\_hobby sh

INNER JOIN sh ON stud.n\_z = sh.student\_id

INNER JOIN hobby h ON h.id = sh.hobby\_id

WHERE sh.finished\_at IS NOT NULL

------------------------5------------------

SELECT stud.surname, stud.name, stud.n\_z, stud.age

FROM student stud

Inner JOIN (

SELECT student\_id, (count(sh.started\_at)-count(sh.finished\_at)) as counter

FROM student\_hobby sh

JOIN hobby h on sh.hobby\_id = h.id

GROUP BY student\_id) as nt

ON stud.n\_z=nt.student\_id

WHERE stud.age>19 and nt.counter>1



-----------------6-------------

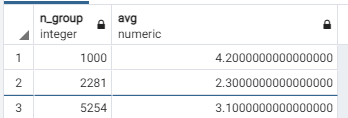
SELECT stud.n\_group, AVG(stud.score)

FROM student stud

INNER JOIN student\_hobby sh on stud.n\_z=sh.student\_id

WHERE sh.finished\_at IS NULL and sh.started\_at IS NOT NULL

GROUP BY stud.n\_group



--------------------7-------------

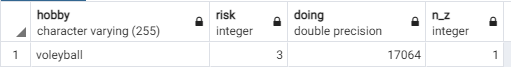
SELECT h.hobby\_name as hobby, h.risk, 12\*extract( days from (NOW ()-sh.started\_at)) as doing, stud.n\_z

FROM student stud

INNER JOIN student\_hobby sh ON stud.n\_z = sh.student\_id

INNER JOIN hobby h ON h.id = sh.hobby\_id

WHERE sh.finished\_at IS NULL and stud.n\_z=1

ORDER BY sh.started\_at

У меня выдало 2 строки, но можно и прописать limit 1

------------------8----------------

SELECT h.hobby\_name as hobby

FROM hobby h

INNER JOIN student\_hobby sh on h.id=sh.hobby\_id

INNER JOIN( SELECT stud.n\_z FROM student stud

WHERE stud.score=(select MAX(score) from student)) as nt on nt.n\_z = sh.student\_id

WHERE sh.finished\_at IS NULL



Вроде работает, а вроде и нет

Мне кажется просто не сраслось по студентам и хобби

----------------9-------------

SELECT h.hobby\_name as hobby

FROM hobby h

INNER JOIN

student\_hobby sh on h.id = sh.hobby\_id

INNER JOIN (SELECT stud.n\_z

FROM student stud

WHERE stud.score=3 and n\_group/1000=2) stud on stud.n\_z=sh.student\_id

WHERE sh.finished\_at IS NULL

В теории должно работать, потому что я у одногруппника проверял

Ну на его данных, но у меня просто нет троечников на 2 курсе

Вывод: второкурсники-умные

Ну вот что получилось просто у второкурсников

SELECT h.hobby\_name as hobby

FROM hobby h

INNER JOIN

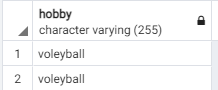
student\_hobby sh on h.id = sh.hobby\_id

INNER JOIN (SELECT stud.n\_z

FROM student stud

WHERE stud.score<6 and n\_group/1000=2) stud on stud.n\_z=sh.student\_id

WHERE sh.finished\_at IS NULL



-----------------------10----------------

Взял этот запрос у Паши, но он мне его на паре объяснил

SELECT course\_act.course

FROM

(SELECT course, COUNT(DISTINCT sh.student\_id )as zanim

FROM student\_hobby sh

INNER JOIN (

SELECT DISTINCT n\_group/1000 as course, st.n\_z

FROM student st) as nt on nt.n\_z = sh.student\_id

WHERE finished\_at IS NULL

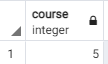
GROUP BY course) as course\_act

RIGHT JOIN

(SELECT n\_group/1000 as course, COUNT(\*) from student st GROUP BY n\_group/1000) as course\_all

on course\_act.course = course\_all.course

WHERE course\_act.zanim\*1./course\_all.count>0.5



-----------------------11--------------

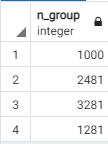
SELECT n\_group FROM student

GROUP BY n\_group

HAVING COUNT(CASE

WHEN score>=4 THEN 1

ELSE NULL END)/COUNT(\*)>=0.6



--------------12-------------

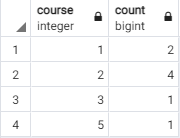
SELECT n\_group/1000 as course, COUNT(DISTINCT h.hobby\_name)

FROM student as stud

RIGHT JOIN student\_hobby as sh on sh.student\_id=stud.n\_z

Left JOIN hobby as h on h.id=sh.hobby\_id

GROUP BY n\_group/1000



-----------------13---------------

SELECT stud.n\_z, stud.surname, stud.name, stud.age, n\_group/1000 as course

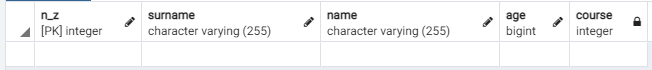
FROM student as stud

LEFT JOIN student\_hobby as sh on sh.student\_id=stud.n\_z

Left JOIN hobby as h on h.id=sh.hobby\_id

WHERE sh.hobby\_id is NULL and score=5

ORDER BY course, stud.age DESC



Ну не оказалось таких, ну бывает

--------------14--------------

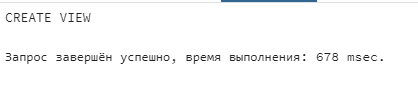
CREATE OR REPLACE VIEW full\_info AS

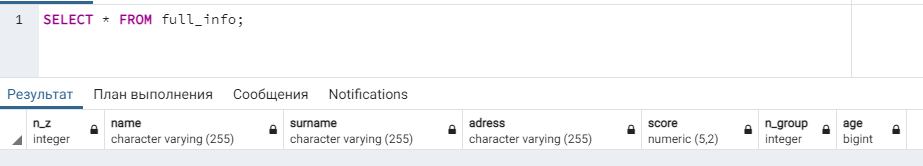
SELECT distinct stud.n\_z, stud.name, stud.surname,stud.adress,stud.score,stud.n\_group,stud.age

FROM student stud

RIGHT JOIN student\_hobby sh on stud.n\_z=sh.student\_id

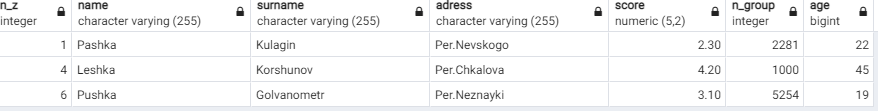
WHERE finished\_at is null and extract(year from (clock\_timestamp ( )-sh.started\_at))>5





Никого у меня нет, кто дольше 5 лет занимается

А это без условия на 5 лет



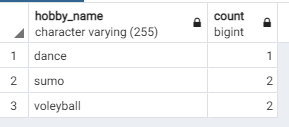
-----------------------15--------------

SELECT h.hobby\_name, COUNT((sh.student\_id, sh.hobby\_id)) as count

FROM hobby h LEFT JOIN student\_hobby sh on h.id=sh.hobby\_id

WHERE sh.finished\_at is null

GROUP BY h.hobby\_name



У меня всего 5 null у finished\_at, поэтому всё ок

--------------16------------------

SELECT id FROM(

SELECT h.id, COUNT((sh.student\_id, sh.hobby\_id))

FROM hobby h LEFT JOIN student\_hobby sh on h.id=sh.hobby\_id

WHERE sh.finished\_at is null

GROUP BY h.id

ORDER BY count desc limit 1) as popular\_hobby



-----------17------------

SELECT \* FROM student stud RIGHT JOIN (

Select student\_id from student\_hobby sh

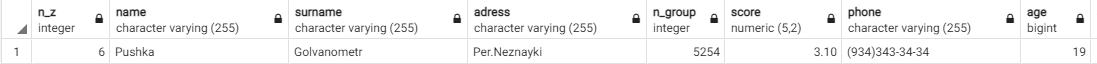
RIGHT JOIN( SELECT h.id FROM hobby h LEFT JOIN student\_hobby sh on h.id=sh.hobby\_id

WHERE sh.finished\_at is null

GROUP BY h.id

ORDER BY COUNT(distinct (sh.student\_id, sh.hobby\_id)) desc limit 1) as besth on besth.id=sh.hobby\_id

WHERE finished\_at is null) as stid on stud.n\_z = stid.student\_id

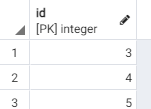


И ещё колонка со student\_id =6

--------------18--------------

SELECT id FROM hobby ORDER BY risk desc limit 3

Мне страшно, что такой маленький запрос получился



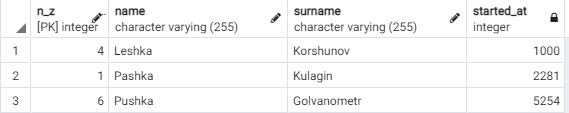
---------------------19-------------------

SELECT distinct stud.n\_z, stud.name, stud.surname,stud.n\_group sh.started\_at FROM

student stud RIGHT JOIN student\_hobby sh on stud.n\_z=sh.student\_id

WHERE sh.finished\_at is null

ORDER BY sh.started\_at limit 5



О, там даже я есть(самый первый)

-------------------20----------------

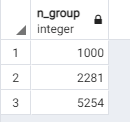
SELECT distinct n\_group from

(SELECT distinct stud.n\_z, stud.name, stud.surname,stud.n\_group, sh.started\_at FROM

student stud RIGHT JOIN student\_hobby sh on stud.n\_z=sh.student\_id

WHERE sh.finished\_at is null

ORDER BY sh.started\_at limit 5 ) as groupsss



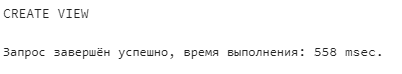
----------21-----------------

CREATE OR REPLACE VIEW not\_full\_info AS

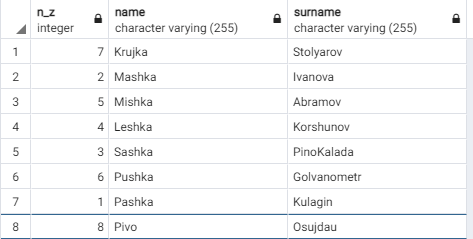
SELECT n\_z, name, surname

FROM student

ORDER BY score desc







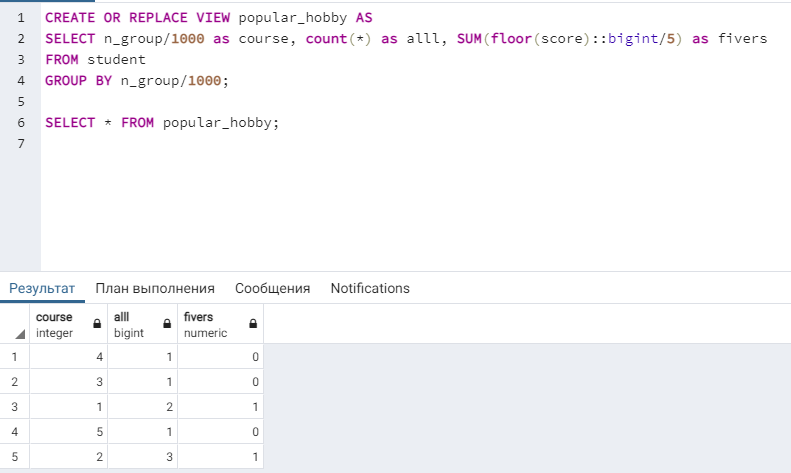
-----------------22-----------

Фиг знает как сделать, что-то пробовал но ничего не получилось

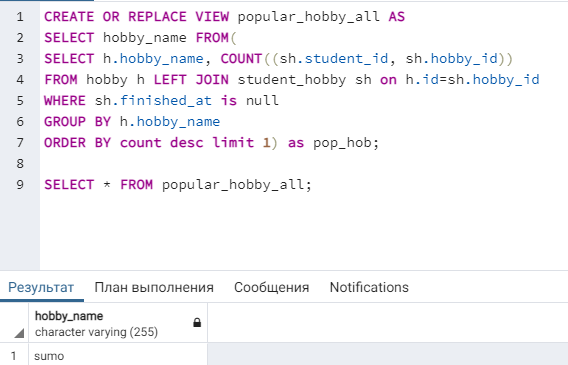
----------------23-----------

Тоже самое

-----------------24-------------



--------------------25----------------------



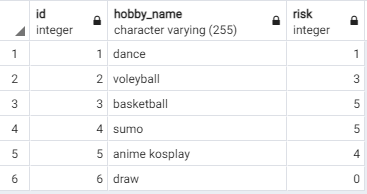
Ахахахахаха

------------------26---------------------

CREATE VIEW updatable as

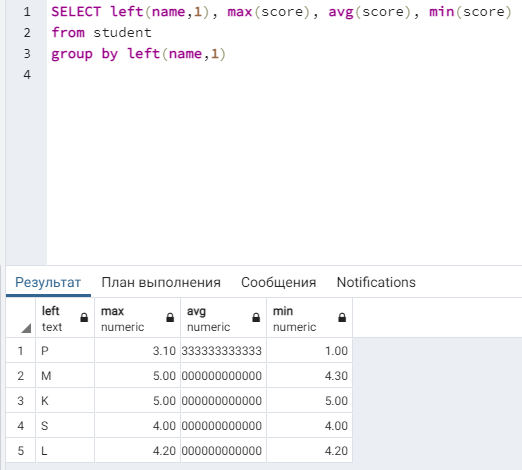
SELECT \* FROM hobby;

SELECT \* FROM updatable;



В таких представлениях мы можем изменить или удалить строки или добавить в них новые строки.

-----------27-----------------



И добавить HAVING max(score)>3.6

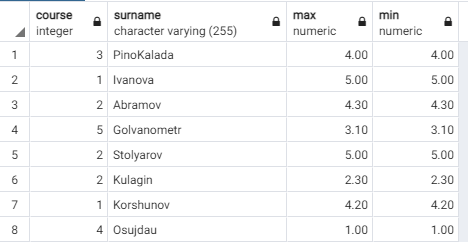
Тогда улетят Pashka, Pushka и Pivo(без буквы P)

----------28----------------

SELECT n\_group/1000 as course, surname, max(score),min(score)

from student

group by n\_group/1000, surname

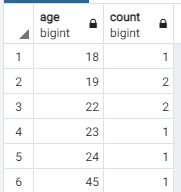


-------------29-------------

SELECT age, COUNT(distinct hobby\_id)

FROM student stud RIGHT JOIN student\_hobby sh on stud.n\_z = sh.student\_id

GROUP BY age



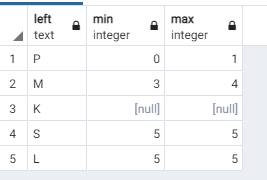
-----------------30---------------

SELECT left(stud\_tabl.name,1), min(risk), max(risk)

FROM (student stud RIGHT JOIN student\_hobby sh on stud.n\_z = sh.student\_id) as stud\_tabl

LEFT JOIN hobby h on h.id=stud\_tabl.student\_id

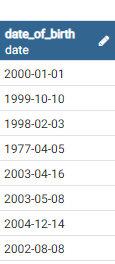
GROUP BY left(stud\_tabl.name,1)



---------------31---------------

Тут Паша сказал создать колонку с годом рождения, ну я послушался и создал

Через след запросы



ALTER TABLE student

ADD date\_of\_birth DATE;

Потом добавим ограничения

ALTER TABLE student

ALTER COLUMN date\_of\_birth

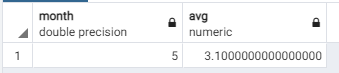
SET NOT NULL;

SELECT extract(month from date\_of\_birth) as month, avg(score)

FROM (student stud right join student\_hobby sh on sh.student\_id=stud.n\_z) as month\_table left join hobby h on h.id=month\_table.hobby\_id

WHERE finished\_at is null and h.hobby\_name='dance'

GROUP BY month



-------------32------------

SELECT distinct 'Имя: '||stud.name||', фамилия: '||stud.surname||', группа: '||stud.n\_group #эта всё одна строчка

FROM student stud RIGHT JOIN student\_hobby sh on stud.n\_z=sh.student\_id



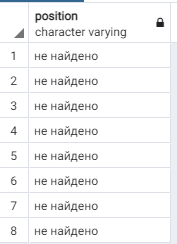
-------------------33--------------

SELECT case position('ов' in surname)::varchar

when '0' then 'не найдено'

else position('ов' in surname)::varchar end

FROM student



-------------------34----------------

SELECT case

when (length(surname)>10) then surname

else rpad(surname,10,'#') end

FROM student



---------35----------------

SELECT rtrim(surname,'#')

FROM( SELECT case

when (length(surname)>10) then surname

else rpad(surname,10,'#')

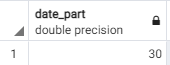
end as surname

from student) as norm\_tabl



------------36------------

SELECT extract(days FROM date\_trunc('month', '4-1-2018'::date) + interval '1 month - 1 day');



С февралём всё тоже работает

SELECT extract(days FROM date\_trunc('month', '2-1-2020'::date) + interval '1 month - 1 day');



SELECT extract(days FROM date\_trunc('month', '2-1-2018'::date) + interval '1 month - 1 day');

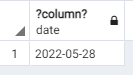


-----------37-----------

Нашёл 2 способа на stackoverflow пришлось их чуток переделать

SELECT current\_date + cast(abs(extract(dow FROM current\_date) - 7) - 1 AS int);

SELECT current\_date+7 - (( cast(extract(dow FROM current\_date) AS int)+1) %8)

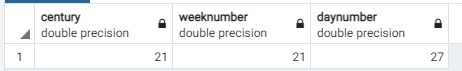


---------38--------------

Select

extract(century from current\_date) as century,

extract(weeks from current\_date) as weeknumber,

extract(days from current\_date) as daynumber; 

---------39-------------

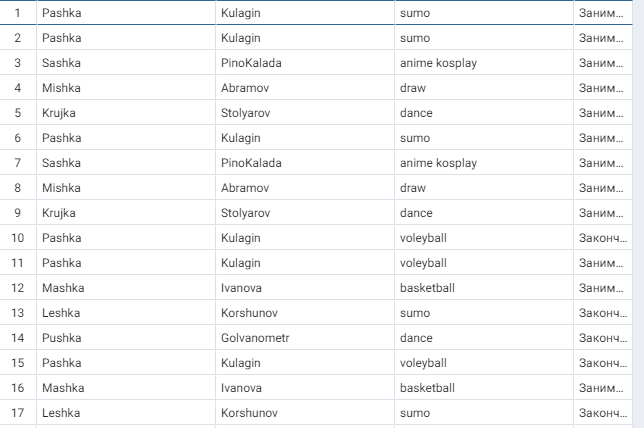
SELECT stud.name, stud.surname, h.hobby\_name,

case

when (sh.finished\_at is null) then 'Закончил'

else 'Занимается' end

FROM student stud RIGHT JOIN student\_hobby sh on stud.n\_z=sh.student\_id LEFT JOIN hobby h on h.id = sh.hobby\_id



Наконец-то привычные запросы

А то прошлые с датами мне не понравились

---------------------ураааа последний---------------

SELECT n\_group,

count(case round(score) when 2 then 1 end) as "2",

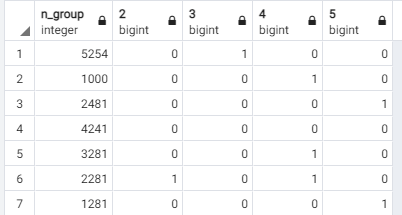
count(case round(score) when 3 then 1 end) as "3",

count(case round(score) when 4 then 1 end) as "4",

count(case round(score) when 5 then 1 end) as "5"

from student

group by n\_group



Всё

В целом то, что было непонятно я спросил у Паши с Костей, а так нужно всё на практике фигачить